

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
Ministère de l'enseignement supérieur et de la recherche scientifique

Université Mohammed Seddik Benyahia- Jijel
Faculté des Sciences et de la Technologie
Département d'Electronique



THESE

Présentée pour l'obtention du diplôme de

Doctorat 3^{ème} cycle LMD

Spécialité : **ELECTRONIQUE ET ANALYSE DES SYSTEMES**

Présentée par

Mme. Kheira KAHILI

Intitulée

CONTRIBUTION A LA COMMANDE INTELLIGENTE DES SYSTEMES

MIMO NON LINEAIRES INCERTAINS

Présentée le: 11 juillet 2022

Devant le jury :

Président :	Abdelkrim BOUKABOU	Prof, Université de Jijel
Rapporteur :	Omar BOUHALI	Prof, Université de Jijel
Co-rapporteur:	Nassim RIZOUG	HDR, ESTACA France
Examineurs :	Ammar SOUKKOU	MCA, Université de Jijel
	Mohamed GUIATNI	Prof, EMP Alger
	Fouad YACEF	MRA, CDTA Alger

Dédicace

À ma mère

À mon père

À mes frères

À toute ma Famille

À tous mes enseignantes et enseignants depuis le primaire

À mes amis et collègues

ملخص

هذه الأطروحة مخصصة للتحكم الذكي في الأنظمة غير الخطية غير المؤكدة. أولاً ، نقدم تصميم وحدة التحكم الذكية العصبية التكيفية المباشرة وغير المباشرة القائمة على الموجة (WNN) للتحكم في أنظمة SISO غير الخطية غير المؤكدة. نتعامل مع مشكلة اختيار وظيفة تنشيط الموجات وتأثيرها على أداء جهاز التحكم العصبي التكيفي من خلال دراسة مقارنة بين العديد من وظائف تنشيط الموجات. بعد ذلك ، نقدم امتداد وحدة التحكم هذه للتحكم التكيفي لأنظمة MIMO غير الخطية غير المؤكدة مع الاختيار الأمثل لوظيفة تنشيط الموجات التي حصلت عليها الدراسة المقارنة. بعد ذلك ، نعالج مشكلة التحكم في الضبابية التكيفية القائمة على الموجة لأنظمة MIMO غير الخطية غير المؤكدة. بعد ذلك ، نستغل نتائج هذه الدراسات لتطوير وحدة تحكم FS-WNN هجينة جديدة تجمع بين وحدة التحكم WNN التكيفية ووحدة التحكم FS التكيفية من خلال تقنية التعديل. تحدد تقنية التعديل درجة مساهمة لكل وحدة تحكم ذكية. الغرض من هذه المجموعة هو دمج مزايا وحدة التحكم WNN ووحدة التحكم FS في شبكة هجينة واحدة وتحقيق تعميم أفضل. المساهمة الثانية لهذه الأطروحة هي تطوير خوارزمية جديدة للتكيف عبر الإنترنت لمعاملات التعديل لضبط في الوقت الحقيقي درجات مساهمة الأنظمة الذكية التي تشكل وحدة تحكم FS-WNN الهجينة. تظهر نتائج المحاكاة على روبات مناور أن خوارزمية التكيف المقترحة توفر لوحدة تحكم FS-WNN الهجينة مزيداً من المرونة وجودة أفضل للتقريب مقارنةً بوحدة التحكم FS-WNN مع معاملات التعديل الثابتة مسبقاً من قبل المصمم. المساهمة الثالثة لهذه الأطروحة هي تطوير قانون تحكم ذكي متكيف يعتمد على FS-WNN الهجين مع معاملات تعديل متغيرة للتحكم في فئة معقدة من أنظمة MIMO غير المؤكدة غير المؤكدة. يتم تنفيذ تطبيق عن طريق المحاكاة على صندوق أدوات Pixhawk PX4 quadrotor لتسليط الضوء على صلاحية وتفوق قانون التحكم الجديد الذي تم تطويره في إطار هذه الأطروحة.

الكلمات المفتاحية: الأنظمة الغير خطية الغير مؤكدة، MIMO، النمذجة، التحكم التكيفي المباشر، التحكم التكيفي الغير مباشر، الشبكات العصبية، الموجات، أنظمة المقاربة الضبابية، التهجين، الاس-تقرار معاملات التعديل.

Abstract

This thesis is dedicated to the intelligent control of uncertain nonlinear systems. First, we present the design of the wavelet-based direct and indirect adaptive neural smart controller (WNN) for the control of uncertain nonlinear SISO systems. We deal with the problem of the choice of the wavelet activation function and its influence on the performance of the adaptive neural controller through a comparative study between several wavelet activation functions. Then, we present the extension of this controller for the adaptive control of uncertain nonlinear MIMO systems with the optimal choice of the wavelet activation function obtained by the comparative study. Next, we address the problem of wavelet-based adaptive fuzzy control (FS) of uncertain nonlinear MIMO systems. Subsequently, we exploit the results of these studies to develop a new hybrid FS-WNN controller which combines the adaptive WNN controller with the adaptive FS controller through the modulation technique. The modulation technique assigns a degree of contribution to each intelligent controller. The purpose of this combination is to integrate the advantages of WNN controller and FS controller into a single hybrid network and achieve better generalization. The second contribution of this thesis is the development of a new online adaptation algorithm of the modulation parameters to adjust in real time the degrees of contribution of the intelligent systems which form the hybrid FS-WNN controller. The simulation results on a manipulator robot show that the proposed adaptation algorithm offers the hybrid FS-WNN controller more flexibility and a better quality of approximation compared to the FS-WNN controller with modulation parameters fixed a priori by the designer. The third contribution of this thesis is the development of an adaptive intelligent control law based on hybrid FS-WNN with variable modulation parameters for the control of a complex class of underactuated uncertain MIMO systems. An application by simulation on a Pixhawk PX4 quadrotor toolbox is carried out to highlight the validity and the superiority of the new control law developed within the framework of this thesis.

Keywords: Uncertain nonlinear systems, MIMO, Modeling, Direct adaptive control, Indirect adaptive control, Neural networks, Wavelets, Fuzzy systems, Hybridization, Stability, Modulation.

Résumé

Cette thèse est dédiée au contrôle intelligent des systèmes non linéaires incertains. D'abord, nous présentons la conception du contrôleur intelligent neuronal adaptatif direct et indirect à base d'ondelettes (WNN) pour la commande des systèmes SISO non linéaires incertains. On traite le problème du choix de la fonction d'activation ondelette et son influence sur les performances du contrôleur neuronal adaptatif à travers une étude comparative entre plusieurs fonctions d'activation ondelettes. Ensuite, nous présentons le contrôle adaptatif des systèmes MIMO non linéaires incertains. Après, nous abordons le problème de contrôle flou adaptatif (FS) à base d'ondelettes des systèmes MIMO non linéaires incertains. La première contribution de cette thèse est l'exploitons de ces connaissances sur les systèmes intelligents WNN et FS pour développer un nouveau contrôleur FS-WNN hybride qui combine le contrôleur WNN adaptatif avec le contrôleur FS adaptatif à travers la technique de modulation. La technique de modulation attribue un degré de contribution à chaque contrôleur intelligent. L'objectif de cette combinaison est l'intégration des avantages du contrôleur WNN et du contrôleur FS dans un seul réseau hybride et l'obtention d'une meilleure généralisation. La deuxième contribution de cette thèse est le développement d'un nouvel algorithme d'adaptation en ligne des paramètres de modulation pour ajuster en temps réel les degrés de contribution des systèmes intelligents qui forment le contrôleur FS-WNN hybride. Les résultats de simulation sur un robot manipulateur montrent que l'algorithme d'adaptation proposé offre au contrôleur FS-WNN hybride plus de flexibilité et une meilleure qualité d'approximation par rapport au contrôleur FS-WNN avec des paramètres de modulation fixés à priori par le concepteur. La troisième contribution de cette thèse est le développement d'une loi de commande intelligente adaptative à base du FS-WNN hybride avec des paramètres de modulation variables pour le contrôle du quadrotor. Une application par simulation sur un toolbox de quadrotor Pixhawk PX4 est réalisée pour mettre en évidence la validité et la supériorité de la nouvelle loi de commande développée dans le cadre de cette thèse.

Mots clés: Systèmes non linéaires incertains, MIMO, Modélisation, Commande adaptative directe, Commande adaptative indirecte, Réseaux de neurones, Ondelettes, Systèmes flous, hybridation, Stabilité, Modulation.

Contributions

Publications:

- 1- Kheira Kahili, Omar Bouhali, Nassim Rizoug et Fouad Khenfri "Hybrid FS–WNN approximator in indirect adaptive control of uncertain non-linear MIMO systems," *IET Control Theory & Applications*, 2019, pp. 1-12, doi: 10.1049/iet-cta.2019.0078

Conferences:

- 1- K. Kahili, O. Bouhali, N. Rizoug et F. Khenfri, "Novel Hybrid Intelligent Controller for Chaotic Engineering Systems," International Workshop on Healthy – Energy Efficiency & Intelligent Building Systems (HEIBS), Paris, France. July 2021,
- 2- K. Kahili, O. Bouhali, F. Khenfri et N. Rizoug, "Novel Hybrid Intelligent Backstepping Controller for Chaotic Systems," CoDIT'20: 7thInternational Conference on Control, Decision and Information Technologies, Prague, Czech Republic, 2020.
- 3- K. Kahili, O. Bouhali, F. Khenfri et N. Rizoug, "Robust Intelligent Self-tuning PID Controller for the Body-rate Stabilization of Quadrotors," *IECON 2019 - 45th Annual Conference of the IEEE Industrial Electronics Society*, Lisbon, Portugal, 2019, pp. 5281-5286. doi: 10.1109/IECON.2019.8927751
- 4- K. Kahili, O. Bouhali et N. Rizoug, "Wavelet Functions Performances in Intelligent Direct Adaptive Control of Uncertain Nonlinear Systems," *2018 International Conference on Electrical Sciences and Technologies in Maghreb (CISTEM)*, Algiers, 2018, pp. 1-6. doi: 10.1109/CISTEM.2018.8613605

Remerciements

Je remercie dieu le tout puissant.

Je tiens tout d'abord à exprimer ma reconnaissance envers les membres du jury, dont la renommée et la qualité scientifique honore grandement ce travail :

– Monsieur Abdelkrim BOUKABOU, professeur à l'université de Jijel, pour m'avoir fait l'honneur de présider le jury de ma thèse.

– Monsieur Ammar SOUKKOU, Maitre de conférences à l'université de Jijel, Monsieur Mohamed GUIATNI, professeur à l'école militaire polytechnique (EMP) de Alger, et Monsieur Fouad YACEF, Maitre de recherche à centre de développement de technologies avancées (CDTA) de Alger. pour m'avoir fait l'honneur d'être examinateurs de ce travail.

Tout d'abord, j'adresse mes sincères remerciements au Prof. **Omar BOUHALI**⁽¹⁾, mon directeur de thèse de l'université de Jijel. Je lui remercie de m'avoir encadré, orienté, aidé et conseillé. Je lui suis infiniment reconnaissante pour ses conseils, sa patience et sa rigueur scientifique. Sans lui, ce travail ne serait pas ce qu'il est aujourd'hui et n'aurait même jamais vu le jour. Je le remercie également pour les lectures attentives qu'il a faites de ce mémoire aux différentes étapes de son accomplissement et pour les bons conseils qu'il m'a donné face aux différents problèmes que j'ai rencontré.

Ce travail a été réalisé en collaboration entre le laboratoire de Mécatronique de l'université de Jijel LMT'Lab et le laboratoire des systèmes et énergies embarqués pour les transports S2ET de l'école ESTACA France.

Je tiens à exprimer toute ma gratitude à tous les membres de laboratoire LMT de l'université de Jijel et plus précisément **Fatima Zahra KHEMILI**, **Lyas BEKRAR**, et **Youssef BIBI** qui m'ont accompagné durant ce travail et pour l'ambiance familiale. Je tiens à remercier le personnel du service de vice-rectorat de la post-graduation de l'université de Jijel notamment **Samira**, **Amira** et **Dalila** pour tous leurs services et leur encouragement pendant toute la durée de ma thèse.

Ensuite, J'adresse de chaleureux remerciements au Prof. **Fouad KHENFRI**⁽²⁾, mon co-directeur de thèse durant la bourse Profas b+2018/2019, et enseignant chercheur dans le laboratoire systèmes et énergies embarqués pour le transport S2ET de l'ESTACA, C'est un honneur pour moi de travailler avec lui et je ne peux qu'admirer son talent. Je lui suis infiniment reconnaissante, non seulement parce qu'il a accepté de me prendre en thèse, mais aussi parce qu'il a partagé ses idées avec moi. Il a

dirigé ma thèse avec beaucoup de patience et il a dédié beaucoup de temps à mon travail en étant toujours très disponible, ce qui m'a énormément encouragée.

Je remercie particulièrement le HDR. **Nassim RIZOUG**⁽³⁾, mon co-directeur de thèse et enseignant chercheur dans le laboratoire systèmes et énergies embarqués pour le transport S2ET de l'ESTACA, de m'avoir accueillie dans son équipe de recherche et d'avoir accepté de diriger ce travail. Je lui remercie pour sa rigueur scientifique, sa disponibilité et ses qualités humaines qui m'ont profondément touchée.

Je remercie également le Prof. **Cherif LAROUCI**⁽⁴⁾, Responsable du pôle Systèmes et énergies embarqués pour le transport S2ET, qui m'a accueilli au sein de son laboratoire. Mes remerciements s'adressent également à **Sandrine PIOGER**⁽⁵⁾, Responsable administrative de l'ESTACA, pour sa gentillesse et pour toute l'aide qu'elle m'a apportée durant mon séjour à L'ESTACA Campus Ouest.

Enfin mes remerciements les plus chaleureux vont à tous mes collègues et membres ESTACA pour leurs encouragements et pour l'ambiance agréable tout au long de mon stage, grâce à eux j'ai passé d'excellents moments.

Je n'aurais jamais pu réaliser ce travail doctoral sans le soutien de ces personnes dont la générosité, la bonne humeur et l'intérêt manifestés à l'égard de ma recherche m'ont permis de progresser dans ma thèse.

(1) Prof. Omar BOUHALI

Mon directeur de thèse, vice-recteur chargé des relations extérieures de l'université Mohamed Seddik Benyahia de Jijel.

(2) Prof. Fouad KHENFRI

Mon co-directeur de thèse durant la bourse Profas b+2018/2019, Enseignant-Chercheur à L'ESTACA Campus Ouest, et membre du pôle S2ET (Systèmes et Energies Embarqués pour les Transports).

(3) HDR. Nassim RIZOUG

Mon co-directeur de thèse, Enseignant-Chercheur à L'ESTACA Campus Ouest, et membre du pôle S2ET (Systèmes et Energies Embarqués pour les Transports).

(4) Prof. Cherif LAROUCI

Responsable du pôle S2ET (Systèmes et Energies Embarqués pour les Transports), ESTACA.

(5) Mme. SANDRINE Pioger

Responsable appui administratif de l'ESTACA.

Table des matières

<i>Dédicace</i>	<i>i</i>
<i>ملخص</i>	<i>ii</i>
<i>Abstract</i>	<i>iii</i>
<i>Résumé</i>	<i>iv</i>
<i>Contributions</i>	<i>v</i>
<i>Remerciements</i>	<i>vi</i>
<i>Table des matières</i>	<i>viii</i>
<i>Liste des nomenclatures</i>	<i>xi</i>
<i>Liste des abréviations</i>	<i>xiii</i>
<i>Liste des tableaux</i>	<i>xiv</i>
<i>Liste des Figures</i>	<i>xv</i>
<i>Introduction Générale</i>	<i>1</i>
<i>Chapitre 1</i>	<i>6</i>
Réseaux de neurones à base d'ondelettes dans la commande adaptative des systèmes SISO incertains.	6
1. Introduction.....	6
2. Réseau de neurones à base d'ondelettes.....	7
3. Architecture d'un réseau de neurones à base d'ondelettes.....	9
4. Types des familles d'ondelettes.....	10
4.1 Ondelette Chapeau Mexicain.....	10
4.2 Ondelette Morlet.....	10
4.3 Ondelette Gaussienne.....	11

5. Commande adaptative directe par les réseaux de neurones à base d'ondelettes des systèmes SISO incertains	11
6. Commande adaptative indirecte par les réseaux de neurones à base d'ondelettes des systèmes SISO incertains	24
7. Conclusion	36
Chapitre 2.....	38
Commande adaptative neuronale indirecte à base d'ondelettes des systèmes MIMO incertains	38
1. Introduction	38
2. Réseau de neurones MIMO à base d'ondelettes	39
3. Commande adaptative indirecte par réseaux de neurones MIMO à base d'ondelettes	41
4. Résultats de simulation	51
5. Conclusion	55
Chapitre 3.....	57
Commande adaptative floue indirecte à base d'ondelettes des systèmes MIMO incertains	57
1. Introduction	57
2. Système flou MIMO.....	59
3. Commande adaptative indirecte par systèmes flous MIMO:	62
4. Résultats de simulation	66
5. Conclusion	69
Chapitre 4.....	71
Commande adaptative indirecte par FS-WNN hybride des systèmes MIMO incertains.....	71
1. Introduction	71
2. Approximateur FS-WNN hybride.....	73
3. Commande adaptative par FS-WNN hybride :.....	75
4. Résultats de simulation	84
5. Conclusion	99

Chapitre 5.....	101
Commande adaptative indirecte par FS-WNN hybride-variable des systèmes MIMO incertains	101
1. Introduction	101
2. Commande adaptative indirecte par FS-WNN hybride-variable:.....	106
4. Résultats de simulation	122
5. Conclusion	133
Chapitre 6.....	134
Commande adaptative directe par VFS-WNN hybride-variable de l'attitude de quadrotor.....	134
1. Introduction	134
2. Dynamiques de quadrotor	137
3. Contrôle par VFS-WNN hybride-variable de l'attitude de quadrotor.....	141
4. Résultats de simulation	148
5. Conclusion	160
Conclusion générale	162
Annexe A.....	165
Annexe B.....	172
Biblio-webographie.....	174

Liste des nomenclatures

$\dot{x} = \frac{dx}{dt}$	la dérivée de la variable x par rapport au temps
$x^{(i)}$	$i^{\text{ème}}$ dérivée de x par rapport au temps
x	Vecteur état instantané
$\ \cdot \ $	Une norme sur R^n
$ \cdot $	Valeur absolue d'un nombre réel ou module d'un nombre complexe
x^T	Transposé de x
R^n	Espace vectoriel de dimension n construit sur le corps des réels
R	Ensemble des nombres réels
$diag = (x_1, \dots, x_n)$	Matrice diagonale de vecteur $[x_1, \dots, x_n]^T$
$x \cdot y$	Produit scalaire de deux variables
$x^T \cdot y$	Produit matriciel
$tr(A)$	Trace de la matrice A
$Sgn(\cdot)$	Fonction signe
e	Erreur de poursuite
s	Surface de glissement
t	Variable de temps
u	Signal de commande
V	Fonction du Lyapunov
V_i	Fonction local de Lyapunov
I	Matrice d'identité
ε	Erreur d'approximation des paramètres
$\bar{\varepsilon}$	Valeur maximale de l' erreur d'approximation des paramètres
α	Paramètre de modulation pour la fonction f
β	Paramètre de modulation pour la fonction G
f, g	Fonctions inconnues
\hat{f}	Approximation de la fonction f

\tilde{f}	Erreur d'approximation de la fonction f
x_i	Variable i de vecteur d'état x
y_d	Trajectoire désirée ou de référence
$\gamma, \Upsilon, \tau, \sigma$	Gains d'adaptation
θ_i	Poids de sortie du système flou
ω_i	Poids de sortie du réseau de neurones
$\text{Max}(u)$	Le maximum de la valeur absolue du signal de commande
$E(u)$	L'énergie d'un signal (u)
B	Repère lié au corps de quadrotor
E	Repère fixe
m	masse
l	longueur
\prod	Produit
u_{eq}	Loi de commande équivalente
u_r	Loi de commande robuste
u_p	Loi de commande proportionnelle
u_{PI}	Loi de commande proportionnelle intégrale
ν	Seuil de l'erreur filtrée pour commutation de la loi de commande u_r
d	perturbation
D, \mathfrak{D}	Borne supérieure de la perturbation

Liste des abréviations

<i>NN</i>	Réseau de neurones (Neural Network)
<i>WNN</i>	Réseau de neurones à base d'ondelettes (Wavelet Neural Network)
<i>FS</i>	Système flou (Fuzzy System)
<i>FS-WNN</i>	Approximateur Fuzzy System-Wavelet Neural Network hybride
<i>VFS-WNN</i>	Approximateur Fuzzy System-Wavelet Neural Network hybride- variable
<i>MSE</i>	La moyennes des erreurs au carré
<i>PID</i>	Contrôleur proportionnel intégral dérivateur
<i>UAV</i>	véhicule sans pilote (unmanned aerial vehicle)
<i>VTOL</i>	Décollage et atterrissage vertical (Vertical takeoff and landing)
<i>SMC</i>	Contrôle par mode glissant (sliding mode controller)
<i>SISO</i>	Une seule entrée une seule sortie (single input single output)
<i>MIMO</i>	Multi-entrées multi-sorties (multiple input multiple output)
<i>ANFIS</i>	Adaptive neuro fuzzy inference system
<i>FWNN</i>	Fuzzy wavelet neural network
<i>Mh</i>	Fonction chapeau mexicain (mexican hat)

Liste des tableaux

Tableau 1.1 : Comparaison des performances de poursuite des trois contrôleurs WNNs en période transitoire.....	22
Tableau 1.2 : Comparaison des performances de poursuite des trois contrôleurs WNNs en régime permanent.....	22
Tableau 1.3 : Classification des fonctions ondelettes sur la base de leurs performances dans les deux périodes transitoires et permanentes.....	23
Tableau 1.4 : Comparaison des performances de poursuite des trois contrôleurs WNNs indirectes en période transitoire.....	35
Tableau 1.5 : Comparaison des performances de poursuite des trois contrôleurs WNNs indirectes en régime permanent.....	35
Tableau 4.1 : Comparaison de MSE de l'erreur de poursuite des trois contrôleurs (WNN, FS, FS-WNN) en l'absence des perturbations et des variations paramétriques.....	89
Tableau 4.2 : Comparaison de la valeur maximale de signal de commande des trois contrôleurs (WNN, FS, FS-WNN) en l'absence des perturbations et des variations paramétriques.....	89
Tableau 4.3 : Comparaison des performances des trois contrôleurs (WNN, FS, FS-WNN) dans le cas de présence des perturbations externes et des variations paramétriques.....	92
Tableau 5.1 : Comparaison des performances de contrôleur FS-WNN hybride-variable avec le contrôleur FS-WNN dans le cas d'absence des perturbations externes et des variations paramétriques.....	128
Tableau 5.2 : Comparaison des performances de contrôleur FS-WNN hybride-variable avec le contrôleur FS-WNN dans le cas de présence des perturbations externes et des variations paramétriques.....	132

Liste des Figures

Figure 1.1 : Architecture d'un réseau de neurones à base d'ondelettes	9
Figure 1.2 : Schéma de l'ondelette Chapeau Mexicain.	10
Figure 1.3 : Schéma de l'ondelette Morlet.....	11
Figure 1.4 : Schéma de l'ondelette Gaussienne	11
Figure 1.5 : Schéma du contrôleur WNN adaptatif direct en boucle fermée	14
Figure 1.6 : Pendule inversé	15
Figure 1.7 : Trajectoires de l'état du système x_1 des trois WNNs et de la référence y_d (WNN Chapeau mexicain (Mh), WNN Morlet (M) et WNN Gaussienne (G)).....	16
Figure 1.8 : Trajectoires de l'état du système x_2 des trois WNNs et de la référence y_d (WNN Chapeau mexicain (Mh), WNN Morlet (M) et WNN Gaussienne (G)).....	17
Figure 1.9 : Trajectoires de l'erreur de poursuite $e(t)$ des trois WNNs (WNN Chapeau mexicain (Mh), WNN Morlet (M) et WNN Gaussienne (G)).	17
Figure 1.10 : Signal de commande $u(t)$ des trois WNNs (WNN Chapeau mexicain (Mh), WNN Morlet (M) et WNN Gaussienne (G)).	18
Figure 1.11 : Cas de 02 neurones cachés.....	19
Figure 1.12 : Cas de 04 neurones cachés.....	19
Figure 1.13 : Cas de 09 neurones cachés.....	20
Figure 1.14 : Cas de 16 neurones cachés.....	20
Figure 1.15 : Cas de 25 neurones cachés.....	21
Figure 1.16 : Architecture d'une boucle de commande adaptative indirecte en utilisant des approximateurs WNNs	26
Figure 1.17 : Trajectoires d'état du système x_1 des trois WNNs et de la référence y_d (WNN Chapeau mexicain (Mh), WNN Morlet (M) et WNN Gaussienne (G)).....	30
Figure 1.18 : Trajectoires d'état système x_2 de trois WNNs et de la référence y_d (WNN Chapeau mexicain (Mh), WNN Morlet (M) et WNN Gaussienne (G)).....	30
Figure 1.19 : Trajectoires de l'erreur de poursuite $e(t)$ des trois WNNs (WNN Chapeau mexicain (Mh), WNN Morlet (M) et WNN Gaussienne (G)).	31

Figure 1.20 : Signal de commande $u(t)$ des trois WNNs (WNN Chapeau mexicain (Mh), WNN Morlet (M) et WNN Gaussienne (G)).	31
Figure 1.21 : Cas de 02 neurones cachés.	32
Figure 1.22 : Cas de 04 neurones cachés.	32
Figure 1.23 : Cas de 09 neurones cachés.	33
Figure 1.24 : Cas de 16 neurones cachés.	33
Figure 1.25 : Cas de 25 neurones cachés.	34
Figure 2.1 : Réseau de neurones MIMO à base d'ondelettes.	39
Figure 2.2 : Architecture d'une boucle de commande adaptative indirecte en utilisant des approximateurs WNNs.	47
Figure 2.3 : Robot manipulateur.	51
Figure 2.4 : Performance de poursuite de y_1 d'une trajectoire désirée yd_1 dans le cas de contrôleur WNN.	53
Figure 2.5 : Performance de poursuite de y_2 d'une trajectoire désirée yd_2 dans le cas de contrôleur WNN.	53
Figure 2.6 : Signal de commande u_1 du contrôleur WNN.	53
Figure 2.7 : Signal de commande u_2 dans le cas du contrôleur WNN.	54
Figure 2.8 : Erreur de poursuite e_1 dans le cas du contrôleur WNN.	54
Figure 2.9 : Erreur de poursuite e_2 dans le cas du contrôleur WNN.	55
Figure 3.1 : Procédure de conception d'un système flou.	58
Figure 3.2 : Système flou MIMO.	60
Figure 3.3 : Architecture d'une boucle de commande adaptative indirecte en utilisant des approximateurs FSs.	63
Figure 3.4 : Performance de poursuite de y_1 d'une trajectoire désirée yd_1 dans le cas du contrôleur FS.	67
Figure 3.5 : Performance de poursuite de y_2 d'une trajectoire désirée yd_2 dans le cas de contrôleur FS.	67
Figure 3.6 : Signal de commande u_1 du contrôleur FS.	67
Figure 3.7 : Signal de commande u_2 dans le cas du contrôleur FS.	68
Figure 3.8 : Erreur de poursuite e_1 dans le cas du contrôleur FS.	68
Figure 3.9 : Erreur de poursuite e_2 dans le cas du contrôleur FS.	69
Figure 4.1 : Approximateur FWNN.	72
Figure 4.2 : Approximateur à base de NNs Multiples.	73

Figure 4.3 : Architecture de l'approximateur FS-WNN hybride dans une boucle de commande	74
Figure 4.4 : Performance de poursuite de $y1d$ d'une trajectoire désirée $yd1$ pour les trois contrôleurs (WNN, FS, FS-WNN) en absence des perturbations et des variations paramétriques.	86
Figure 4.5 : Performance de poursuite de $y2d$ d'une trajectoire désirée $yd2$ pour les trois contrôleurs (WNN, FS, FS-WNN) en absence des perturbations et des variations paramétriques.	86
Figure 4.6 : Signal de commande $u1$ pour les trois contrôleurs (WNN, FS, FS-WNN) en absence des perturbations et des variations paramétriques.	87
Figure 4.7 : Signal de commande $u2$ pour les trois contrôleurs (WNN, FS, FS-WNN) en absence des perturbations et des variations paramétriques.	87
Figure 4.8 : Erreur de poursuite $e1$ pour les trois contrôleurs (WNN, FS, FS-WNN) en absence des perturbations et des variations paramétriques.	88
Figure 4.9 : Erreur de poursuite $e2$ pour les trois contrôleurs (WNN, FS, FS-WNN) en absence des perturbations et des variations paramétriques.	88
Figure 4.10 : Performance de poursuite de $y1$ pour les trois contrôleurs (WNN, FS, FS-WNN) dans le cas de présence des perturbations et des variations paramétriques.	90
Figure 4.11 : Performance de poursuite de $y2$ pour les trois contrôleurs (WNN, FS, FS-WNN) dans le cas de présence des perturbations et des variations paramétriques	91
Figure 4.12 : Signal de commande $u1$ pour les trois contrôleurs (WNN, FS, FS-WNN) dans le cas de présence des perturbations et des variations paramétriques	91
Figure 4.13 : Signal de commande $u2$ pour les trois contrôleurs (WNN, FS, FS-WNN) dans le cas de présence des perturbations et des variations paramétriques	91
Figure 4.14 : Performance du contrôleur FS-WNN hybride pour $\alpha=0$ et $\beta=0$	94
Figure 4.15 : Performances du contrôleur FS-WNN hybride pour $\alpha=0.1I$ et $\beta=0.1I$	Error!
Bookmark not defined.	
Figure 4.16 : Performances du contrôleur FS-WNN hybride pour $\alpha=0.3I$ et $\beta=0.3I$	95
Figure 4.17 : Performances du contrôleur FS-WNN hybride pour $\alpha=0.5I$ et $\beta=0.5I$	95
Figure 4.18 : Performances du contrôleur FS-WNN hybride pour $\alpha=0.7I$ et $\beta=0.7I$	96
Figure 4.19 : Performances du contrôleur FS-WNN hybride pour $\alpha=0.9I$ et $\beta=0.9I$	96
Figure 4.20 : Performances du contrôleur FS-WNN hybride pour $\alpha=I$ et $\beta=I$	97
Figure 5.1 : Architecture de l'approximateur FS-WNN hybride-variable dans une boucle de commande.	104
Figure 5.2 : Performance de poursuite de $y1d$ d'une trajectoire désirée $yd1$ en absence des perturbations et des variations paramétriques.	124

Figure 5.3 : Performance de poursuite de y_2 d'une trajectoire désirée y_{d2} en absence des perturbations et des variations paramétriques.	124
Figure 5.4 : Signal de commande u_1 en absence des perturbations et des variations paramétriques.	125
Figure 5.5 : Signal de commande u_2 en absence des perturbations et des variations paramétriques.	125
Figure 5.6 : Erreur de poursuite e_1 en absence des perturbations et des variations paramétriques..	126
Figure 5.7 : Erreur de poursuite e_2 en absence des perturbations et des variations paramétriques..	126
Figure 5.8 : Trajectoire des éléments diagonaux de facteur de modulation α en fonction du temps en absence des perturbations et des variations paramétriques.	127
Figure 5.9 : Trajectoire des éléments diagonaux de facteur de modulation β en fonction du temps en absence des perturbations et des variations paramétriques.	127
Figure 5.10 : Performance de poursuite de y_1 d'une trajectoire désirée y_{d1} en présence des perturbations et des variations paramétriques	129
Figure 5.11 : Performance de poursuite de y_2 d'une trajectoire désirée y_{d2} en absence des perturbations et des variations paramétriques	129
Figure 5.12 : Signal de commande u_1 en présence des perturbations et des variations paramétriques	129
Figure 5.13 : Signal de commande u_2 en présence des perturbations et des variations paramétriques	130
Figure 5.14 : Erreur de poursuite e_1 en présence des perturbations et des variations paramétriques	130
Figure 5.15 : Erreur de poursuite e_2 en présence des perturbations et des variations paramétriques	131
Figure 5.16 : Trajectoire des éléments diagonaux de facteur de modulation α en fonction du temps en présence des perturbations et des variations paramétriques	131
Figure 5.17 : Trajectoire des éléments diagonaux de facteur de modulation β en fonction du temps en présence des perturbations et des variations paramétriques	132
Figure 6.1 : Configuration du quadrotor	138
Figure 6.2 : Architecture du modèle dynamique du quadrotor	141
Figure 6.3 : Architecture du contrôleur adaptatif direct à base du VFS-WNN hybride-variable.	143
Figure 6.4 : Architecture d'une boucle de contrôle d'un quadrotor par FS-WNN hybride-variable	146
Figure 6.5 : Performances de poursuite de roulis par le contrôleur FS-WNN hybride et le PID.....	150

Figure 6.6 : Performances de poursuite de tangage par le contrôleur FS-WNN hybride et le PID .. 150

Figure 6.7 : Performances de poursuite de lacet par le contrôleur FS-WNN hybride et le PID 150

Figure 6.8 : Erreur de poursuite de roulis par le contrôleur FS-WNN hybride et le PID 151

Figure 6.9 : Erreur de poursuite de tangage par le contrôleur FS-WNN hybride et le PID 151

Figure 6.10 : Erreur de poursuite de lacet par le contrôleur FS-WNN hybride et le PID 151

Figure 6.11: Signal de commande de roulis 152

Figure 6.12: Signal de commande de tangage..... 152

Figure 6.13 : Signal de commande de lacet..... 153

Figure 6.14: Forces de commande du contrôleur FS-WNN hybride et le contrôleur PID..... 154

Les performances de poursuite des roulis et tangage sont illustrées dans la Figure 6.15 et la Figure 6.16, respectivement, dans le cas du contrôleur PID et du contrôleur FS-WNN hybride-variable.

La Figure 6.15 montre les performances de poursuite du lacet dans les deux contrôleurs. 154

Figure 6.16 : Performances de poursuite de roulis par le contrôleur FS-WNN hybride et le contrôleur PID..... 155

Figure 6.17 : Performances de poursuite de tangage par le contrôleur FS-WNN hybride et le contrôleur PID 155

Figure 6.18: Performances de poursuite de roulis par le contrôleur FS-WNN hybride et le contrôleur PID..... 155

Figure 6.19: Forces de commande du contrôleur FS-WNN hybride et le contrôleur PID..... 156

Figure 6.20: Paramètres de modulation du contrôleur FS-WNN hybride pour le mouvement de roulis et tangage pour $\alpha(0) = 0.1$ 156

Figure 6.21: Performances de poursuite de roulis par le contrôleur FS-WNN hybride-variable et le PID pour $\alpha(0) = 0.1$ 157

Figure 6.22: Performances de poursuite de tangage par le contrôleur FS-WNN hybride-variable et le PID pour $\alpha(0) = 0.1$ 157

Figure 6.23 : Performances de poursuite de lacet par le contrôleur FS-WNN hybride-variable et le PID pour $\alpha(0) = 0.1$ 158

Figure 6.24: Forces de commande du contrôleur FS-WNN hybride-variable 159

Figure 6.25: Paramètres de modulation du contrôleur FS-WNN hybride pour le mouvement de roulis et tangage (en haut) et pour le lacet (en bas)..... 159

Figure 6.26: Schéma de commande de l'attitude de quadrotor dans le Toolbox Matlab Px4 Pixhawk 172

Figure 6.27: Schéma de contrôleur de l'attitude de quadrotor dans le Toolbox Matlab Px4 Pixhawk 173

Introduction Générale

Au cours des dernières décennies, le contrôle des systèmes non linéaires est devenu un domaine actif de recherche due au développement de nouvelles technologies combinant l'électronique, la mécanique, l'informatique et les méthodes de recherche opérationnelle [1]. Ce développement a offert plus de confort et a facilité les tâches quotidiennes à l'utilisateur. Par exemple, il existe un besoin important d'augmenter le degré d'autonomie pour la conduite assistée et le stationnement automatique dans les nouvelles véhicules pour assurer la sécurité routière et diminuer les accidents de la route, de nombreux projets sont en cours de réalisation dans ce domaine comme le Traffic Jam Assist de Volkswagen [2]. En raison de ce développement technologique, les processus industriels sont devenus de plus en plus complexes et sophistiqués. Des efforts considérables sont fournis pour développer des systèmes de contrôle qui peuvent assurer le bon fonctionnement et la sécurité de ces systèmes [3], [4].

La conception d'un système de contrôle pour un système dynamique s'appuie sur le modèle obtenu pour ce système [5]. Devant la complexité des systèmes rencontrés en pratique, il est difficile d'obtenir un modèle mathématique fiable et simple à exploiter qui décrit avec précision la dynamique du système réel à commander [6], [7]. Le modèle mathématique obtenu contient souvent des incertitudes de modélisation de différentes origines telles que: Les paramètres de modèle sont approximativement connus ou variables, les dynamiques négligées ou mal connues, la modifications des conditions de fonctionnement, l'approximations linéaires de comportements non linéaires, les erreurs d'estimation dans un modèle identifié à partir de données mesurées [8], [9]. La non-linéarité et l'incertitude du modèle affectent directement les performances réalisables du système de contrôle [10]. Lorsque le modèle du système est partiellement inconnu ou défini de manière inexacte, les approches traditionnelles de contrôle dont on cite : la commande linéarisante par bouclage, la commande par mode glissant, le backstepping, qui dépendent du modèle de système ont des performances limités et ne peuvent pas contourner tous les problèmes de commande [11]–[13].

La commande adaptative est une technique prometteuse de conception de contrôleurs performants pour les systèmes non linéaires et incertains [14]–[16]. La commande adaptative pour un système non linéaire et linéarisable par bouclage a suscité beaucoup d'intérêt parmi les concepteurs des systèmes de

contrôle depuis plusieurs décennies [17]–[20]. Cependant, le niveau de performance souhaité ou les problèmes de poursuite avec une région de fonctionnement large nécessitent que la non-linéarité soit directement considérée dans la conception du système de contrôle [21]. Pour résoudre ce problème, la technique de commande adaptative a été combinée avec des approximateurs intelligents des fonctions non linéaires tels que les réseaux de neurones artificiels et les systèmes flous[22]–[25]. Selon le théorème d'approximation universelle, ces approximateurs intelligents ont la capacité d'approximer n'importe quelle fonction non linéaire inconnue d'un système dynamique [26], [27]..

Les systèmes flous et les réseaux de neurones sont deux systèmes intelligents d'architectures différentes qui peuvent être utilisés comme une boîte noire d'entrées/sorties pour estimer des fonctions non linéaires et donc dans la synthèse des contrôleurs intelligents [28]–[30]. Considérant les différents avantages et inconvénients de chaque approximateur, la combinaison des deux approximateurs a été proposée dans de nombreuses schémas de contrôle. Han et al. [31] ont proposé un système neuro-flou coopératif qui a une phase de prétraitement où les mécanismes d'apprentissage du réseau de neurones définissent certains sous-blocs du système flou. Une fois ces sous-blocs flous calculés, les méthodes d'apprentissage du réseau de neurones sont supprimées et seul le système flou sera exécuté. Eboule et hasan [32] ont proposé un système neuro-flou concurrent, ce système se diffère de réseau coopératif en ce que le réseau de neurones et le système flou fonctionnent ensemble en permanence, les réseaux de neurones prétraitant les entrées (ou post-traitant les sorties) du système flou. La fusion des systèmes flous avec les réseaux de neurones a été proposée initialement par Jang et al. [33] qui ont proposé le Adaptive Neuro-Fuzzy Inference System (ANFIS). ANFIS constitue une fusion entre un système flou et des réseaux de neurones conventionnels dans un seul réseau intelligent [34]. De même, Hajian et styles [35] ont développé une autre approche de fusion entre un système flou avec des réseaux de neurones à base d'ondelettes. La fusion des systèmes flous avec des réseaux de neurones dans un seul réseau est une méthode efficace pour exploiter les avantages de chaque système intelligent. Cependant, les réseaux intelligents fusionnés souffrent du problème de minimum local [36]–[40]. Ce qui affecte négativement leurs performances dans l'identification des systèmes incertains et ainsi dans la synthèse des contrôleurs intelligents performants.

Pour résoudre le problème des minimums locaux et pour améliorer l'approximation des dynamiques des systèmes incertains, Zhang [41] a combiné un ensemble de réseaux de neurones en parallèle pour obtenir un approximateur plus fiable qu'un seul réseau de neurones. Dans ce travail, trois schémas de combinaison ont été proposés et comparés, à savoir, la moyenne des sorties des

NNs et la moyenne pondérée des sorties des NNs. Les résultats obtenus ont montré que les réseaux de neurones combinés sont plus performants et plus robustes qu'un seul réseau de neurones.

D'un autre côté, Perrone et al. [42] ont constaté suite à leur étude dans l'estimation des fonctions en utilisant plusieurs estimateurs que: (i) étant donné une population d'estimations d'une fonction, on construit une estimation hybride de cette fonction pour améliorer l'erreur quadratique. (ii) en prenant la moyenne dans l'espace fonctionnel au lieu de l'espace paramétrique, on évite le problème des minimums locaux. Puisque, le nombre de minimums locaux distincts dans l'espace fonctionnel est inférieur au nombre de minimums locaux dans l'espace des paramètres. Donc, la moyenne des sorties d'estimateurs qui tombent dans des minimums locaux et dont les performances sont médiocres dans différentes régions de l'espace fonctionnel permet de surmonter la difficulté des minimums locaux. (iii) les éléments de la population sont choisis de n'importe quelle structure qui tend à réduire l'erreur moyenne quadratique.

De point de vue mathématique, Cho [43] a prouvé mathématiquement que la moyenne de (m) réseaux de neurones réduit l'erreur quadratique MSE par un facteur de $(1/m)$, et a précisé que les réseaux combinés doivent avoir des structures différentes. Cho a expliqué dans son travail que la précision d'approximation se diffère d'une architecture à une autre, par conséquent, il n'est pas recommandable de prendre la moyenne simple des sorties de ces structures. Il est plus judicieux d'attribuer un poids de pondération à chaque structure. Selon Zhang [41], dans le calcul de la moyenne pondérée, un bon choix des facteurs de pondération est très important dans la modélisation. Le problème dans le design des estimateurs hybrides ou multi-estimateurs réside dans le choix de l'architecture des estimateurs, et comment combiner les résultats des différents estimateurs pour extraire les meilleurs résultats d'estimation.

Une structure multi-modèles à base des NNs pour l'identification non linéaire des dynamiques d'un avion sous différents types de défauts d'actionneur a été proposé par Emami et Banazadeh [44]. Dans cette structure, une hybridation non linéaire d'un ensemble de réseaux de neurones a été utilisée pour construire un approximateur hybride plus fiable qu'un seul réseau de neurones. Chaque réseau neuronal dans cet ensemble représente un modèle local. Le premier modèle local représente le fonctionnement nominal de l'avion (cas sans défauts). Ensuite, chaque modèle local est entraîné sur un défaut d'actionneur différent. La fonction de validité de chaque modèle local est calculée en prenant la gaussienne de l'erreur moyenne absolue normalisée (NMAE) de sa sortie. La sortie globale est calculée en prenant la somme de toutes les fonctions de validité normalisées multipliées par leurs sorties des modèles locaux correspondantes. D'autres techniques non linéaires

d'agrégation ont été proposées. Cependant, trouver une bonne fonction d'agrégation non linéaire est plus compliqué que trouver une combinaison linéaire [45], [46].

Une technique linéaire intéressante d'hybridation a été proposée par Dimitris et al. [50] pour la synthèse d'un filtre récursif discret. Les différents termes qui constituent l'expression du filtre ont été combinés à l'aide des facteurs de modulation (α , $(1-\alpha)$). Ces facteurs de modulation ont été utilisés pour contrôler le degré de contribution de chaque terme dans l'équation du filtre. Cette technique d'hybridation a l'avantage de permettre au concepteur de fixer le degré de contribution de chaque terme contrairement à la moyenne simple utilisée par Zhang et al. [41], Perrone et al. [47] et Cho [43]. Cette technique a été exploitée par Ghavidel et al. [48] qui a réalisé une hybridation entre un contrôleur PID et un contrôleur flou adaptatif en utilisant la technique de modulation. Plusieurs valeurs des facteurs de modulation ont été testé dans le but de déterminer la meilleure combinaison des valeurs d'hybridation. Bibi et al. [49] ont choisi de faire une hybridation entre un contrôleur flou adaptatif direct et un contrôleur flou adaptatif indirect en utilisant la technique de modulation pour le control des systèmes SISOs incertains. Les facteurs de modulation qui règlent le degré de contribution de chaque contrôleur ont été fixés par le concepteur à priori. Dans le même contexte, Huang et al. [50] ont proposé un contrôleur flou direct/indirect hybride décentralisé pour la commande des systèmes à grand échelle. Néanmoins, dans tous ces travaux d'hybridation en utilisant la technique de modulation, les paramètres de modulation sont sélectionnés à priori par le concepteur après plusieurs essais en utilisant l'erreur quadratique comme critère de sélection. Ce qui représente un grand handicap en temps de calcul, et ne garantit pas d'obtenir du régulateur hybride optimal.

Ces approches d'hybridation des systèmes intelligents nous ont motivé pour développer dans le cadre de cette thèse un nouveau système intelligent hybride qui combine les systèmes flous et les WNNs en utilisant la technique de modulation, pour la commande des système multi entrées multi sorties (MIMOs) non linéaire incertains. Ensuite, nous avons développé un algorithme d'ajustement en ligne des paramètres de modulation pour mieux contrôler le degré de contribution de chaque approximateur et atteindre des performances optimales du nouveau approximateur FS-WNN hybride. Pour mettre en évidence la supériorité du nouveau approximateur hybride et variable proposé dans le cadre de cette thèse, une étude comparative par simulation est effectuée sur un robot manipulateur MIMO et un quadrotor.

Cette thèse est composée de six chapitres, en plus d'une introduction générale, une conclusion générale et des perspectives envisageables, et elle est organisée comme suit:

Dans le premier chapitre nous présentons une étude détaillée de la commande adaptative directe et indirecte des systèmes SISO non linéaires incertains en utilisant les réseaux de neurones à base d'ondelettes. Nous étudions les critères de choix de la fonction d'activation et son influence sur la taille et les performances du réseau WNN [51]. Nous validons ce choix par des tests de simulation sur un pendule inversé.

Dans le deuxième chapitre, nous élargissons notre étude aux systèmes MIMO incertains d'ordre n . Pour ce faire, On augmente la taille du réseau WNN présenté dans le chapitre précédent pour qu'il soit adapté à l'identification des systèmes MIMO. On utilise ce réseau de neurones MIMO à base d'ondelettes pour approximer les dynamiques incertaines d'un système MIMO, les approximations obtenues serviront pour la synthèse de la loi de commande. Les performances de ce schéma de commande seront analysées par simulation sur un robot manipulateur.

Le troisième chapitre présente les fondements théoriques et les notions de base de la commande adaptative indirecte des systèmes MIMO incertains en utilisant les systèmes flous. Les systèmes flous sont utilisés pour approximer les dynamiques d'un système MIMO incertain dans ce chapitre, et une loi de commande est synthétisée à la base de cette approximation. Les performances du contrôleur flou adaptatif indirect sont par la suite évaluées par des tests de simulation sur un robot manipulateur pour établir des conclusions sur cette technique de contrôle.

Dans le quatrième chapitre nous développons un nouveau approximateur FS-WNN hybride [52], [53] qui combine le réseau de neurones à base d'ondelettes présenté dans le chapitre deux avec le système flou présenté dans le chapitre trois en utilisant la technique de modulation comme un moyen d'hybridation. À travers une étude comparative, nous évaluons les performances du nouveau approximateur FS-WNN hybride par rapport aux systèmes flous et les systèmes WNNs.

Dans le cinquième chapitre on envisage l'optimisation des performances du nouveau approximateur FS-WNN hybride à travers le développement d'un algorithme d'optimisation des paramètres de modulation. Les paramètres de modulation fixes seront substitués par des paramètres de modulation adaptatifs qui sont ajustés en ligne à l'aide de cet algorithme. Ce qui permet de surmonter le problème du choix des paramètres de modulation par le concepteur.

Dans le sixième chapitre, on présente la synthèse d'un nouveau schéma de commande basé sur l'approximateur FS-WNN hybride-variable proposé dans le cadre de cette thèse pour le contrôle de l'attitude de quadrotor. Ces derniers sont des systèmes non linéaires et d'une grande complexité car ils sont multi variables, couplés et naturellement instables. Ce qui permet d'examiner au mieux la viabilité du nouveau approximateur FS-WNN hybride-variable proposé dans cette thèse [54]

Chapitre 1

Réseaux de neurones à base d'ondelettes dans la commande adaptative des systèmes SISO incertains

1. Introduction

Au cours des années, de nombreux modèles intelligents ont été développés pour approximer les fonctions inconnues tels que les réseaux de neurones conventionnels et les réseaux de neurones à base d'ondelettes. Les réseaux de neurones à base d'ondelettes (Wavelet Neural Networks (WNNs)) sont une extension des réseaux de neurones conventionnels et sont proposés par Zhang et Benveniste en 1992[55]. Et depuis, les réseaux de neurones à base d'ondelettes ont attiré de plus en plus l'attention des chercheurs. Cet intérêt est dû au fait que ces réseaux regroupent la capacité des réseaux de neurones conventionnels dans l'apprentissage et les caractéristiques inhérentes des fonctions ondelettes dans la décomposition des signaux. Les réseaux de neurones à base d'ondelettes ne permettent pas seulement d'assurer la convergence de l'algorithme d'apprentissage mais d'augmenter également sa vitesse de convergence par rapport à un réseau de neurones conventionnel de même taille [56], [57].

Un réseau de neurones à base d'ondelette est un réseau de neurones qui possède des fonctions ondelettes dans les neurones cachés au lieu des fonctions Gaussiennes ou Sigmoides conventionnelles. Les fonctions ondelettes trouvent leur origine dans la théorie de traitement de signal depuis le début du vingtième siècle quand Alfred Haar a écrit sa dissertation intitulée « la théorie des systèmes des fonctions orthogonales » en 1909. L'idée de départ de Haar était de construire une transformation, pour la décomposition et la reconstruction des signaux, plus commode que la transformation de Fourier qui malheureusement fournit une analyse en fréquence mais ne permet pas la localisation temporelle des changements brusques, notamment pour des signaux de durée finie. Les recherches de Haar ont conduit au développement d'un ensemble de fonctions de base rectangulaires[58]. Plus tard, des progrès majeurs dans le domaine ont été attribués à Jean Morlet. Morlet cherchait à identifier l'existence d'huile sous la croûte terrestre en

envoyant des impulsions acoustiques dans le sol et en analysant les échos reçus. Lorsque Morlet a d'abord utilisé la Transformée de Fourier à court terme pour analyser les échos, il a trouvé que le fait de garder la largeur de la fonction fenêtre (Window function) fixe ne fonctionnait pas. Il a essayé par ailleurs de maintenir constante la fréquence de la fonction fenêtre tout en modifiant la largeur de la fonction fenêtre en étirant ou en pressant la forme de la fonction fenêtre. Les formes d'onde résultantes de largeurs variables ont été appelées par Morlet le Wavelet (ondelette) [59]-[60], et ceci a marqué le début de l'ère de la recherche d'ondelette. En 1984, Jean Morlet en collaboration avec Alex Grossmann ont démontré qu'un signal pouvait être transformé sous la forme d'une ondelette et ensuite retransformé dans sa forme originale sans aucune perte d'information [61].

Plusieurs fonctions ondelettes ont apparu depuis l'étude de Morlet notamment l'ondelette Gaussienne qui est la dérivée première de la fonction Gaussienne et l'ondelette Chapeau Mexicain qui est le négatif normalisé de la dérivée seconde de la fonction Gaussienne [62]. Les fonctions ondelettes possèdent des propriétés intéressantes pour l'approximation des fonctions non linéaires. En particulier, elles possèdent la propriété d'approximation universelle, ce qui suggère leur utilisation dans des modèles « boîte noire » pour la modélisation et la commande des systèmes non linéaires. Ce modèle "boîte noire" est construit essentiellement sur la base de mesures effectuées sur les entrées et les sorties du processus à modéliser. A partir des mesures disponibles, un apprentissage des paramètres est établi de manière à obtenir la meilleure précision possible avec le plus petit nombre possible de paramètres ajustables.

2. Réseau de neurones à base d'ondelettes

Un réseau de neurones à base d'ondelettes est un réseau de neurones à une seule couche cachée avec des fonctions d'activations ondelettes qui est capable d'estimer toute fonction non linéaire continue sur un ensemble compact [63]. Les WNNs ont une forte tolérance d'approximation par rapport aux réseaux de neurones conventionnels, ce qui leur confère de forts avantages dans l'estimations en ligne [64]. Contrairement aux fonctions sigmoïdales utilisés dans les réseaux de neurones conventionnels, les fonctions d'ondelettes sont spatialement localisées, de sorte que la capacité d'apprentissage de WNN est plus efficace que la fonction sigmoïdale conventionnelle du réseau neuronal pour l'identification des systèmes. Les algorithmes d'apprentissage pour WNN convergent généralement dans un plus petit nombre d'itérations que pour les réseaux neuronaux conventionnels [65]. Ces réseaux WNNs combinent la capacité

d'apprentissage des réseaux de neurones artificiels et la capacité de décomposition des ondelettes. Ainsi, les systèmes de contrôle basés sur les WNNs peuvent atteindre de meilleures performances de contrôle que les systèmes de contrôle basés sur NNs.

En général, les performances du réseau de neurones à base d'ondelettes dépendent de nombreux facteurs tels que la connectivité du réseau, le nombre des nœuds cachés, l'algorithme d'apprentissage et le choix de la famille d'ondelettes. Cependant, les sujets les plus traités dans la littérature sont liés à l'architecture et au problème de la sélection des paramètres du réseau à l'aide des algorithmes d'apprentissage, négligeant l'importance du choix appropriée des fonctions d'activation dans les performances du réseau [66]. Bien que le choix de la fonction d'activation influe d'une manière intensive les performances du réseau. Une sélection appropriée de la fonction d'activation est très importante pour obtenir des performances optimales du réseau [67]. Le chapeau mexicain, la Morlet et l'ondelette Gaussienne sont les fonctions d'activation ondelettes les plus utilisées par les chercheurs dans le domaine de la modélisation et de la commande des systèmes [68]. Cependant, il n'existe pas de méthode claire pour le choix de la fonction d'activation la plus pertinente parmi les différentes familles existantes dans la littérature [69].

Dans ce chapitre, nous présentons en détails l'architecture d'un réseau de neurones à base d'ondelettes, et les différentes familles d'ondelettes les plus utilisées dans la littérature de contrôle intelligent des systèmes. Ensuite, nous étudierons l'impact du choix de la famille d'ondelettes sur la taille et les performances du réseau neuronal dans des problèmes de commande adaptative directe et indirecte des systèmes incertains. Pour ce faire, nous synthétisons des contrôleurs intelligents neuronaux à la base de différentes familles d'ondelettes. Ces contrôleurs intelligents seront appliqués pour contrôler en ligne un système non linéaire incertain. Leurs performances à la fois dans les régimes transitoires et stationnaires seront comparées afin d'établir des conclusions sur les avantages de chaque fonction ondelette dans des problèmes d'identification et de contrôle des systèmes incertains. La famille d'ondelettes sélectionnée doit assurer un signal de contrôle lisse, une bonne poursuite et une erreur quadratique minimale de poursuite de la trajectoire désirée. Deux critères d'évaluation dans ce cas sont établis : La fonction d'erreur quadratique moyenne (*MSE*) et le maximum de la valeur absolue de l'effort de contrôle (*Max |u|*).

3. Architecture d'un réseau de neurones à base d'ondelettes

Le réseau de neurones à base d'ondelettes est un réseau neuronal à trois couches avec des ondelettes comme des fonctions d'activation dans la couche cachée, comme le montre la Figure 1.1.

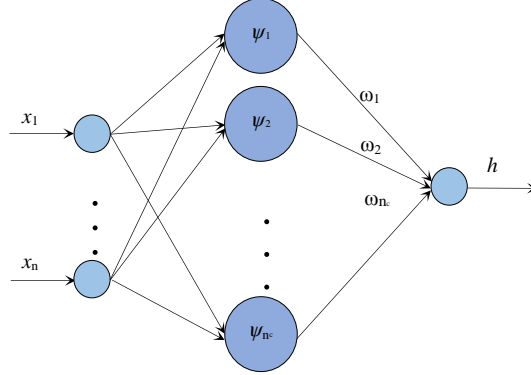


Figure 1.1 : Architecture d'un réseau de neurones à base d'ondelettes avec des entrées multiples et une seule sortie

La Figure 1.1 montre un réseau de neurones à base d'ondelettes multi entrées avec une seule sortie, où $\mathbf{x}^T = [x_1 x_2 \dots x_n] \in R^n$ est le vecteur d'entrée du WNN et n est le nombre d'entrées. h est la sortie du réseau de neurones à base d'ondelettes. La fonction ondelette ψ_{ij} pour l'entrée x_i et le nœud caché j peut être choisie comme une fonction Chapeau Mexicain, Morlet ou ondelette Gaussienne. Le produit des ondelettes ψ_{ij} pour $i = 1 \dots n$ construit l'ondelette multidimensionnelle $\psi_j(\mathbf{x})$, comme suit:

$$\psi_j(\mathbf{x}) = \prod_{i=1}^n \psi_{ij} \left(\frac{x_i - c_{ij}}{\sigma_{ij}} \right), \quad j = 1, \dots, n_c \quad (1.1)$$

Avec n_c est le nombre total des neurones cachés. $c \in R^{n \times n_c}$ et $\sigma \in R^{n \times n_c}$ sont les matrices de translation et de dilatation, respectivement, de l'ondelette ψ_j pour $j = 1, \dots, n_c$ par rapport à l'entrée x_i pour $i = 1 \dots n$. La sortie du réseau de neurones illustré dans la Figure 2.1 est donnée par [70]:

$$h(\mathbf{x}) = \sum_{j=1}^{n_c} \omega_j \psi_j(\mathbf{x}) = \mathbf{w}^T \Psi(\mathbf{x}) \quad (1.2)$$

avec $\Psi^T(\mathbf{x}) = [\psi_1 \psi_2 \dots \psi_{n_c}] \in R^{n_c}$ est le vecteur des fonctions d'activation de la couche cachée et $\mathbf{w}^T = [\omega_1 \omega_2 \dots \omega_{n_c}] \in R^{n_c}$ est un vecteur des poids ω_j qui connectent les nœuds cachés j pour $j = 1, \dots, n_c$ avec la sortie h . les poids ω_j sont initialisés arbitrairement dans l'intervalle $[-1,1]$ et sont adaptés en ligne.

4. Types des familles d'ondelettes

Trois Familles d'ondelettes sont étudiées dans ce chapitre, l'ondelette Chapeau Mexicain, la Morlet et l'ondelette Gaussienne:

4.1 Ondelette Chapeau Mexicain

L'ondelette Chapeau Mexicain (Figure 1.2) est la dérivée seconde de la fonction Gaussienne, elle est infiniment différentiable et symétrique. L'expression du chapeau mexicain [19] est donnée comme suit:

$$\psi(x) = (1 - x^2)e^{-\frac{x^2}{2}} \quad (1.3)$$

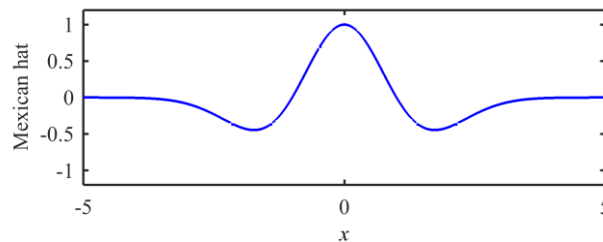


Figure 1.2 : Schéma de l'ondelette Chapeau Mexicain.

4.2 Ondelette Morlet

L'ondelette de Morlet (Figure 1.3) est infiniment différentiable et symétrique. L'expression de l'ondelette Morlet est donnée comme suit :

$$\psi(x) = \cos(5x)e^{-\frac{x^2}{2}} \quad (1.4)$$

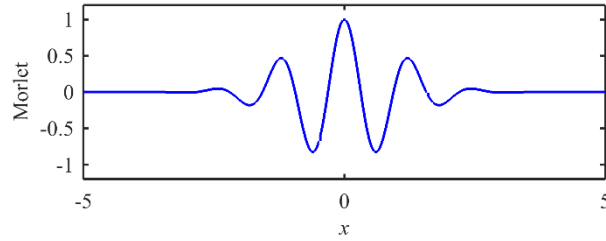


Figure 1.3 : Schéma de l'ondelette Morlet

4.3 Ondelette Gaussienne

L'ondelette Gaussienne (Figure 1.4) est la première dérivée de la fonction gaussienne, elle est infiniment différentiable, anti-symétrique. L'expression de l'ondelette gaussienne est donnée comme suit :

$$\psi(x) = -xe^{-\frac{x^2}{2}} \quad (1.5)$$

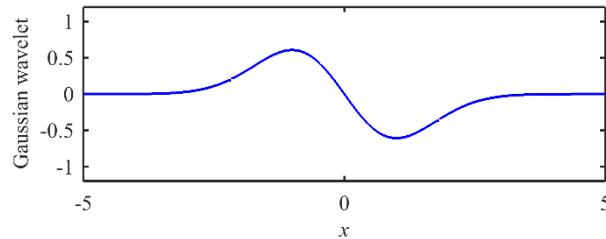


Figure 1.4 : Schéma de l'ondelette Gaussienne

5. Commande adaptative directe par les réseaux de neurones à base d'ondelettes des systèmes SISO incertains

Soit le système dynamique SISO suivant:

$$\begin{aligned} y^{(n)} &= f(\mathbf{x}) + g(\mathbf{x})u \\ y &= x \end{aligned} \quad (1.6)$$

où $\mathbf{x}^T = [x_1, x_2, \dots, x_n] = [x, \dot{x}, \dots, x^{(n-1)}]$ est le vecteur des variables d'état, $u \in \mathbb{R}$ est le signal de commande, $y \in \mathbb{R}$ est la sortie du système. $f(\mathbf{x}) \in \mathbb{R}$ et $g(\mathbf{x}) \in \mathbb{R}$ sont des fonctions non linéaires inconnues. La fonction du gain $g(\mathbf{x})$ est supposée bornée, définie positive et varie lentement dans le temps.

L'objectif de contrôle consiste à concevoir une loi de commande qui permet à la sortie du système de suivre une trajectoire bornée désirée $y_d \in R$.

On définit l'erreur de poursuite comme suit:

$$\mathbf{e} = [e_1, e_2, \dots, e_n] = [e, \dot{e}, \dots, e^{(n-1)}] \quad (1.7)$$

avec $e = y_d - y$.

on définit aussi l'erreur filtrée de poursuite $s \in R$ comme suit :

$$s(x) = \left(\frac{\partial}{\partial t} + k_c \right)^{n-1} \mathbf{e} = e^{(n-1)} + k_{c_{n-1}} e^{(n-2)} + \dots + k_{c_i} e = \mathbf{k}_c^T \mathbf{e} \quad (1.8)$$

$\mathbf{k}_c^T = [k_{c_1}, k_{c_2}, \dots, k_{c_{n-1}}, 1]$, $k_{c_i} \in R$ sont choisies des constantes positives telles que le polynôme $\tau(p) = p^{(n-1)} + k_{c_{n-1}} p^{(n-2)} + \dots + k_{c_i}$ est Hurwitz et p est l'opérateur de Laplace.

La dérivée par rapport au temps de l'erreur filtrée dans (1.8) est donnée par:

$$\dot{s} = y_d^{(n)} + \sum_{i=1}^{n-1} k_{c_i} e_i - f(\mathbf{x}) - g(\mathbf{x})u \quad (1.9)$$

Dans le cas où les fonctions non linéaires du système $f(\mathbf{x})$ et $g(\mathbf{x})$ sont connues avec précision, la loi de commande optimale u^* est synthétisée comme :

$$u^* = u_{eq}^* + u_{pr} \quad (1.10)$$

Avec :

$$\begin{cases} u_{eq}^* = \frac{1}{g(\mathbf{x})} \left(-f(\mathbf{x}) + y_d^{(n)} + \sum_{i=1}^{n-1} k_{c_i} e_i \right) \\ u_{pr} = ks \quad k > 0 \end{cases} \quad (1.11)$$

Où u_{eq}^* est le terme de contrôle équivalent idéal et u_{pr} est le contrôleur proportionnel.

Remplaçant (1.10) dans (1.9), donne $\dot{s}(t) = -gks(t)$, ce qui implique que $s(t) \rightarrow 0$ quand $t \rightarrow \infty$. Ainsi, $\mathbf{e} \rightarrow 0$. Cependant, les non-linéarités du système $f(\mathbf{x})$ et $g(\mathbf{x})$ sont inconnues. Ainsi, la loi de commande optimale u^* ne peut pas être implémentée. Pour résoudre ce problème, le terme de contrôle équivalent idéal u_{eq}^* est approximé à l'aide du réseau de neurones à base d'ondelette WNN décrit dans la section précédente.

Par conséquent, le terme de contrôle équivalent peut être exprimé sous la forme suivante:

$$u_{eq}^* = \mathbf{w}^{*T} \Psi(\mathbf{x}) + \varepsilon \quad (1.12)$$

où ε représente l'erreur d'approximation du terme de contrôle équivalent et elle est supposés bornée par $\bar{\varepsilon} > 0$, son expression est donnée par:

$$\varepsilon = u_{eq}^* - \mathbf{w}^{*T} \Psi(\mathbf{x}) \quad (1.13)$$

Le vecteur des paramètres optimaux \mathbf{w}^* vérifie:

$$\mathbf{w}^* = \arg \min_{\mathbf{w} \in \mathbb{R}^{n_c}} \left[\sup_{\mathbf{x} \in \mathbb{R}^{n_c}} |u_{eq}^* - u_{eq}(\mathbf{x}, \hat{\mathbf{w}})| \right] \quad (1.14)$$

On définit aussi $\tilde{\mathbf{w}} = \mathbf{w}^* - \hat{\mathbf{w}}$, comme \mathbf{w}^* est un vecteur de paramètres de poids idéaux, son estimation est utilisée pour concevoir la loi de commande équivalente (1.12) comme suit:

$$u_{eq} = \hat{\mathbf{w}}^T \Psi(\mathbf{x}) \quad (1.15)$$

La loi de commande globale est synthétisée comme suit:

$$u = u_{eq} + u_{Pr} + u_r \quad (1.16)$$

Ou bien :

$$u = \hat{\mathbf{w}}^T \Psi(\mathbf{x}) + ks + K \operatorname{sgn}(s) \quad (1.17)$$

Avec $u_r = K \operatorname{sgn}(s)$ est le terme de contrôle robuste qui compense l'effet des erreurs d'approximation et K est une constante positive. En additionnant et en soustrayant $g(\mathbf{x})u^*$ à (1.9), on obtient:

$$\dot{s} = y_d^{(n)} + \sum_{i=1}^{n-1} k_{c_i} e_i - f(\mathbf{x}) - g(\mathbf{x})u + g(\mathbf{x})u^* - g(\mathbf{x})u^* \quad (1.18)$$

En utilisant (1.11), on aura :

$$\dot{s} = g(\mathbf{x})(u^* - u) \quad (1.19)$$

En utilisant (1.12) et (1.15), on obtient :

$$\dot{s} = g(\mathbf{x})(\mathbf{w}^{*T} \Psi(\mathbf{x}) - \hat{\mathbf{w}}^T \Psi(\mathbf{x})) + g(\mathbf{x})\varepsilon - g(\mathbf{x})ks - g(\mathbf{x})K \operatorname{sgn}(s) \quad (1.20)$$

$$\dot{s} = g(\mathbf{x})\tilde{\mathbf{w}}^T \Psi(\mathbf{x}) + g(\mathbf{x})\varepsilon - g(\mathbf{x})ks - g(\mathbf{x})K \operatorname{sgn}(s) \quad (1.21)$$

Pour atteindre l'objectif de contrôle, il est nécessaire d'ajuster le vecteur des poids $\hat{\mathbf{w}}$ tel que $\mathbf{e}(t) \rightarrow 0$ quand $\hat{\mathbf{w}} \rightarrow \mathbf{w}^*$. Ainsi, la loi d'adaptation est choisie comme:

$$\dot{\hat{\mathbf{w}}} = \gamma s \Psi(\mathbf{x}) \quad (1.22)$$

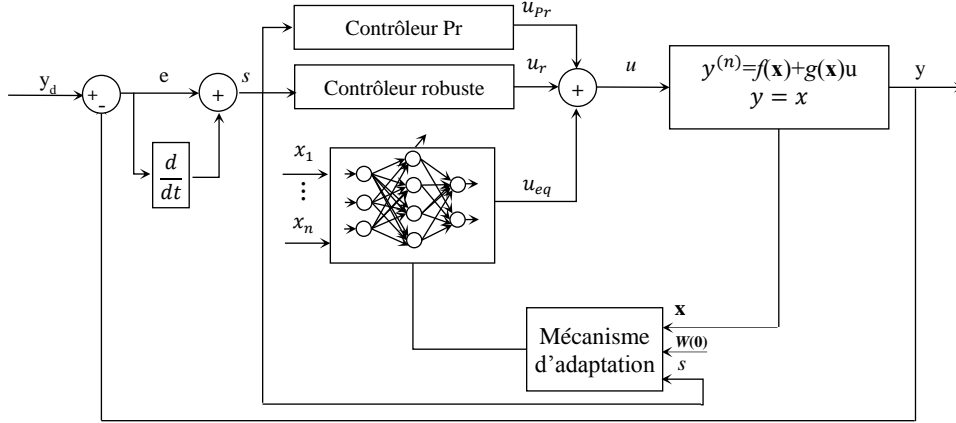


Figure 1.5 : Schéma du contrôleur WNN adaptatif direct en boucle fermée

Théorème:

Considérons le système en boucle fermée de la Figure 1.5, avec le système non linéaire (1.6). Le contrôleur adaptatif est choisi comme (1.16) et la loi d'adaptation des paramètres (1.22). Le contrôleur WNN adaptatif garantit les propriétés suivantes:

- (1) Le système en boucle fermée est stable.
- (2) $\mathbf{e}(t) \rightarrow 0$ quand $\mathbf{w} \rightarrow \mathbf{w}^*$

Preuve du Théorème :

Considérant la fonction candidate de Lyapunov (1.23) :

$$V = \frac{g^{-1}}{2} s^2 + \frac{1}{2\gamma} \tilde{\mathbf{w}}^T \tilde{\mathbf{w}} \quad (1.23)$$

La dérivée par rapport au temps de (1.23) est donnée comme suit :

$$\dot{V} = g^{-1} s \dot{s} + \frac{1}{\gamma} \tilde{\mathbf{w}}^T \dot{\tilde{\mathbf{w}}} \quad (1.24)$$

On remplace par (1.21) dans (1.24), on obtient :

$$\dot{V} = s(\tilde{\mathbf{w}}^T \Psi(\mathbf{x}) + \varepsilon - ks - K \operatorname{sgn}(s)) + \frac{1}{\gamma} \tilde{\mathbf{w}}^T \dot{\tilde{\mathbf{w}}} \quad (1.25)$$

On développe (1.25) comme suit :

$$\dot{V} = s\tilde{\mathbf{w}}^T \Psi(\mathbf{x}) + \frac{1}{\gamma} \tilde{\mathbf{w}}^T \dot{\tilde{\mathbf{w}}} + s\varepsilon - ks^2 - K s \operatorname{sgn}(s) \quad (1.26)$$

Sachant que $s \operatorname{sgn}(s)$ est équivalent à $|s|$, on peut écrire (1.26) sous la forme suivante :

$$\dot{V} = \frac{1}{\gamma} \tilde{\mathbf{w}}^T (\gamma s \Psi(\mathbf{x}) + \dot{\tilde{\mathbf{w}}}) + s\varepsilon - ks^2 - K |s| \quad (1.27)$$

Avec $\dot{\tilde{\mathbf{w}}} = -\hat{\dot{\mathbf{w}}}$, et en utilisant (23), on obtient :

$$\dot{V} = s\varepsilon - ks^2 - K |s| \quad (1.28)$$

$$\dot{V} \leq (\bar{\varepsilon} - K)|s| - ks^2 \quad (1.29)$$

Il suffit de choisir $K \geq \bar{\varepsilon}$ pour obtenir $\dot{V} \leq 0$, pour que le système soit stable.

Exemple de contrôle adaptative directe par les réseaux de neurones à base d'ondelettes

Dans cette section, le système de pendule inversé de la Figure 1.6 est utilisé pour évaluer les performances des contrôleurs synthétisés avec les fonctions d'activation Chapeau Mexican, Morlet et l'ondelette Gaussienne.

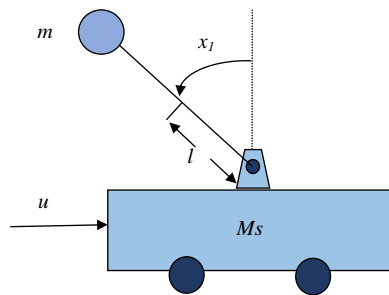


Figure 1.6 : Pendule inversé

Les équations d'état du pendule inversé sont données comme suit:

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = f(\mathbf{x}) + g(\mathbf{x})u \end{cases}$$

Avec :

$$f(x) = \frac{g_r \sin x_1 - \frac{mlx_2^2 \sin x_1 \cos x_1}{m+M}}{l\left(\frac{4}{3} - \frac{m \cos^2 x_1}{m+M}\right)}, \quad g(x) = \frac{\frac{\cos x_1}{m+M}}{l\left(\frac{4}{3} - \frac{m \cos^2 x_1}{m+M}\right)}$$

Tandis que $g_r = 9.81m/s^2$ est l'accélération due à la gravité, $l = 0.5m$ est la moitié de la longueur de la tige rigide, $M_s = 1.0kg$ est la masse du chariot et $m = 0.1kg$ la masse de la tige et u est le signal de commande. Soit $k_c = [1, 20]^T$ et $k = 3$ aussi $K = 0.001$. La trajectoire de référence est définie comme $y_d = 0.1 \sin(t)$. Et les conditions initiales sont prises comme $x_0 = [0, 0]^T$.

L'objectif de contrôle est que la sortie du système suive la trajectoire désirée y_d , selon le système en boucle fermée représenté sur la Figure 1.5. Trois WNNs avec deux entrées x_1, x_2 , une sortie u , et 16 nœuds cachés sont utilisés pour contrôler le système de pendule inversé. Il est à noter que les trois WNNs ont été conçus en utilisant les mêmes paramètres et que la famille des fonctions d'activation ondelettes qui est différente d'un réseau à un autre. Les paramètres de translation des fonctions d'activation ondelettes sont initialisés à partir de -0.08 avec un pas de 0.04 pour les deux variables d'état et les paramètres de dilatation sont fixés à 0.4. Les poids de sortie des réseaux WNNs sont initialisés arbitrairement dans l'intervalle $[-1, 1]$ et adaptés en ligne en utilisant les loi d'adaptations. Les résultats de simulation de la position du pendule inversé sont illustrés dans la Figure 1.7 et la vitesse angulaire du pendule sur la Figure 1.8. L'erreur de poursuite est présentée sur la Figure 1.9 et le signal de commande est présenté dans la Figure 1.10.

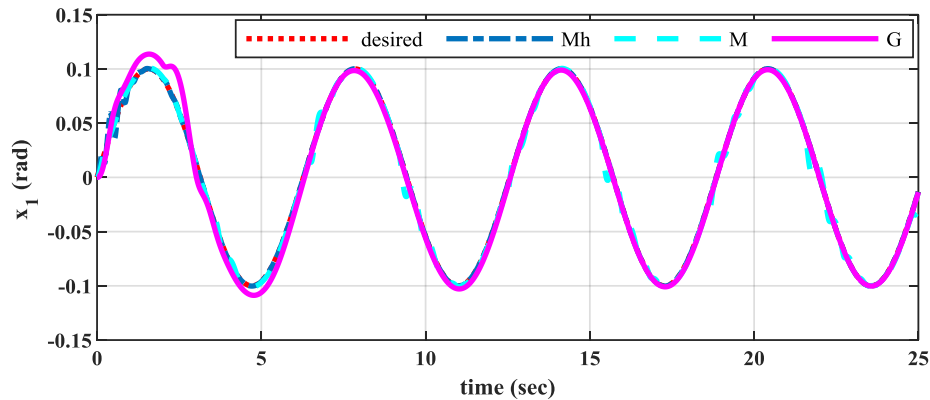


Figure 1.7 : Trajectoires de l'état du système x_1 des trois WNNs et de la référence y_d (WNN Chapeau mexicain (Mh), WNN Morlet (M) et WNN Gaussienne (G))

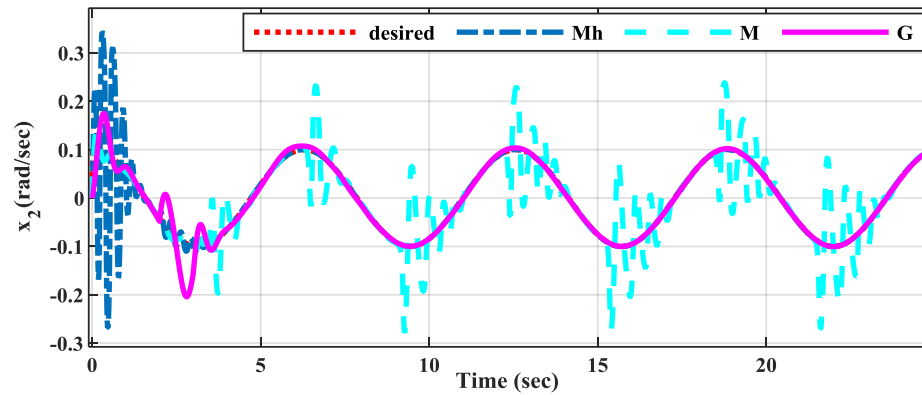


Figure 1.8 : Trajectoires de l'état du système x_2 des trois WNNs et de la référence \dot{y}_d (WNN Chapeau mexicain (Mh), WNN Morlet (M) et WNN Gaussienne (G)).

La Figure 1.7 et La Figure 1.8 montrent les performances de poursuite des états du système contrôlé par les trois réseaux de neurones à base d'ondelettes. Les trajectoires des états du système x_1, x_2 contrôlées par le contrôleur WNN à base de l'ondelette Chapeau Mexicain (MhWNN) et le contrôleur du réseau neuronal à ondelettes Gaussiennes (GWNN) sont satisfaisantes, avec une réponse transitoire plus lisse et plus lente dans le cas de GWNN. Cependant, le contrôleur réseau neuronal à base d'ondelettes Morlet (MWNN) a maintenu une erreur de poursuite statique.

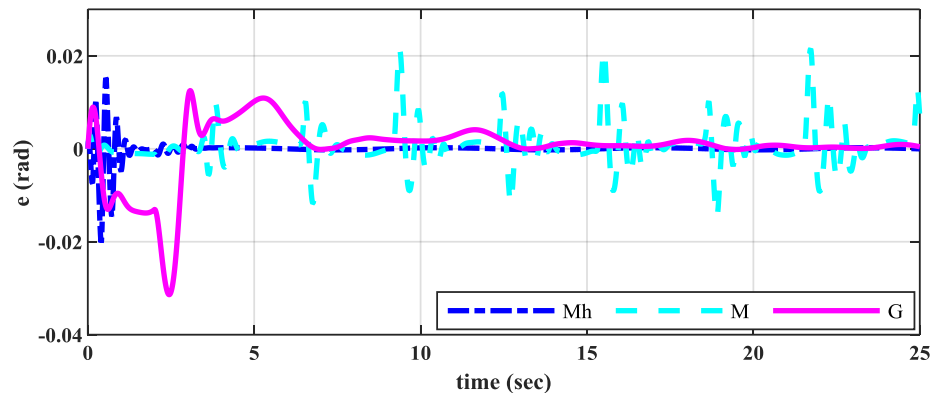


Figure 1.9 : Trajectoires de l'erreur de poursuite $e(t)$ des trois WNNs (WNN Chapeau mexicain (Mh), WNN Morlet (M) et WNN Gaussienne (G)).

En observant la Figure 1.9, nous constatons que l'erreur de poursuite dans le cas du contrôleur GWNN est meilleure que le contrôleur MWNN, même si l'erreur de poursuite du GWNN diminue lentement. De plus, les résultats de la Figure 1.9 montrent que l'erreur de poursuite obtenue dans le cas du contrôleur MhWNN diminue rapidement en comparaison avec les deux autres contrôleurs.

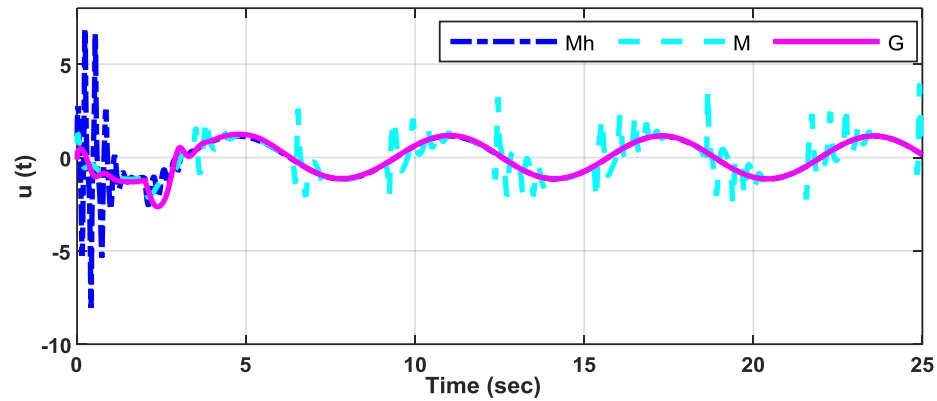
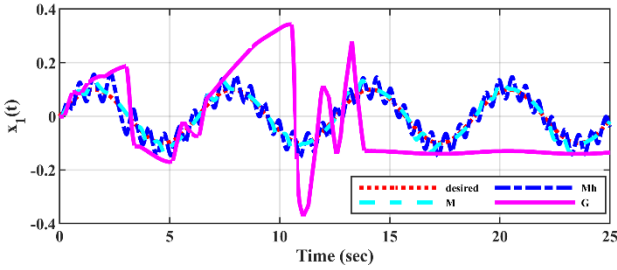


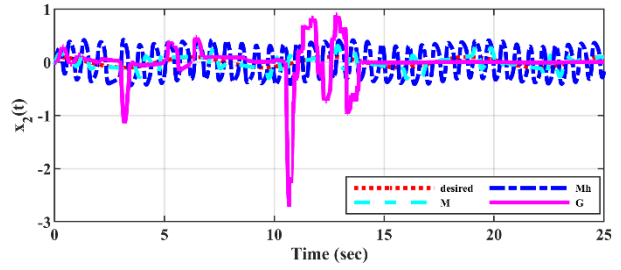
Figure 1.10 : Signal de commande $u(t)$ des trois WNNs (WNN Chapeau mexicain (Mh), WNN Morlet (M) et WNN Gaussienne (G)).

Les efforts de commande des trois contrôleurs WNNs sont donnés sur la Figure 1.10, où le signal de commande généré par le contrôleur MWNN est loin d'être lisse et présente des changements brusques sur toute la trajectoire.

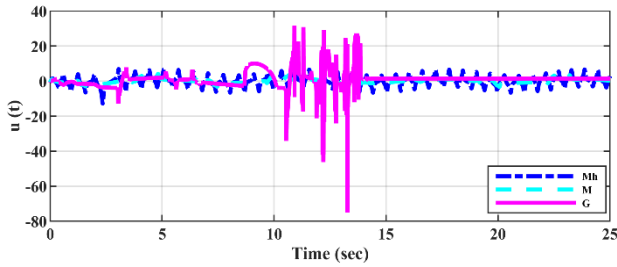
Pour observer l'influence du choix de la fonction d'activation sur la taille du réseau en terme de précision d'approximation, on établit une étude comparative entre les trois contrôleurs neuronaux à base d'ondelettes pour différentes tailles de réseau allant d'un réseau avec 02 neurones cachés jusqu'à un réseau avec 25 neurones cachés. Les résultats de simulations des performances de poursuite ainsi que le signal de commande et l'erreur de poursuite sont donnés dans les Figures 1.11 à 1.15. Les critères de comparaison sont la précision d'approximation à l'aide de la fonction MSE et la valeur maximale de la valeur absolue de l'effort de contrôle ($Max |u|$), à la fois en régime transitoire et en régime permanent. Pour observer l'influence de la fonction d'activation sur la taille du réseau, le nombre de nœuds cachés (n_c) est défini comme un facteur de comparaison de base, les résultats obtenus sont donnés dans le Tableau 1.1 pour le temps (0 sec à 06 sec) comme période transitoire et Tableau 1.2 pour le temps (06 sec à 25 sec) comme période permanente.



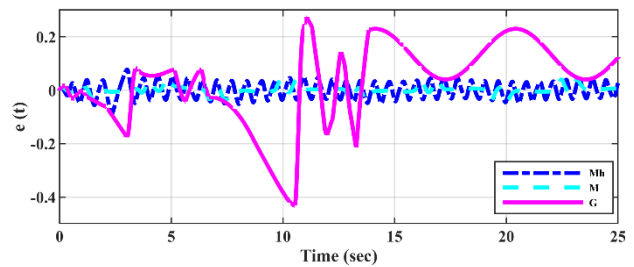
a) Trajectoires d'état système x_1 des trois WNNs et de la référence y_d dans le cas de 02 nœuds cachés



b) Trajectoires d'état système x_2 des trois WNNs et de la référence y_d dans le cas de 02 nœuds cachés

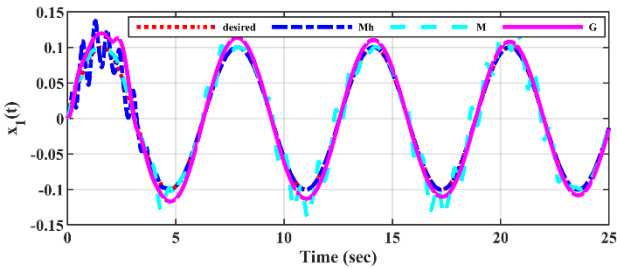


c) Signal de commande $u(t)$ des trois WNNs dans le cas de 02 nœuds cachés

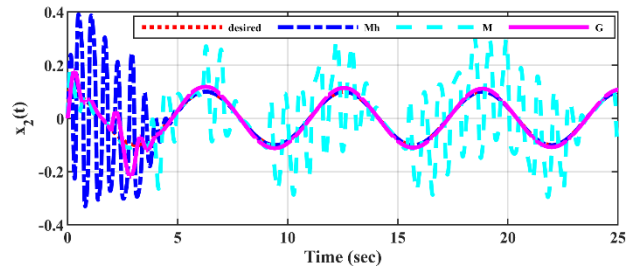


d) Erreur de poursuite $e(t)$ des trois WNNs dans le cas de 02 nœuds cachés

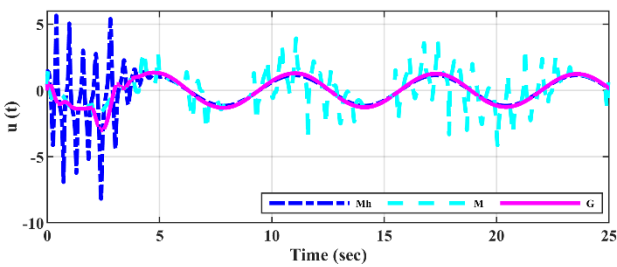
Figure 1.11 : Cas de 02 neurones cachés



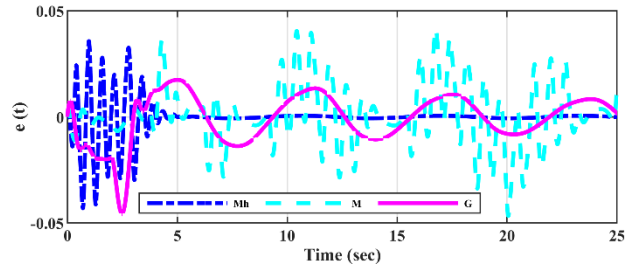
a) Trajectoires d'état système x_1 des trois WNNs et de la référence y_d dans le cas de 04 nœuds cachés



b) Trajectoires d'état système x_2 des trois WNNs et de la référence y_d dans le cas de 04 nœuds cachés

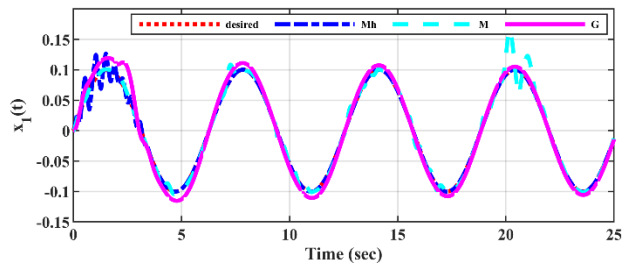


c) Signal de commande $u(t)$ des trois WNNs dans le cas de 04 nœuds cachés

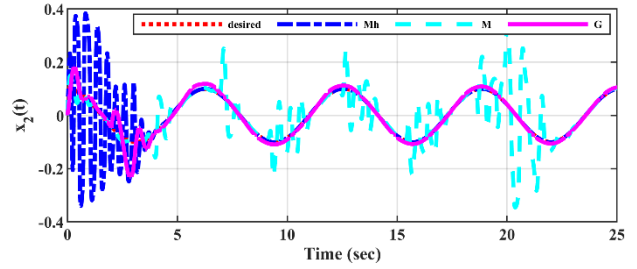


d) Erreur de poursuite $e(t)$ des trois WNNs dans le cas de 04 nœuds cachés

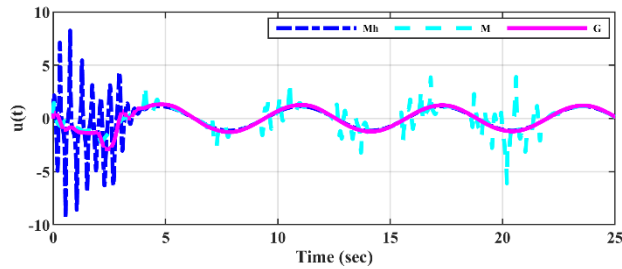
Figure 1.12 : Cas de 04 neurones cachés



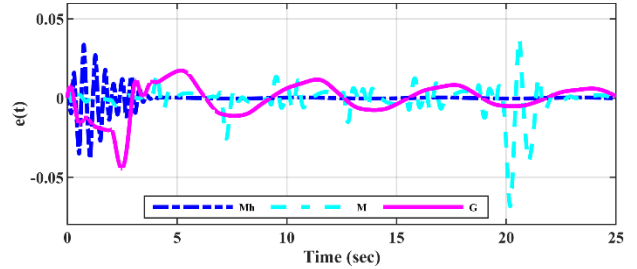
a) Trajectoires d'état système x_1 des trois WNNs et de la référence y_d dans le cas de 09 nœuds cachés



b) Trajectoires d'état système x_2 des trois WNNs et de la référence y_d dans le cas de 09 nœuds cachés

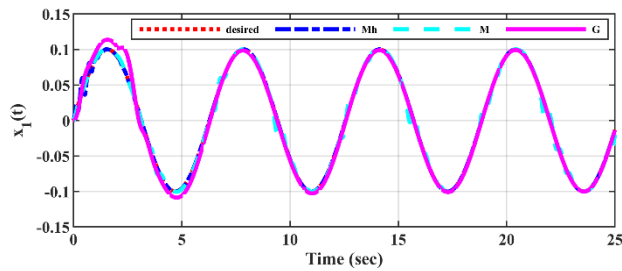


c) Signal de commande $u(t)$ des trois WNNs dans le cas de 09 nœuds cachés

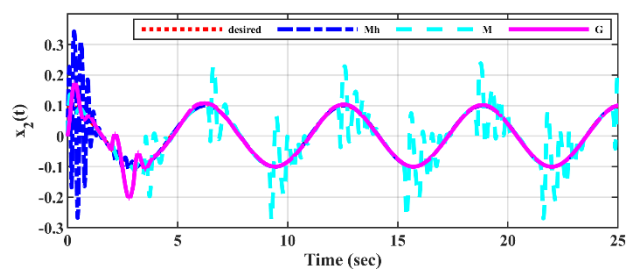


d) Erreur de poursuite $e(t)$ des trois WNNs dans le cas de 09 nœuds cachés

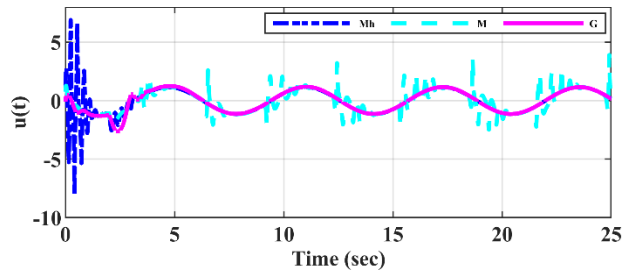
Figure 1.13 : Cas de 09 neurones cachés



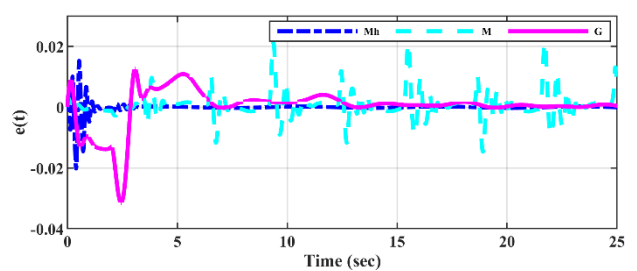
a) Trajectoires d'état système x_1 des trois WNNs et de la référence y_d dans le cas de 16 nœuds cachés



b) Trajectoires d'état système x_2 des trois WNNs et de la référence y_d dans le cas de 16 nœuds cachés



c) Signal de commande $u(t)$ des trois WNNs dans le cas de 16 nœuds cachés



d) Erreur de poursuite $e(t)$ des trois WNNs dans le cas de 16 nœuds cachés

Figure 1.14 : Cas de 16 neurones cachés

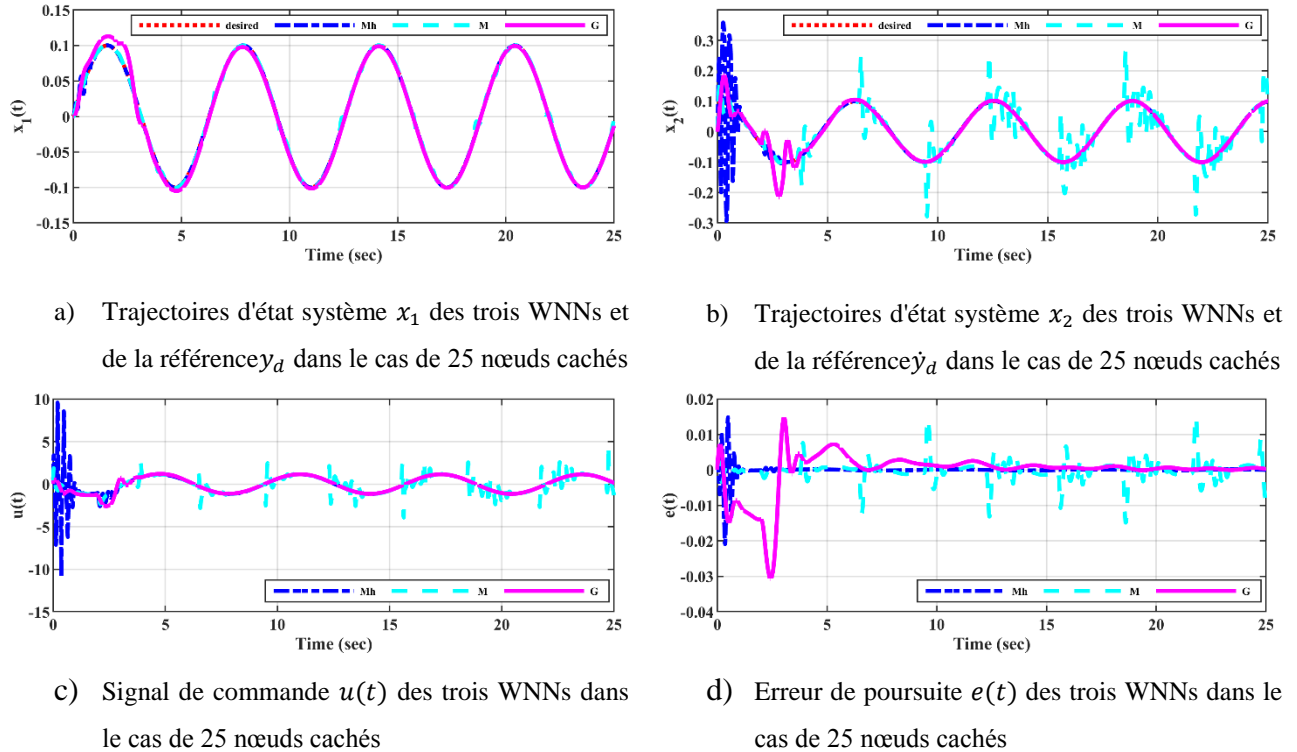


Figure 1.15 : Cas de 25 neurones cachés

La Figure 1.11 jusqu'à La Figure 1.15 montrent clairement la différence entre les performances des réseaux WNNs similaires mais avec des fonctions d'activation différentes, on remarque par exemple qu'un réseau WNN de 04 neurones cachés avec fonction activation Chapeau mexicain donne une meilleure poursuite qu'un WNN de 25 neurones cachés avec fonction d'activation Morlet.

Pour mieux observer cette différence, les performances des trois contrôleurs neuronaux à base d'ondelettes sont comparées en termes de précision de poursuite à l'aide de la fonction MSE et de la valeur maximale de l'effort de contrôle ($\text{Max } |u|$), à la fois en régime transitoire et en régime permanent. Pour observer l'influence de la fonction d'activation sur la taille du réseau, le nombre de nœuds cachés (n_c) est défini comme un facteur de comparaison de base, les résultats obtenus sont donnés dans le Tableau 1.1 pour le temps (0 sec à 06 sec) comme période transitoire et Tableau 1.2 pour le temps (06 sec à 25 sec) comme période permanente.

Tableau 1.1 : Comparaison des performances de poursuite des trois contrôleurs WNNs en période transitoire.

WNN	Chapeau Mexicain		Morlet		Ondelette Gaussienne	
	<i>MSE</i>	<i>MAX u </i>	<i>MSE</i>	<i>MAX u </i>	<i>MSE</i>	<i>MAX u </i>
2	8.294e-04	6.665	2.140e-04	3.790	9.676e-04	4.301
4	1.738e-04	5.647	5.667e-05	2.714	1.990e-04	1.358
9	1.105e-04	8.248	2.747e-05	2.684	1.914e-04	1.344
16	1.343e-05	6.906	6.388e-06	2.122	9.175e-05	1.271
25	1.115e-05	9.575	1.871e-06	2.395	7.118e-05	1.229

Tableau 1.2 : Comparaison des performances de poursuite des trois contrôleurs WNNs en régime permanent.

WNN	Chapeau Mexicain		Morlet		Ondelette Gaussienne	
	<i>MSE</i>	<i>MAX u </i>	<i>MSE</i>	<i>MAX u </i>	<i>MSE</i>	<i>MAX u </i>
2	7.248e-04	6.623	2.258e-04	3.360	0.114	10.493
4	1.189e-07	1.155	2.498e-04	3.966	7.118e-05	1.309
9	3.938e-08	1.153	1.243e-04	6.220	4.581e-05	1.283
16	2.672e-08	1.153	2.296e-05	3.988	2.412e-06	1.181
25	1.257e-08	1.152	1.007e-05	4.006	7.481e-07	1.165

Le Tableau 1.1 (période transitoire) et le Tableau 1.2 (période permanente) montrent les résultats suivants:

- i. L'ondelette chapeau mexicain: période transitoire oscillatoire, convergence rapide avec la plus petite erreur statique.
- ii. L'ondelette Morlet: période transitoire moins oscillatoire, convergence rapide avec l'erreur statique la plus élevée.
- iii. L'ondelette Gaussienne: période transitoire non oscillatoire, convergence lente avec une erreur statique plus petite.

Après étude des résultats des deux tableaux et selon le théorème d'approximation universel qui affirme que toute fonction non linéaire peut être approximée à n'importe quelle précision

souhaitée. On prend le cas de 16 neurones cachés (Tableau 1.2) qui affiche une précision à l'ordre de 10^{-8} avec une ondelette Chapeau mexicain en régime permanent. Le temps que les ondelettes Morlet et Gaussienne n'offrent pas une précision au-delà de 10^{-6} (Tableau 1.2) pour le même nombre de neurones cachés.

On remarque aussi qu'une ondelette Chapeau mexicain arrive à atteindre une précision de 10^{-8} en régime permanent même avec 09 neurones cachés seulement (Tableau 1.2). Cependant, dans le régime transitoire, on remarque que le signal de commande dans le cas de 16 neurones cachés est plus lisse et moins oscillatoire que dans le cas de 09 neurones cachés (Tableau 1.1).

Au-delà de 16 neurones cachés (25 et plus (Tableau 1.2)), l'amélioration au niveau de la précision n'est pas significative vis-à-vis la complexité du réseau (taille de réseau et nombre de variables à ajuster). Donc on se limite à 16 neurones cachés.

A La base de ces résultats et leur interprétation, la classification des fonctions ondelettes en fonction de leurs performances en périodes transitoires et permanentes est donnée dans le tableau 1.3 comme suit:

Tableau 1.3 : Classification des fonctions ondelettes sur la base de leurs performances dans les deux périodes transitoires et permanentes

	Chapeau Mexicain	Ondelette Gaussienne	Morlet
Periode transitoire	Oscillatoire	Non oscillatoire	Non oscillatoire
Periode Permanente	Erreur faible	Erreur faible	Erreur importante
Convergence	Rapide	lent	lent

Il est clair du Tableau 1.3 que le chapeau mexicain est la fonction d'activation ondelette la plus appropriée pour les objectifs de contrôle avec une caractéristique de convergence rapide en période transitoire et la plus petite erreur statique, tandis que l'ondelette Gaussienne arrive en deuxième position avec une convergence lente, et en dernière position, l'ondelette Morlet par une erreur permanente plus élevée.

6. Commande adaptative indirecte par les réseaux de neurones à base d'ondelettes des systèmes SISO incertains

On reconsidère le système dynamique SISO (1.6), où $f(\mathbf{x}) \in \mathbb{R}$ et $g(\mathbf{x}) \in \mathbb{R}$ sont des fonctions non linéaires inconnues. Notre objectif de contrôle consiste toujours à trouver une loi de commande qui permet à la sortie du système de suivre une trajectoire désirée $y_d \in \mathbb{R}$. La loi de commande idéale (1.10) ne peut être implémentée car $f(\mathbf{x})$ et $g(\mathbf{x})$ ne sont pas connues. Pour cela, Dans cette section, on va investiguer la commande adaptative indirecte des systèmes non linéaires incertains en utilisant les WNNs développés dans la section 3. En différence du travail présenté dans la section 3, les WNNs seront exploités pour approximer les dynamiques inconnues du système $f(\mathbf{x})$ et $g(\mathbf{x})$ comme suit :

$$\begin{aligned} f(\mathbf{x}) &= f_{WNN}^*(\mathbf{x}, \mathbf{W}_f^*) + \varepsilon_{f_{WNN}}(\mathbf{x}) \\ &= \mathbf{W}_f^{*T} \Psi_f(\mathbf{x}) + \varepsilon_{f_{WNN}}(\mathbf{x}) \end{aligned} \quad (1.30)$$

$$\begin{aligned} g(\mathbf{x}) &= g_{WNN}^*(\mathbf{x}, \mathbf{W}_g^*) + \varepsilon_{g_{WNN}}(\mathbf{x}) \\ &= \mathbf{W}_g^{*T} \Psi_g(\mathbf{x}) + \varepsilon_{g_{WNN}}(\mathbf{x}) \end{aligned} \quad (1.31)$$

Hypothèse 1:

Le vecteur des erreurs d'approximation $\varepsilon_{f_{WNN}}(\mathbf{x})$ et $\varepsilon_{g_{WNN}}(\mathbf{x})$ sont bornés par $\bar{\varepsilon}_{f_{WNN}}$ et $\bar{\varepsilon}_{g_{WNN}}$, respectivement, avec $\bar{\varepsilon}_{f_{WNN}}$ et $\bar{\varepsilon}_{g_{WNN}}$ des constantes positives inconnues.

Pratiquement, les paramètres optimaux du vecteur \mathbf{W}^* ne peuvent pas être déterminés, son estimation $\widehat{\mathbf{W}}$ est utilisée. Par conséquent, les fonctions inconnues du système sont approximées comme suit:

$$\hat{f}_{WNN}(\mathbf{x}, \widehat{\mathbf{W}}_f) = \widehat{\mathbf{W}}_f^T \Psi_f(\mathbf{x}) \quad (1.32)$$

$$\hat{g}_{WNN}(\mathbf{x}, \widehat{\mathbf{W}}_g) = \widehat{\mathbf{W}}_g^T \Psi_g(\mathbf{x}) \quad (1.33)$$

Avec :

$$\begin{aligned} f_{WNN}^*(\mathbf{x}, \mathbf{W}_f^*) &= \hat{f}_{WNN}(\mathbf{x}, \widehat{\mathbf{W}}_f) + \tilde{f}_{WNN}(\mathbf{x}, \widehat{\mathbf{W}}_f) \\ &= \widehat{\mathbf{W}}_f^T \Psi_f(\mathbf{x}) + \tilde{\mathbf{W}}_f^T \Psi_f(\mathbf{x}) \end{aligned} \quad (1.34)$$

$$\begin{aligned} g_{WNN}^*(\mathbf{x}, \mathbf{W}_g^*) &= \hat{g}_{WNN}(\mathbf{x}, \widehat{\mathbf{W}}_g) + \tilde{g}_{WNN}(\mathbf{x}, \widehat{\mathbf{W}}_g) \\ &= \widehat{\mathbf{W}}_g^T \Psi_g(\mathbf{x}) + \widetilde{\mathbf{W}}_g^T \Psi_g(\mathbf{x}) \end{aligned} \quad (1.35)$$

Et les erreurs d'approximation des paramètres sont données comme suit: $\widetilde{\mathbf{W}}_f = \mathbf{W}_f^* - \widehat{\mathbf{W}}_f$ et $\widetilde{\mathbf{W}}_g = \mathbf{W}_g^* - \widehat{\mathbf{W}}_g$.

Les fonctions inconnues du system $F(\mathbf{x})$ et $G(\mathbf{x})$ sont approximées en utilisant des réseaux de neurones WNNs, leurs approximations sont utilisées dans la synthèse de la loi de commande adaptative neuronale indirecte à base d'ondelettes comme suit:

On reprend l'équation de la dérivée de l'erreur filtrée par rapport au temps (1.9) et on rajoute et on soustraire $\hat{g}_{WNN}(\mathbf{x}, \widehat{\mathbf{W}}_g)u$, on aura:

$$\dot{s} = y_d^{(n)} + \sum_{i=1}^{n-1} k_{c_i} e_i - f(\mathbf{x}) - \left(g(\mathbf{x}) - \hat{g}_{WNN}(\mathbf{x}, \widehat{\mathbf{W}}_g) \right) u - \hat{g}_{WNN}(\mathbf{x}, \widehat{\mathbf{W}}_g) u$$

dans ce cas-là, où les fonctions non linéaires du système $f(\mathbf{x})$ et $g(\mathbf{x})$ sont approximées en utilisant le réseau WNN de la Figure 1.1,

On choisit la loi de commande comme (1.12), où u_{eq} dans ce cas est défini comme suit :

$$u = \hat{g}_{WNN}(\mathbf{x}, \widehat{\mathbf{W}}_g)^{-1} \left(y_d^{(n)} + \sum_{i=1}^{n-1} k_{c_i} e_i + K \operatorname{sgn}(s) - \hat{f}_{WNN}(\mathbf{x}, \widehat{\mathbf{W}}_f) \right) \quad (1.36)$$

Pour éviter le phénomène de chattering engendré par le terme glissant dans (2.21), une loi de commande par PI (proportionel-intégrateur) [32] est mise en œuvre quand les états du système est dans la couche cachée délimitée par ν , comme suit:

$$u_r = \begin{cases} k_1 s + k_2 \int s dt, & |s| < \nu \\ k \operatorname{sgn}(s), & |s| \geq \nu \end{cases} \quad (1.37)$$

En remplaçant (1.37) dans (1.35), on obtient:

$$\dot{s} = -[f(\mathbf{x}) - \hat{f}_{WNN}(\mathbf{x}, \widehat{\mathbf{W}}_f)] - [g(\mathbf{x}) - \hat{g}_{WNN}(\mathbf{x}, \widehat{\mathbf{W}}_g)]u - u_r - d \quad (1.38)$$

On remplaçant par (1.34)-(1.35) dans (1.38), on trouve :

$$\begin{aligned} \dot{s} &= -[f_{WNN}^*(\mathbf{x}, \widehat{\mathbf{W}}_f) + \varepsilon_{f_{WNN}}(\mathbf{x}) - \hat{f}_{WNN}(\mathbf{x}, \widehat{\mathbf{W}}_f)] \\ &\quad - [g_{WNN}^*(\mathbf{x}, \widehat{\mathbf{W}}_g) + \varepsilon_{g_{WNN}}(\mathbf{x}) - \hat{g}_{WNN}(\mathbf{x}, \widehat{\mathbf{W}}_g)]u - u_r - d \end{aligned} \quad (1.39)$$

Qui est équivalent à :

$$\begin{aligned} \dot{s} = & - \left[\left(f_{WNN}^*(\mathbf{x}, \widehat{\mathbf{W}}_f) - \hat{f}_{WNN}(\mathbf{x}, \widehat{\mathbf{W}}_f) \right) + \varepsilon_{f_{WNN}}(\mathbf{x}) \right] \\ & - \left[\left(g_{WNN}^*(\mathbf{x}, \widehat{\mathbf{W}}_g) - \hat{g}_{WNN}(\mathbf{x}, \widehat{\mathbf{W}}_g) \right) + \varepsilon_{g_{WNN}}(\mathbf{x}) \right] u - u_r - d \end{aligned} \quad (1.40)$$

Alors :

$$\dot{s} = - \left[\tilde{f}_{WNN}(\mathbf{x}, \widehat{\mathbf{W}}_f) + \varepsilon_{f_{WNN}}(\mathbf{x}) \right] - \left[\tilde{g}_{WNN}(\mathbf{x}, \widehat{\mathbf{W}}_g) + \varepsilon_{g_{WNN}}(\mathbf{x}) \right] u - u_r - d \quad (1.41)$$

Pour atteindre les objectifs de contrôle, il est nécessaire d'ajuster les paramètres des approximateurs WNNs pour avoir $e(t) \rightarrow 0$ lorsque $t \rightarrow \infty$. Pour cela, les lois d'adaptation des paramètres des réseaux WNNs sont données comme suit:

$$\dot{\widehat{\mathbf{W}}}_f = -\gamma_{W_f} \Psi_f(\mathbf{x}) s^T \quad (1.42)$$

$$\dot{\widehat{\mathbf{W}}}_g = -\gamma_{W_g} \Psi_g(\mathbf{x}) u s^T \quad (1.43)$$

Avec $\gamma_{W_f}, \gamma_{W_g}$ sont des gains positives d'adaptation.

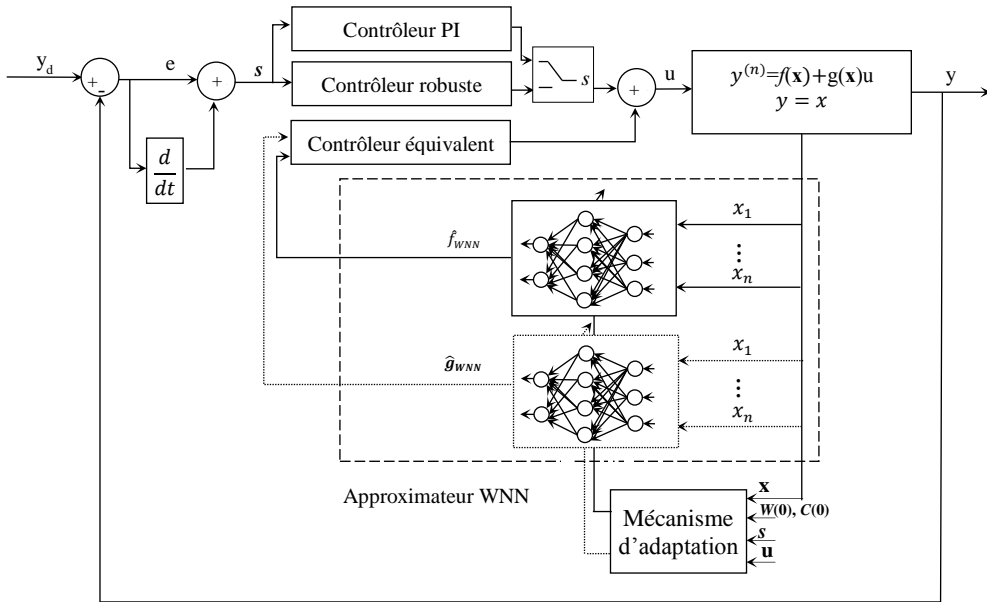


Figure 1.16 : Architecture d'une boucle de commande adaptative indirecte en utilisant des approximateurs WNNs

Théorème 1:

Considérant la boucle fermée de commande de la Figure 1.16, avec le système MIMO non linéaire incertain (1.6). La loi de commande neuronale adaptative à base d'ondelettes est choisie comme (1.37) et les lois d'adaptation des paramètres sont données à partir de (1.42)-(1.43). Le contrôleur à base des approximateurs WNNs adaptatifs garantit que le système de commande en boucle fermée est stable, et que tous les signaux sont bornés.

Preuve du Théorème 1:

Considérant une fonction candidate de Lyapunov comme suit :

$$V = \frac{1}{2} s^2 + V_{WNN} \quad (1.44)$$

Avec :

$$V_{WNN} = V_{fWNN} + V_{gWNN} \quad (1.45)$$

Et :

$$\begin{aligned} V_{fWNN} &= \frac{1}{2\gamma_{W_f}} \widetilde{W}_f^T \widetilde{W}_f \\ V_{gWNN} &= \frac{1}{2\gamma_{W_g}} \widetilde{W}_g^T \dot{\widetilde{W}}_g \end{aligned} \quad (1.46)$$

La dérivée par rapport au temps la fonction candidate de Lyapunov (1.44) est donnée par:

$$\dot{V} = s\dot{s} + \dot{V}_{WNN} \quad (1.47)$$

Avec :

$$\dot{V}_{WNN} = \dot{V}_{fWNN} + \dot{V}_{gWNN} \quad (1.48)$$

Et :

$$\begin{aligned} \dot{V}_{fWNN} &= \frac{1}{\gamma_{W_f}} \widetilde{W}_f^T \dot{\widetilde{W}}_f \\ \dot{V}_{gWNN} &= \frac{1}{\gamma_{W_g}} \widetilde{W}_g^T \dot{\widetilde{W}}_g \end{aligned} \quad (1.49)$$

Alors, en utilisant (1.42), on trouve :

$$\begin{aligned} \dot{V} = & -s^T([\tilde{f}_{WNN}(\mathbf{x}, \tilde{\mathbf{W}}_f) + \varepsilon_{f_{WNN}}(\mathbf{x})] + [\tilde{g}_{WNN}(\mathbf{x}, \tilde{\mathbf{W}}_g) + \varepsilon_{g_{WNN}}(\mathbf{x})]u + u_r + d) \\ & + \dot{V}_{WNN} \end{aligned} \quad (1.50)$$

Réorganisant (1.50) comme suit :

$$\begin{aligned} \dot{V} = & -(s^T \tilde{f}_{WNN}(\mathbf{x}, \tilde{\mathbf{W}}_f) + \dot{V}_{f_{WNN}}) - (s^T \tilde{g}_{WNN}(\mathbf{x}, \tilde{\mathbf{W}}_g)u + \dot{V}_{g_{WNN}}) \\ & - s^T(\varepsilon_{f_{WNN}}(\mathbf{x}) + \varepsilon_{g_{WNN}}(\mathbf{x})u + u_r + d) \end{aligned} \quad (1.51)$$

Alors, en utilisant (1.8)-(1.9), on aura :

$$\begin{aligned} \dot{V} = & -(s^T \tilde{\mathbf{W}}_f^T \Psi_f(\mathbf{x}) - \dot{V}_{f_{WNN}}) - (s^T \tilde{\mathbf{W}}_g^T \Psi_g(\mathbf{x})u - \dot{V}_{g_{WNN}}) \\ & - s^T(\varepsilon_{f_{WNN}}(\mathbf{x}) + \varepsilon_{g_{WNN}}(\mathbf{x})u + u_r + d) \end{aligned} \quad (1.52)$$

Aussi, en utilisant (1.23), on trouve :

$$\begin{aligned} \dot{V} = & -\left(s^T \tilde{\mathbf{W}}_f^T \Psi_f(\mathbf{x}) - \frac{1}{\gamma_{W_f}} \tilde{\mathbf{W}}_f^T \dot{\tilde{\mathbf{W}}}_f\right) - \left(s^T \tilde{\mathbf{W}}_g^T \Psi_g(\mathbf{x})u - \frac{1}{\gamma_{W_g}} \tilde{\mathbf{W}}_g^T \dot{\tilde{\mathbf{W}}}_g\right) \\ & - s^T(\varepsilon_{f_{WNN}}(\mathbf{x}) + \varepsilon_{g_{WNN}}(\mathbf{x})u + u_r + d) \end{aligned} \quad (1.53)$$

A la fin, on obtient:

$$\begin{aligned} \dot{V} = & \tilde{\mathbf{W}}_f^T \left(\dot{\tilde{\mathbf{W}}}_f - \gamma_{W_f} s^T \Psi_f(\mathbf{x})\right) + \tilde{\mathbf{W}}_g^T \left(\dot{\tilde{\mathbf{W}}}_g - \gamma_{W_g} s^T \Psi_g(\mathbf{x})u\right) \\ & - s^T(\varepsilon_{f_{WNN}}(\mathbf{x}) + \varepsilon_{g_{WNN}}(\mathbf{x})u + u_r + d) \end{aligned} \quad (1.54)$$

Avec $\dot{\tilde{\mathbf{W}}}_f = -\dot{\tilde{\mathbf{W}}}_f$, $\dot{\tilde{\mathbf{W}}}_g = -\dot{\tilde{\mathbf{W}}}_g$.

De même, en utilisant (1.16)-(1.17), on obtient :

$$\dot{V} = -s^T(\varepsilon_{f_{WNN}}(\mathbf{x}) + \varepsilon_{g_{WNN}}(\mathbf{x})u + u_r + d) \quad (1.55)$$

Et si on prend en considération l'hypothèse1:

$$\dot{V} \leq s^T(-k \operatorname{sgn}(s) + \mathfrak{D} + \bar{\varepsilon}_{f_{WNN}} + \bar{\varepsilon}_{g_{WNN}}|u|) \quad (1.56)$$

$$\dot{V} \leq \|s^T\|(-k + \xi_{WNN}) \quad (1.57)$$

Étant donné que k et $\xi_{WNN} = \mathfrak{D} + \bar{\varepsilon}_{f_{WNN}} + \bar{\varepsilon}_{g_{WNN}}|u|$ sont sélectionnés par le choix du concepteur, k doit être choisi de telle sorte que $\dot{V} < 0$. Par conséquent, le système en boucle fermée est stable. Ainsi, tous les signaux sont bornés ce qui implique que l'erreur de poursuite \mathbf{e} et tous ses dérivés sont bornés, également, \mathbf{s} est bornée, c'est-à-dire, $\mathbf{s} \in L_\infty$.

L'intégration des deux côtés de (1.57) donne:

$$\int_0^t \|s\| d\tau \leq \frac{1}{k - \xi_{WNN}} (|V(0)| + |V(t)|) \quad (1.58)$$

Ce qui signifie que $s \in L_1$, en se basant sur le Lemme de Barbalat [95], on trouve que: $\lim_{t \rightarrow \infty} \|s(t)\| = 0$, donc $\lim_{t \rightarrow \infty} \|e(t)\| = 0$. Ainsi, l'erreur de poursuite convergera vers zéro quand $t \rightarrow \infty$.

Exemple de contrôle adaptatif indirect par les réseaux de neurones à base d'ondelettes

On reprend le modèle du pendule inversé de la Figure 1.6. Les conditions initiales du pendule sont également choisies comme $x_0 = [0, 0]^T$. Dans cette partie, Deux WNNs MISO sont synthétisés pour approximer les fonctions inconnues $f(\mathbf{x})$ et $g(\mathbf{x})$ du système. Leurs approximations sont utilisées dans la synthèse de la loi de commande du contrôleur neuronal adaptatif indirect à base d'ondelettes donnée par (1.36). Les paramètres de design de la loi de commande (1.36) sont sélectionnés comme dans le cas du contrôleur WNN Soit $k_c = [1, 20]^T$ et $k = 3$ aussi $K = 0.001$. Les gains d'adaptation des réseaux WNNs sont choisis comme : $\gamma_{w_f} = -0.1, \gamma_{w_g} = -0.1$. Chaque système WNN à 02 entrées, une sorties et 81 neurones cachés. Les paramètres de translation des fonctions d'activation ondelettes sont initialisés à partir de -0.08 avec un pas de 0.04 pour les deux variables d'état et les paramètres de dilatation sont fixés à 0.4. Les poids de sortie des réseaux WNNs sont initialisés arbitrairement dans l'intervalle $[-1, 1]$ et adaptés en ligne en utilisant les loi d'adaptations pour assurer la poursuite de la trajectoires désirées $y_d = 0.1 \sin(t)$.

L'objectif de contrôle est que la sortie du système suive la trajectoire désirée y_d , selon le système en boucle fermée représenté sur la Figure 1.5. Trois contrôleurs adaptatifs indirects sont utilisés pour contrôler le système de pendule inversé. Chaque contrôleur adaptatif indirect utilise des réseaux WNNs avec une famille des fonctions d'activation différente, soit : Chapeau mexicain, Morlet ou ondelette Gaussienne, pour l'approximation des dynamiques du système. Il est à noter que les trois WNNs ont été conçus en utilisant les mêmes paramètres. Les résultats de simulation de la position du pendule sont illustrés sur la Figure 1.17 et la vitesse angulaire du pendule sur la

Figure 1.18. L'erreur de poursuite est présentée sur la Figure 1.19 et le signal de commande est présenté dans la Figure 1.20.

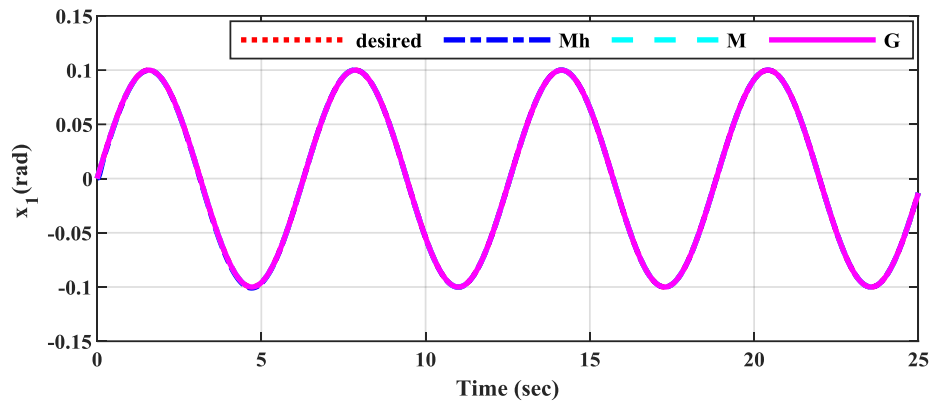


Figure 1.17 : Trajectoires d'état du système x_1 des trois WNNs et de la référence y_d (WNN Chapeau mexicain (Mh), WNN Morlet (M) et WNN Gaussienne (G))

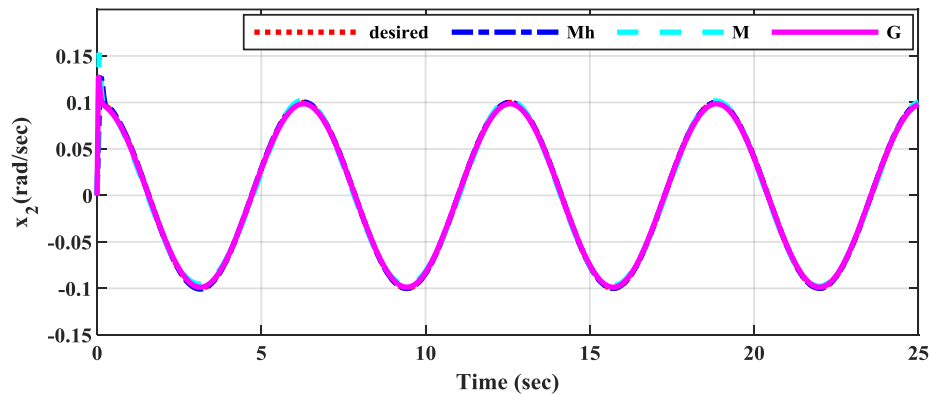


Figure 1.18 : Trajectoires d'état système x_2 de trois WNNs et de la référence \dot{y}_d (WNN Chapeau mexicain (Mh), WNN Morlet (M) et WNN Gaussienne (G)).

La Figure 1.117 et la Figure 1.18 montrent les performances de poursuite des états du système contrôlé par les trois contrôleurs adaptatifs neuronaux à base d'ondelettes. Les trajectoires des états du système x_1, x_2 contrôlées par le contrôleur neuronal adaptatif indirect WNN à base de l'ondelette chapeau mexicain (MhWNN) et le contrôleur neuronal à ondelettes gaussiennes (GWNN) ainsi que celui à base d'ondelettes (MWNN) sont satisfaisantes, avec une erreur statique un peu plus importante dans le cas du contrôleur neuronal à base d'ondelettes Morlet (MWNN).

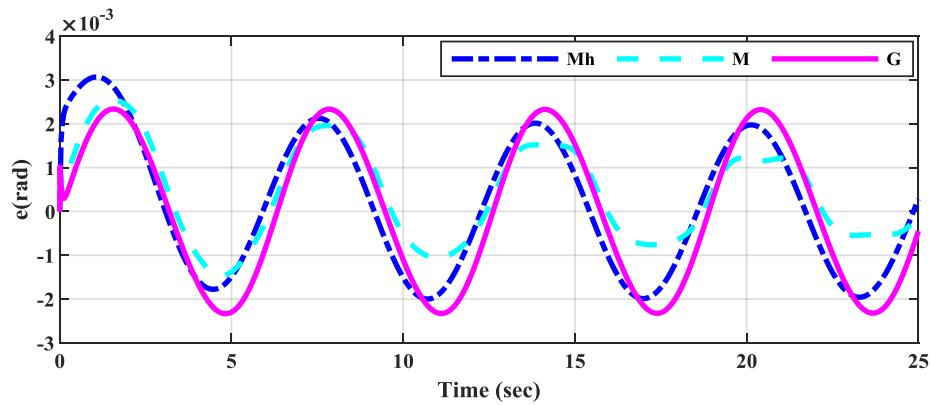


Figure 1.19 : Trajectoires de l'erreur de poursuite $e(t)$ des trois WNNs (WNN Chapeau mexicain (Mh), WNN Morlet (M) et WNN Gaussienne (G)).

En observant la Figure 1.19, nous constatons que l'erreur de poursuite dans le cas du contrôleur GWNN est meilleure que le contrôleur MWNN. De plus, les résultats de la Figure 1.19 montrent que l'erreur de poursuite obtenue dans le cas du contrôleur MhWNN converge rapidement vers 0 en comparaison avec les deux autres contrôleurs.

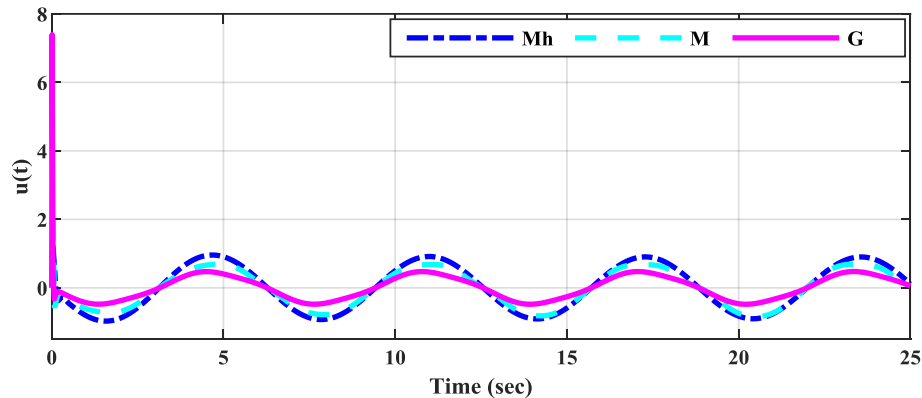


Figure 1.20 : Signal de commande $u(t)$ des trois WNNs (WNN Chapeau mexicain (Mh), WNN Morlet (M) et WNN Gaussienne (G)).

Les efforts de commande des trois contrôleurs WNNs sont donnés sur la Figure 2.10, où le signal de commande généré par les trois contrôleurs sont lisses avec un pic important dans la phase transitoire de contrôleur adaptatif indirect GWNN.

Pour valider les résultats de l'étude comparative qu'on a effectuée sur l'influence du choix de la fonction d'activation sur la taille du contrôleur adaptatif direct des systèmes incertains, nous établissons une deuxième étude comparative entre les trois familles d'ondelettes dans le contrôle adaptatif indirect des systèmes incertains. Nous construirons des réseaux de neurones de différentes

tailles allant d'un réseau avec 02 neurones cachés jusqu'à un réseau avec 25 neurones cachés. Les résultats de simulations sont donnés dans les Figures 1.21 à 1.25.

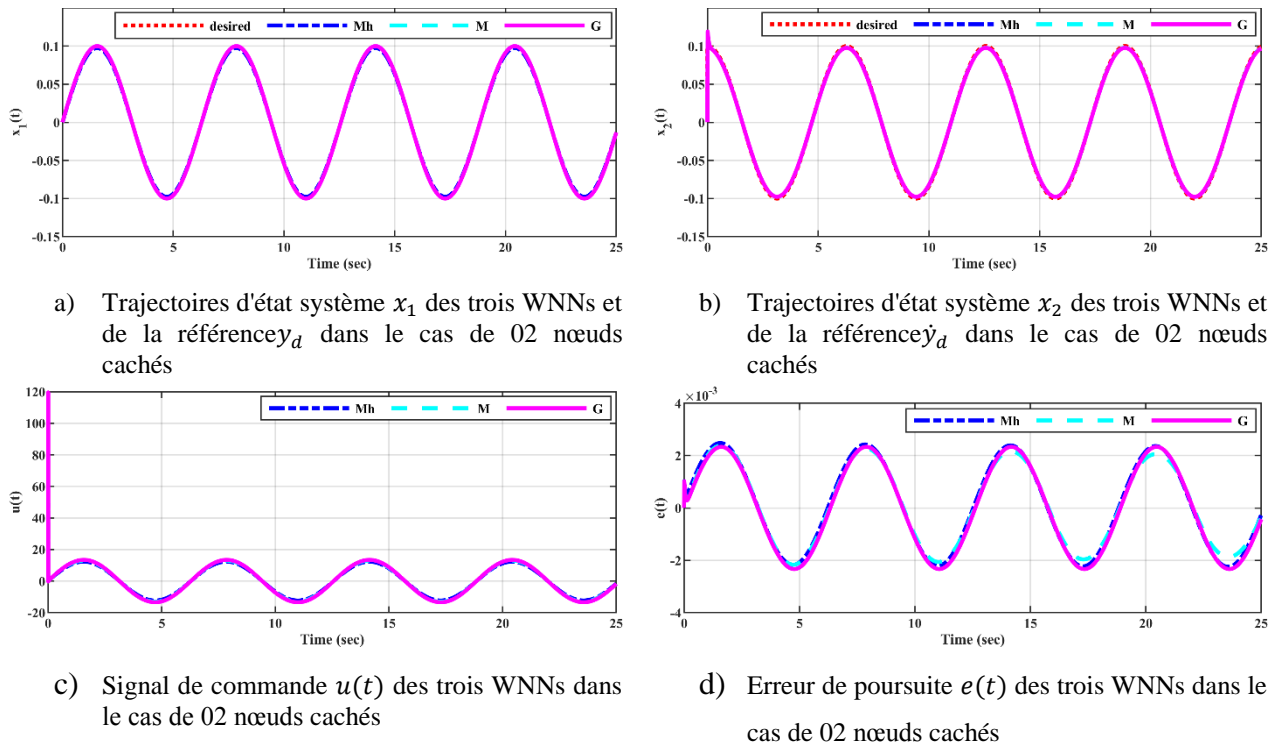


Figure 1.21 : Cas de 02 neurones cachés

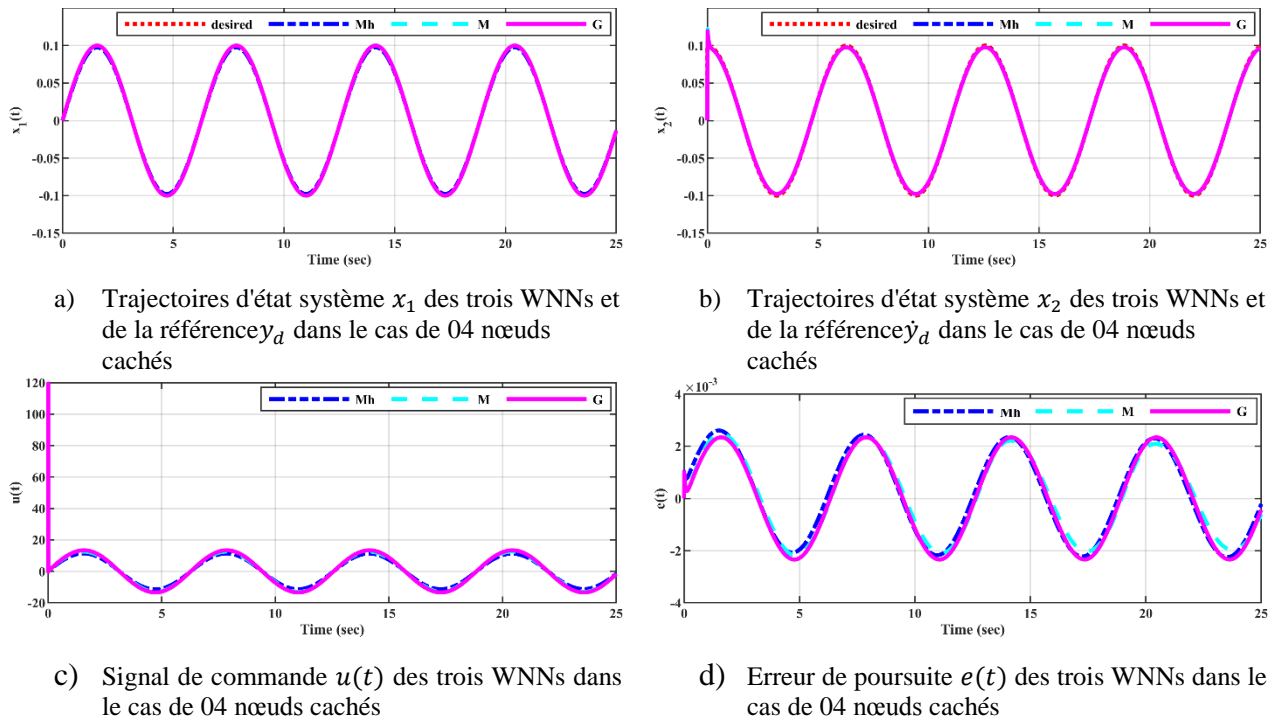


Figure 1.22 : Cas de 04 neurones cachés

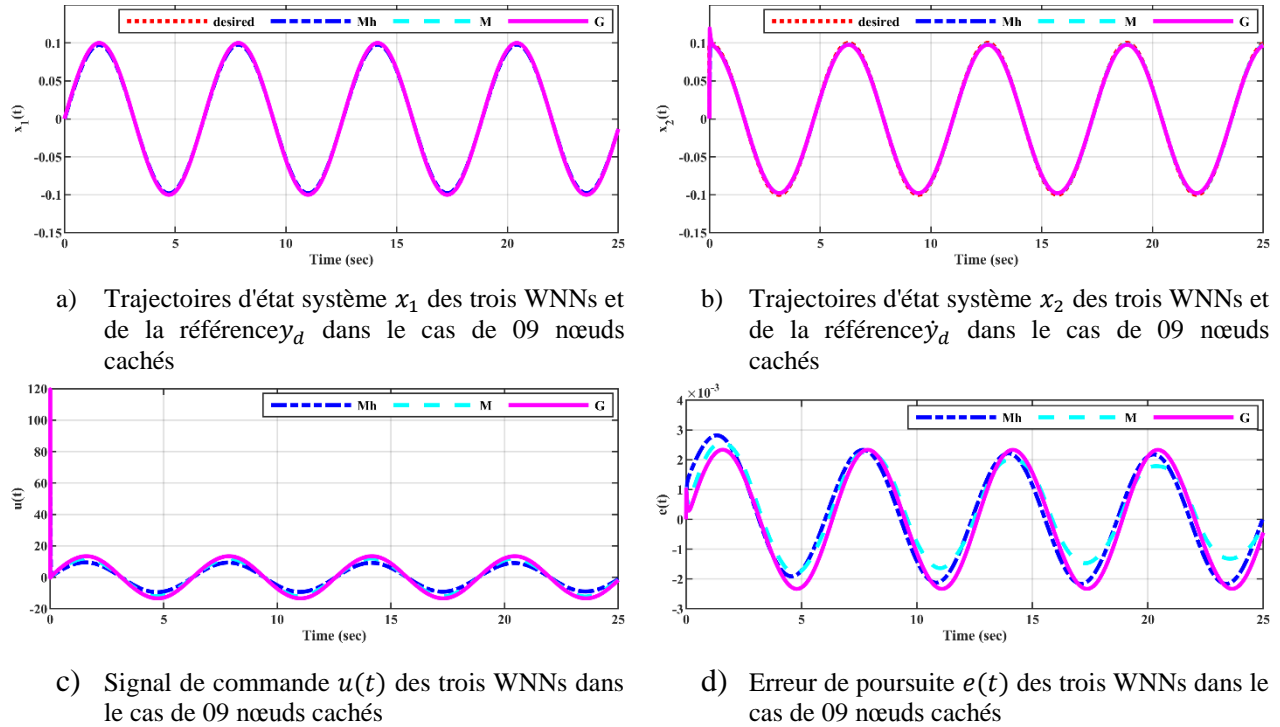


Figure 1.23 : Cas de 09 neurones cachés

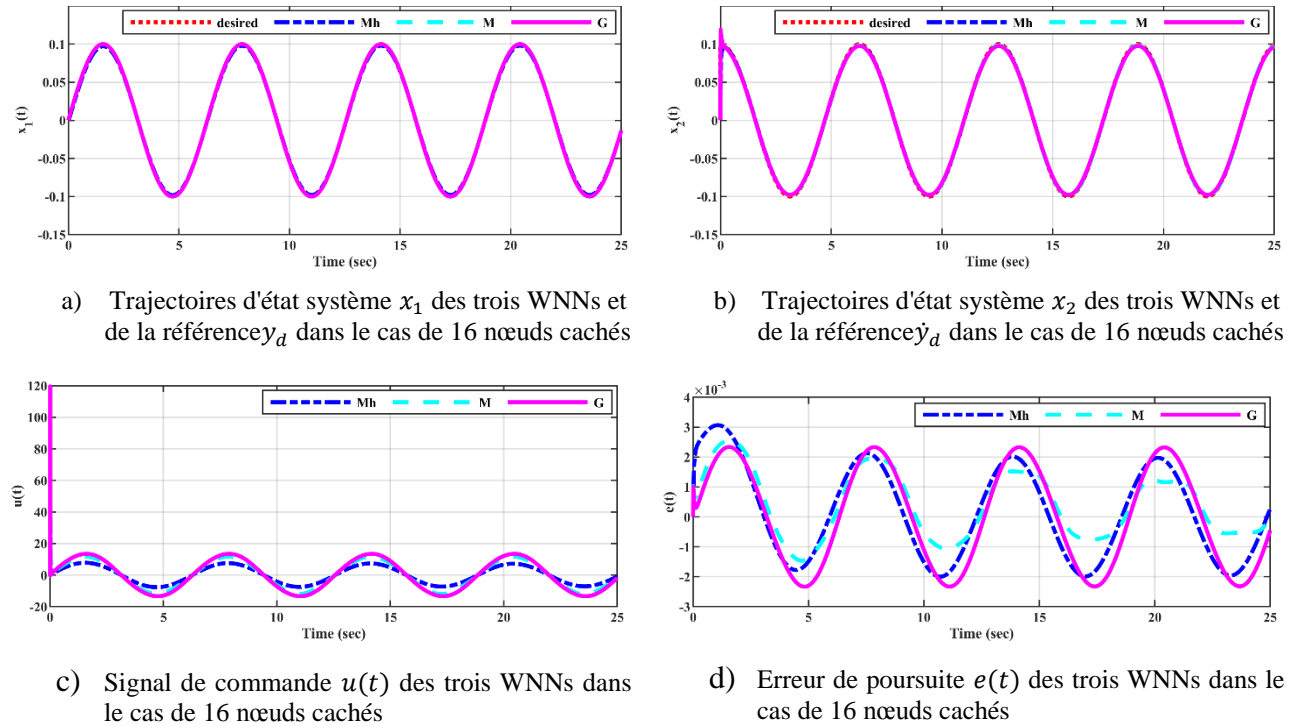


Figure 1.24 : Cas de 16 neurones cachés

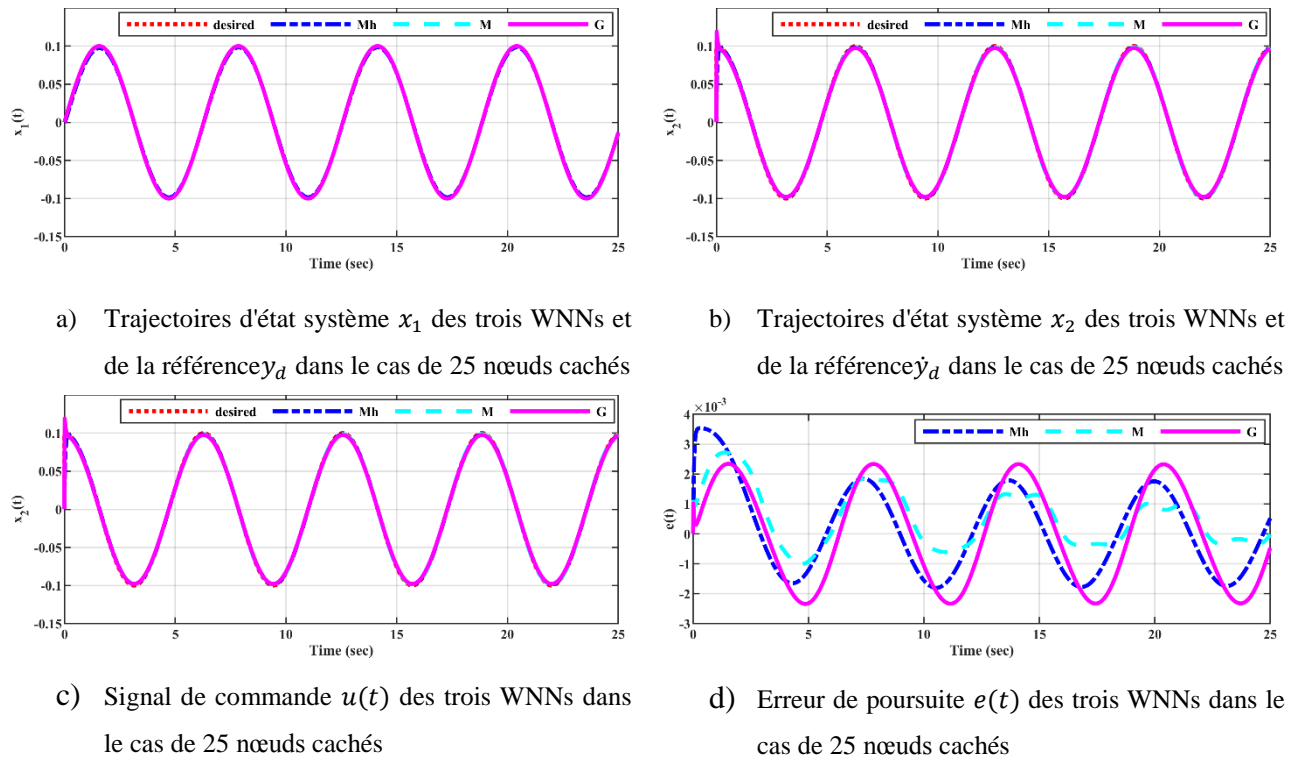


Figure 1.25 : Cas de 25 neurones cachés

On remarque des Figures 1.21 à 1.25 que la performance de poursuite s'améliore avec l'augmentation de la taille du réseau WNN. Malgré qu'avec un réseau de 02 neurones cachés on pouvait assurer une bonne poursuite (Figures 1.21 (a) et (b)) de la trajectoire désirée par les trois contrôleurs adaptatifs indirects, l'erreur de poursuite s'améliore considérablement avec un réseau de plus grande taille. Chose qu'on remarque clairement de la Figure 1.25 (d) avec un réseau de 25 neurones cachés. Cependant, la fonction Chapeau Mexicain a montré une amélioration de l'erreur de poursuite dès que le nombre de neurones cachés a augmenté par 02 neurones supplémentaires. Le temps que les autres familles n'ont pas pu bénéficier de cette augmentation et l'erreur de poursuite est restée inchangée. Pour l'ondelette Gaussienne et Morlet, il fallait atteindre au moins 09 neurones cachés pour remarquer une amélioration de l'erreur de poursuite (Figures 1.23 (d)).

Pour analyser plus précisément les performances des trois contrôleurs adaptatifs indirects, on compare leurs performances en régime transitoire et en régime permanent en utilisant deux critères de comparaison arithmétique. Les critères de comparaison sont la fonction MSE et le maximum de la valeur absolue de l'effort de contrôle ($Max |u|$). Pour observer l'influence de la fonction d'activation sur la taille du réseau, on considère des réseaux de neurones de 02 nœuds cachés jusqu'à 25 neurones cachés. Les résultats obtenus sont donnés dans le Tableau 1.4 pour le temps (0

sec à 06 sec) comme période transitoire et Tableau 1.5 pour le temps (06 sec à 25 sec) comme période permanente.

Tableau 1.4 : Comparaison des performances de poursuite des trois contrôleurs WNNs indirectes en période transitoire.

WNN	Chapeau Mexicain		Morlet		Ondelette Gaussienne	
	<i>MSE</i>	<i>MAX u </i>	<i>MSE</i>	<i>MAX u </i>	<i>MSE</i>	<i>MAX u </i>
n_c						
2	9.3853e-07	106.2778	6.5216e-07	109.0195	5.4177e-07	119.5415
4	1.3998e-06	95.1271	6.0413e-07	104.0629	5.4960e-07	119.4945
9	3.1707e-06	70.4512	1.1347e-06	80.0950	5.6612e-07	119.5415
16	6.1147e-06	47.2940	1.2737e-06	74.9605	5.8847e-07	119.5419
25	1.1219e-05	23.4792	2.1269e-06	48.8364	6.2118e-07	119.5419

Tableau 1.5 : Comparaison des performances de poursuite des trois contrôleurs WNNs indirectes en régime permanent.

WNN	Chapeau Mexicain		Morlet		Ondelette Gaussienne	
	<i>MSE</i>	<i>MAX u </i>	<i>MSE</i>	<i>MAX u </i>	<i>MSE</i>	<i>MAX u </i>
n_c						
2	2.7576e-06	12.3367	2.3639e-06	13.1357	2.7958e-06	13.4228
4	2.7402e-06	11.4358	2.5085e-06	13.0677	2.8145e-06	13.4379
9	2.5940e-06	9.5181	1.9904e-06	12.2278	2.7903e-06	13.4344
16	2.3128e-06	7.7990	1.3726e-06	12.2006	2.7752e-06	13.4475
25	1.9470e-06	6.1326	1.2217e-06	11.3404	2.7722e-06	13.4697

De Tableau 1.4 de comparaison des performances de poursuite des trois contrôleurs WNNs indirects en période transitoire, si on compare l'erreur de poursuite *MSE* entre les trois familles d'ondelettes pour chaque n_c on trouve que l'ondelette gaussienne a réalisé la plus petite erreur de poursuite et le Chapeau Mexicain a donné la plus grande erreur de poursuite. Par contre l'ondelette gaussienne a gardé un effort de control élevé contrairement aux autres familles qui ont montrés des efforts de contrôle de plus en plus faibles avec l'augmentation de n_c .

De Tableau 1.5 de comparaison des performances de poursuite des trois contrôleurs WNNs indirects en régime permanent, si on compare l'erreur de poursuite MSE entre les trois familles d'ondelettes pour chaque n_c on trouve que l'ondelette gaussienne a donné l'erreur de poursuite la plus élevée et l'ondelette Morlet a pu réaliser l'erreur de poursuite la plus petite en régime permanent. Et l'ondelette Chapeau Mexicain est classé entre les deux. Par contre l'ondelette Chapeau Mexicain a présenté le signal de commande le plus faible entre les trois familles. De ce fait l'ondelette Chapeau mexicain a pu maintenir une bonne poursuite avec un effort de control faible par rapport aux autres familles ce qui confirme notre classement dans le cas de commande adaptative directe. Pour cela on considère l'ondelette Chapeau mexicain dans le design des WNNs dans le reste du travail.

7. Conclusion

Dans ce chapitre, on a présenté la structure détaillée des réseaux de neurones à base d'ondelettes et les familles d'ondelettes les plus utilisées dans la littérature. Pour investiguer l'influence du choix de la famille d'ondelettes sur la qualité d'approximation du réseau WNN, on a élaboré une étude comparative des performances des familles d'ondelettes dans le contrôle adaptatif neuronal direct et indirect des systèmes non linéaires incertains. Trois réseaux WNNs ont été synthétisés en utilisant trois familles d'ondelettes comme des fonctions d'activation: Chapeau Mexicain, Morlet et l'ondelette Gaussienne. L'étude comparative de ces familles a été réalisée en utilisant le MSE de l'erreur de poursuite et le maximum d'effort de contrôle $Max(/u/)$. La méthode directe de Lyapunov a été utilisée pour prouver la stabilité du système de commande sous ces contrôleurs et pour dériver les lois d'adaptation des paramètres des réseaux WNNs. Il a été démontré à travers cette étude que les performances du contrôleur intelligent neuronal à base d'ondelettes sont fortement influencées par le choix de la famille d'ondelette qui sert comme des fonctions d'activation. Les résultats de simulation sur un système Pendule inversé SISO ont démontré que le chapeau mexicain est la fonction d'activation la plus appropriée pour les objectifs de contrôle adaptatif direct et indirect des systèmes avec une caractéristique de convergence rapide en période transitoire et la plus petite erreur statique, tandis que l'ondelette Gaussienne est arrivée en deuxième position avec une convergence plus lente en période transitoire. L'ondelette Morlet est classée dans cette étude en troisième position, avec la plus grande erreur permanente et l'effort de contrôle le plus élevé. Il a été démontré à travers cette étude que la taille de réseau dépend aussi du type de famille d'ondelettes choisie comme fonction d'activation tel qu'un réseau de neurones

avec 04 neurones cachés et des fonction d'activation Chapeaux Mexicains donne la même précision qu'un réseau de neurones qui contient 25 neurones cachés avec fonctions d'activation ondelettes Gaussiennes (voir Tableau 1.2 et Tableau 1.5 en gras). Le WNN synthétisé avec l'ondelette Chapeau Mexicain sera exploité pour la commande des systèmes MIMO non linéaires incertains et complexes dans le chapitre suivant.

Dans le chapitre suivant, nous faisons l'extension de la commande adaptative neuronale indirecte à base d'ondelettes aux systèmes MIMO non linéaires incertains.

Chapitre 2

Commande adaptative neuronale indirecte à base d'ondelettes des systèmes MIMO incertains

1. Introduction

La commande adaptative est une technique qui revient aux années 1950s [71], lorsque les automaticiens se sont rendus compte qu'un contrôleur à paramètres fixes ne pouvait pas assurer les performances désirées dans le cas où les caractéristiques du système évoluent dans le temps [72]. Elle comprend le réajustement de certains paramètres impliqués dans le calcul de la loi de commande en fonction de la dynamique du processus afin de ramener l'erreur de poursuite de la trajectoire désirée à zéro et maintenir les performances du système lorsque les paramètres du modèle changent.

La commande adaptative indirecte des systèmes incertains est basée sur l'approximation du modèle du système en utilisant les systèmes intelligents (les réseaux de neurones conventionnels, les réseaux de neurones à base d'ondelettes, ... etc). La commande adaptative a été combinée avec les réseaux de neurones à base d'ondelettes pour surmonter le problème de connaissance du modèle du système. Les contrôleurs adaptatifs neuronaux à base d'ondelettes offrent une meilleure approximation avec un temps de convergence minimale par rapport aux contrôleurs adaptatifs neuronaux conventionnels [73].

Dans le cas de contrôle neuronal adaptatif indirect à base d'ondelettes, le calcul se base sur l'identification neuronale du système à contrôler. Le principe du contrôle neuronal adaptatif indirect se résume en deux étapes:

- Préparation du modèle d'identification neuronal WNN représentant le système à contrôler en ligne.
- Calcul de l'action de contrôle adaptatif à appliquer au système pour que sa sortie suit avec précision la référence.

Pour la structure de contrôle adaptatif indirect à base d'ondelettes, les paramètres du contrôleur sont ajustés de façon à minimiser l'écart entre la sortie du système et le signal de référence. Le signal de commande généré par le contrôleur est utilisé comme une entrée pour le modèle neuronal et le système commandé. L'erreur entre la sortie du système et la référence désirée sera utilisée pour ajuster les paramètres du modèle neuronal. Pour ce faire, On augmente la taille entrée/sortie du réseau WNN présenté dans le chapitre précédent pour qu'il soit adapté à l'identification et le contrôle des systèmes MIMO. On choisit comme fonction d'activation l'ondelette Chapeau Mexicain car elle est la mieux classée par l'étude comparative réalisée auparavant dans le chapitre un. On utilise ce réseau de neurones MIMO à base d'ondelettes pour approximer les dynamiques incertaines d'un système MIMO. Les approximations obtenues seront exploitées dans la conception des lois de commandes adaptatives qui assurent la stabilité du système commandé. Une analyse des performances du contrôleur WNN adaptatif indirect dans le contrôler d'un Robot manipulateur MIMO incertains à deux degrés de liberté est donnée par simulation et des conclusions sont établis.

2. Réseau de neurones MIMO à base d'ondelettes

Un réseau de neurones multi-entrées multi-sorties MIMO à base d'ondelettes (WNN) est un réseau de neurones à trois couches avec des fonctions d'activation ondelettes dans la couche cachée. La structure d'un réseau de neurone MIMO à base d'ondelettes est présentée dans la Figure 2.1, avec $\mathbf{x}^T = [x_1 x_2 \dots x_n] \in R^n$ est le vecteur d'entrée du réseau WNN. h_{WNNj} est la j^{eme} sortie du réseau WNN avec $j=1, \dots, m$ et m est le nombre total des sorties. $\psi_i = \sum_{k=1}^n \psi_{ki}$ avec ψ_{ki} sont la fonctions d'activation ondelettes de la couche cachée. n_c représente le nombre total des neurones cachés [74].

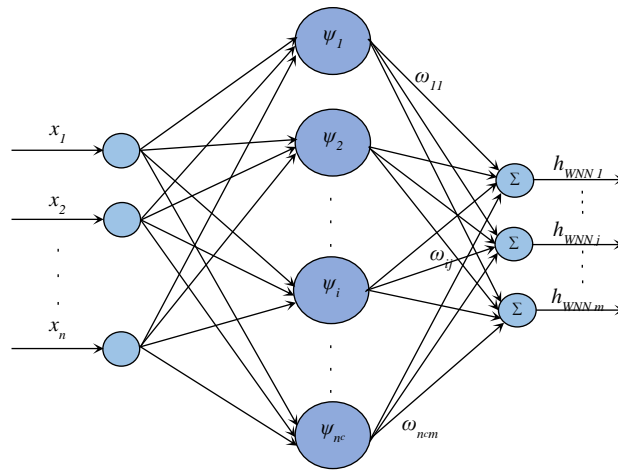


Figure 2.1 : Réseau de neurones MIMO à base d'ondelettes

L'expression de la fonction chapeau Mexicain est donnée par:

$$\psi_{ki}(x_k, c_{ki}) = (1 - (\|x_k - c_{ki}\|)^2) e^{-\frac{1}{2}(\|x_k - c_{ki}\|)^2} \quad (2.1)$$

La j^{eme} sortie du réseau de neurones de la Figure 1.13 est calculée comme suit:

$$h_{WNN_j}(\mathbf{x}) = \sum_{i=1}^{n_c} \omega_{ij} \psi_i(\mathbf{x}, \mathbf{c}_i) = \mathbf{w}_j^T \Psi(\mathbf{x}, \mathbf{C}) \quad (2.2)$$

avec c_{ki} sont les paramètres de translation de la fonction d'activation ψ_i par rapport à l'entrée x_k .

avec $\mathbf{C} = [\mathbf{c}_1 \mathbf{c}_2 \dots \mathbf{c}_{n_c}] \in R^{n \times n_c}$ est la matrice des paramètres de translation dont $\mathbf{c}_i^T = [c_{1i} c_{2i} \dots c_{ni}] \in R^n$. De plus, $\Psi^T = [\psi_1 \psi_2 \dots \psi_{n_c}] \in R^{n_c}$ est le vecteur des fonctions d'activation de la couche cachée et $\mathbf{w}_j^T = [\omega_{1j} \omega_{2j} \dots \omega_{n_c j}] \in R^{n_c}$ est un vecteur des poids ω_{ij} qui connectent le nœud caché i avec la sortie h_{WNN_j} . la sortie globale du réseau de neurone à base d'ondelettes de la Figure 1.13 est donnée sous une forme compacte comme suit:

$$h_{WNN} = \begin{bmatrix} h_{WNN_1} \\ h_{WNN_2} \\ \vdots \\ h_{WNN_m} \end{bmatrix} = \begin{bmatrix} \mathbf{w}_1^T \Psi \\ \mathbf{w}_2^T \Psi \\ \vdots \\ \mathbf{w}_m^T \Psi \end{bmatrix} = \mathbf{W}^T \Psi(\mathbf{x}, \mathbf{C}) \quad (2.3)$$

Avec $\mathbf{W} = [\mathbf{w}_1 \mathbf{w}_2 \dots \mathbf{w}_m] \in R^{n_c \times m}$. Se référant au théorème d'approximation universel [75], théoriquement, il existe \mathbf{W}^* et \mathbf{C}^* matrices des paramètres optimaux de \mathbf{W} et \mathbf{C} pour approximer $h_{WNN}(\mathbf{x}) \in R^m$ et minimiser le vecteur des erreurs d'approximation $\boldsymbol{\varepsilon}_{WNN}(\mathbf{x})$ [76]:

$$h_{WNN}(\mathbf{x}) = \mathbf{W}^{*T} \Psi^*(\mathbf{x}, \mathbf{C}^*) + \boldsymbol{\varepsilon}_{WNN}(\mathbf{x}) \quad (2.4)$$

Hypothèse 2:

Le vecteur des erreurs d'approximation $\boldsymbol{\varepsilon}_{WNN}(\mathbf{x})$ est borné par $\bar{\boldsymbol{\varepsilon}}_{WNN}$, qui est une constante positive inconnue.

Les fonctions inconnues du system $\mathbf{F}(\mathbf{x})$ et $\mathbf{G}(\mathbf{x})$ peuvent être approximées en utilisant des réseaux de neurones WNNs MIMO à base d'ondelettes comme suit:

$$\begin{aligned} \mathbf{F}(\mathbf{x}) &= \mathbf{F}_{WNN}^*(\mathbf{x}, \mathbf{W}_F^*, \mathbf{C}_F^*) + \boldsymbol{\varepsilon}_{F_{WNN}}(\mathbf{x}) \\ &= \mathbf{W}_F^{*T} \Psi_F^*(\mathbf{x}, \mathbf{C}_F^*) + \boldsymbol{\varepsilon}_{F_{WNN}}(\mathbf{x}) \end{aligned} \quad (2.5)$$

$$\begin{aligned} \mathbf{G}(\mathbf{x}) &= \mathbf{G}_{WNN}^*(\mathbf{x}, \mathbf{W}_G^*, \mathbf{C}_G^*) + \varepsilon_{G_{WNN}}(\mathbf{x}) \\ &= \mathbf{W}_G^{*T} \Psi_G^*(\mathbf{x}, \mathbf{C}_G^*) + \varepsilon_{G_{WNN}}(\mathbf{x}) \end{aligned} \quad (2.6)$$

Pratiquement, les paramètres optimaux \mathbf{W}^* et \mathbf{C}^* ne peuvent pas être déterminés, leurs estimations $\widehat{\mathbf{W}}$ et $\widehat{\mathbf{C}}$ sont utilisées. Par conséquent, les fonctions inconnues du système sont approximées comme suit:

$$\widehat{\mathbf{F}}_{WNN}(\mathbf{x}, \widehat{\mathbf{W}}_F, \widehat{\mathbf{C}}_F) = \widehat{\mathbf{W}}_F^T \widehat{\Psi}_F(\mathbf{x}, \widehat{\mathbf{C}}_F) \quad (2.7)$$

$$\widehat{\mathbf{G}}_{WNN}(\mathbf{x}, \widehat{\mathbf{W}}_G, \widehat{\mathbf{C}}_G) = \widehat{\mathbf{W}}_G^T \widehat{\Psi}_G(\mathbf{x}, \widehat{\mathbf{C}}_G) \quad (2.8)$$

Avec :

$$\begin{aligned} \mathbf{F}_{NN}^*(\mathbf{x}, \mathbf{W}_F^*, \mathbf{C}_F^*) &= \widehat{\mathbf{F}}_{WNN}(\mathbf{x}, \widehat{\mathbf{W}}_F, \widehat{\mathbf{C}}_F) + \widetilde{\mathbf{F}}_{WNN}(\mathbf{x}, \widetilde{\mathbf{W}}_F, \widetilde{\mathbf{C}}_F) \\ &= \widehat{\mathbf{W}}_F^T \widehat{\Psi}_F(\mathbf{x}, \widehat{\mathbf{C}}_F) + \widetilde{\mathbf{W}}_F^T \widetilde{\Psi}_F(\mathbf{x}, \widetilde{\mathbf{C}}_F) \end{aligned} \quad (2.9)$$

$$\begin{aligned} \mathbf{G}_{WNN}^*(\mathbf{x}, \mathbf{W}_G^*, \mathbf{C}_G^*) &= \widehat{\mathbf{G}}_{WNN}(\mathbf{x}, \widehat{\mathbf{W}}_G, \widehat{\mathbf{C}}_G) + \widetilde{\mathbf{G}}_{WNN}(\mathbf{x}, \widetilde{\mathbf{W}}_G, \widetilde{\mathbf{C}}_G) \\ &= \widehat{\mathbf{W}}_G^T \widehat{\Psi}_G(\mathbf{x}, \widehat{\mathbf{C}}_G) + \widetilde{\mathbf{W}}_G^T \widetilde{\Psi}_G(\mathbf{x}, \widetilde{\mathbf{C}}_G) \end{aligned} \quad (2.10)$$

Donc, les erreurs d'approximation des paramètres sont données comme suit: $\widetilde{\mathbf{W}}_F = \mathbf{W}_F^* - \widehat{\mathbf{W}}_F$, $\widetilde{\mathbf{C}}_F = \mathbf{C}_F^* - \widehat{\mathbf{C}}_F$, $\widetilde{\mathbf{W}}_G = \mathbf{W}_G^* - \widehat{\mathbf{W}}_G$ et $\widetilde{\mathbf{C}}_G = \mathbf{C}_G^* - \widehat{\mathbf{C}}_G$.

3. Commande adaptative indirecte par réseaux de neurones MIMO à base d'ondelettes

Considérant une classe des systèmes MIMO non linéaires incertains représentés par l'équation différentielle suivante [30]:

$$\mathbf{y}^{(r)} = \mathbf{F}(\mathbf{x}) + \mathbf{G}(\mathbf{x})\mathbf{u} + \mathbf{d}(\mathbf{x}) \quad (2.11)$$

Avec $\mathbf{x}^T = [y_1, \dot{y}_1, y_1^{(r_1-1)}, \dots, y_m, \dot{y}_m, y_m^{(r_m-1)}] \in R^n$ est le vecteur d'état, $\mathbf{u} \in R^m$ est l'entrée de commande. $\mathbf{y}^T = [y_1 \dots y_m] \in R^m$ est le vecteur des sorties du système. $\mathbf{y}^{(r)} = [y_1^{(r_1)} \dots y_m^{(r_m)}]^T \in R^m$, $r = (r_1, r_2, \dots, r_m)$ est le vecteur des degrés relatifs du système avec $r_1 + r_2 + \dots + r_m = n$

et n est l'ordre du système. $\mathbf{F}(\mathbf{x}) \in R^m$ et $\mathbf{G}(\mathbf{x}) \in R^{m \times m}$ sont des fonctions non linéaires inconnues. $\mathbf{d}(\mathbf{x}) \in R^m$ est le vecteur des perturbations.

L'objectif de la commande est de synthétiser une loi de commande de tel façon que la sortie du système \mathbf{y} suit une trajectoire désirée $\mathbf{y}_d^T = [y_{d_1} \cdots y_{d_m}] \in R^m$. Pour ce faire, il est nécessaire d'établir quelques hypothèses:

Hypothèse 3:

Vecteur des états \mathbf{x} est supposé observable [30].

Hypothèse 4:

Vecteur des perturbations est supposé bornée par [30]: $|d_i| \leq D_i$ for $i = 1, \dots, m$.

Hypothèse 5:

La trajectoire désirée de référence y_{d_i} est continue et bornée ainsi que toutes ses dérivées jusqu'à l'ordre $r_i, i=1, \dots, m$ [30].

Définissant l'erreur de poursuite $\mathbf{e} \in R^m$ comme suit :

$$\mathbf{e} = \mathbf{y}_d - \mathbf{y} \quad (2.12)$$

Avec :

$$\mathbf{e} = [e_1 \cdots e_m]^T \quad (2.13)$$

La dérivée par rapport au temps de (2.12) est donnée par :

$$\dot{\mathbf{e}}(t) = \dot{\mathbf{y}}_d(t) - \dot{\mathbf{y}}(t) \quad (2.14)$$

La seconde dérivée par rapport au temps de (2.14) est donnée par :

$$\ddot{\mathbf{e}}(t) = \ddot{\mathbf{y}}_d(t) - \ddot{\mathbf{y}}(t) \quad (2.15)$$

On définit l'erreur filtrée de poursuite $\mathbf{s} \in R^m$ comme suit [31]:

$$\begin{aligned} \mathbf{s} &= [s_1 \cdots s_m]^T \\ &= \mathbf{e}^{(r-1)} + \lambda_1 \mathbf{e}^{(r-2)} + \cdots + \lambda_{r-2} \dot{\mathbf{e}} + \lambda_{r-1} \mathbf{e} \end{aligned} \quad (2.16)$$

Où $\lambda_i \in R^{m \times m}, i=1, \dots, r-1$ sont choisi comme des matrices diagonales des constantes positives tel que les racines du polynôme $\mathbf{s}=0$ sont dans le demi-plan gauche du plan complexe.

La dérivée par rapport au temps de (2.16) est donnée par:

$$\dot{\mathbf{s}} = \mathbf{v} - \mathbf{F}(\mathbf{x}) - \mathbf{G}(\mathbf{x})\mathbf{u} - \mathbf{d}(\mathbf{x}) \quad (2.17)$$

Avec $\mathbf{v} = \mathbf{y}_d^{(r)} + \lambda_1 \mathbf{e}^{(r-1)} + \dots + \lambda_{r-2} \mathbf{e}^{(2)} + \lambda_{r-1} \dot{\mathbf{e}}$. Dans le cas où les fonctions $\mathbf{F}(\mathbf{x})$ et $\mathbf{G}(\mathbf{x})$ sont connues avec précision et le système est indépendant des perturbations externes, la loi de commande optimale \mathbf{u}^* sera choisie comme suit:

$$\mathbf{u}^* = \mathbf{u}_{eq} + \mathbf{u}_r \quad (2.18)$$

Où:

$$\mathbf{u}_{eq} = \mathbf{G}(\mathbf{x})^{-1}(-\mathbf{F}(\mathbf{x}) + \mathbf{v}) \quad (2.19)$$

$$\mathbf{u}_r = \mathbf{G}(\mathbf{x})^{-1}K \operatorname{sgn}(\mathbf{s}) \quad (2.20)$$

Avec $K = \operatorname{diag}(k_1, \dots, k_m) \in R^{m \times m}$, $k_i > 0$, $i=1, \dots, m$ et $\operatorname{sgn}(\cdot)$ est la fonction signe. En remplaçant (2.18) dans (2.17), on obtient $\dot{\mathbf{s}} = -K \operatorname{sgn}(\mathbf{s})$. Ce qui implique $\mathbf{s}(t) \rightarrow 0$ quand $t \rightarrow \infty$. Par analogie, $\mathbf{e}(t) \rightarrow 0$ et tous ses dérivées jusqu'à l'ordre $r_i - 1$. Cependant, dans la plupart des cas, les fonctions non linéaires du système $\mathbf{F}(\mathbf{x})$ et $\mathbf{G}(\mathbf{x})$ sont inconnues. La loi de commande optimale \mathbf{u}^* ne peut pas être implémentée. Pour synthétiser la loi de commande \mathbf{u} , les fonctions $\mathbf{F}(\mathbf{x})$ et $\mathbf{G}(\mathbf{x})$ sont approximées par des approximateurs universel neuronaux à base d'ondelettes WNNs de la Figure 2.1, leurs expressions sont données par les équations (2.7) et (2.8). Donc la loi de commande neuronale adaptative à base d'ondelettes est synthétisée comme suit :

On reprend l'équation (2.17) et on rajoute et on soustraire $\widehat{\mathbf{G}}_{WNN}(\mathbf{x}, \widehat{\mathbf{W}}_G, \widehat{\mathbf{C}}_G)\mathbf{u}$, on aura:

$$\dot{\mathbf{s}} = \mathbf{v} - \mathbf{F}(\mathbf{x}) - \left(\mathbf{G}(\mathbf{x}) - \widehat{\mathbf{G}}_{WNN}(\mathbf{x}, \widehat{\mathbf{W}}_G, \widehat{\mathbf{C}}_G) \right) \mathbf{u} - \widehat{\mathbf{G}}_{WNN}(\mathbf{x}, \widehat{\mathbf{W}}_G, \widehat{\mathbf{C}}_G)\mathbf{u} - \mathbf{d} \quad (2.21)$$

Et on choisit la loi de commande comme suit:

$$\mathbf{u} = \widehat{\mathbf{G}}_{WNN}(\mathbf{x}, \widehat{\mathbf{W}}_G, \widehat{\mathbf{C}}_G)^{-1} \left(\mathbf{v} + K \operatorname{sgn}(\mathbf{s}) - \widehat{\mathbf{F}}_{WNN}(\mathbf{x}, \widehat{\mathbf{W}}_F, \widehat{\mathbf{C}}_F) \right) \quad (2.22)$$

Pour éviter le phénomène de chattering engendré par le terme glissant dans (2.21), une loi de commande par PI (proportionnel-intégrateur) [32] est mise en œuvre quand les états du système est dans la couche cachée délimitée par ν , comme suit:

$$\mathbf{u}_r = \begin{cases} K_1 \mathbf{s} + K_2 \int \mathbf{s} dt, & |\mathbf{s}| < \nu \\ K \operatorname{sgn}(\mathbf{s}), & |\mathbf{s}| \geq \nu \end{cases} \quad (2.23)$$

Avec K_1, K_2, K sont des matrices de gains diagonales positives. ν est une constant positive. Donc, la loi de commande (2.22) devient:

$$\mathbf{u} = \widehat{\mathbf{G}}_{WNN}(\mathbf{x}, \widehat{\mathbf{W}}_G, \widehat{\mathbf{C}}_G)^{-1} (\mathbf{v} + \mathbf{u}_r - \widehat{\mathbf{F}}_{WNN}(\mathbf{x}, \widehat{\mathbf{W}}_F, \widehat{\mathbf{C}}_F)) \quad (2.24)$$

En remplaçant (2.24) dans (2.21), on obtient:

$$\dot{\mathbf{s}} = -[\mathbf{F}(\mathbf{x}) - \widehat{\mathbf{F}}_{WNN}(\mathbf{x}, \widehat{\mathbf{W}}_F, \widehat{\mathbf{C}}_F)] - [\mathbf{G}(\mathbf{x}) - \widehat{\mathbf{G}}_{WNN}(\mathbf{x}, \widehat{\mathbf{W}}_G, \widehat{\mathbf{C}}_G)]\mathbf{u} - \mathbf{u}_r - \mathbf{d} \quad (2.25)$$

On remplaçant par (2.5)-(2.6) dans (2.25), on trouve :

$$\begin{aligned} \dot{\mathbf{s}} = & -[\mathbf{F}_{WNN}^*(\mathbf{x}, \mathbf{W}_F^*, \mathbf{C}_F^*) + \boldsymbol{\varepsilon}_{F_{WNN}}(\mathbf{x}) - \widehat{\mathbf{F}}_{WNN}(\mathbf{x}, \widehat{\mathbf{W}}_F, \widehat{\mathbf{C}}_F)] \\ & - [\mathbf{G}_{WNN}^*(\mathbf{x}, \mathbf{W}_G^*, \mathbf{C}_G^*) + \boldsymbol{\varepsilon}_{G_{WNN}}(\mathbf{x}) - \widehat{\mathbf{G}}_{WNN}(\mathbf{x}, \widehat{\mathbf{W}}_G, \widehat{\mathbf{C}}_G)]\mathbf{u} - \mathbf{u}_r - \mathbf{d} \end{aligned} \quad (2.26)$$

Qui est équivalent à :

$$\begin{aligned} \dot{\mathbf{s}} = & -\left[\left(\mathbf{F}_{WNN}^*(\mathbf{x}, \mathbf{W}_F^*, \mathbf{C}_F^*) - \widehat{\mathbf{F}}_{WNN}(\mathbf{x}, \widehat{\mathbf{W}}_F, \widehat{\mathbf{C}}_F) \right) + \boldsymbol{\varepsilon}_{F_{WNN}}(\mathbf{x}) \right] \\ & - \left[\left(\mathbf{G}_{WNN}^*(\mathbf{x}, \mathbf{W}_G^*, \mathbf{C}_G^*) - \widehat{\mathbf{G}}_{WNN}(\mathbf{x}, \widehat{\mathbf{W}}_G, \widehat{\mathbf{C}}_G) \right) + \boldsymbol{\varepsilon}_{G_{WNN}}(\mathbf{x}) \right] \mathbf{u} - \mathbf{u}_r - \mathbf{d} \end{aligned} \quad (2.27)$$

En remplaçant par (2.5)-(2.6) et (2.7)-(2.8) dans (2.27), on trouve:

$$\begin{aligned} \dot{\mathbf{s}} = & -\left[\mathbf{W}_F^{*T} \boldsymbol{\Psi}_F^*(\mathbf{x}, \mathbf{C}_F^*) - \widehat{\mathbf{W}}_F^T \widehat{\boldsymbol{\Psi}}_F(\mathbf{x}, \widehat{\mathbf{C}}_F) + \boldsymbol{\varepsilon}_{F_{WNN}}(\mathbf{x}) \right] \\ & - \left[\left(\mathbf{W}_G^{*T} \boldsymbol{\Psi}_G^*(\mathbf{x}, \mathbf{C}_G^*) - \widehat{\mathbf{W}}_G^T \widehat{\boldsymbol{\Psi}}_G(\mathbf{x}, \widehat{\mathbf{C}}_G) \right) + \boldsymbol{\varepsilon}_{G_{WNN}}(\mathbf{x}) \right] \mathbf{u} - \mathbf{u}_r - \mathbf{d} \end{aligned} \quad (2.28)$$

On utilise (2.9)-(2.10), pour développer (2.28) comme suit:

$$\begin{aligned} \dot{\mathbf{s}} = & -\left[\left(\widehat{\mathbf{W}}_F^T \boldsymbol{\Psi}_F^*(\mathbf{x}, \mathbf{C}_F^*) + \widetilde{\mathbf{W}}_F^T \boldsymbol{\Psi}_F^*(\mathbf{x}, \mathbf{C}_F^*) \right) - \widehat{\mathbf{W}}_F^T \widehat{\boldsymbol{\Psi}}_F(\mathbf{x}, \widehat{\mathbf{C}}_F) + \boldsymbol{\varepsilon}_{F_{WNN}}(\mathbf{x}) \right] \\ & - \left[\widehat{\mathbf{W}}_G^T \boldsymbol{\Psi}_G^*(\mathbf{x}, \mathbf{C}_G^*) + \widetilde{\mathbf{W}}_G^T \boldsymbol{\Psi}_G^*(\mathbf{x}, \mathbf{C}_G^*) - \widehat{\mathbf{W}}_G^T \widehat{\boldsymbol{\Psi}}_G(\mathbf{x}, \widehat{\mathbf{C}}_G) + \boldsymbol{\varepsilon}_{G_{WNN}}(\mathbf{x}) \right] \mathbf{u} - \mathbf{u}_r - \mathbf{d} \end{aligned} \quad (2.29)$$

Après simplification, (2.29) devient:

$$\begin{aligned} \dot{\mathbf{s}} = & - \left[\widehat{\mathbf{W}}_F^T \widetilde{\Psi}_F(\mathbf{x}, \widetilde{\mathbf{C}}_F) + \widehat{\mathbf{W}}_F^T \Psi_F^*(\mathbf{x}, \mathbf{C}_F^*) + \boldsymbol{\varepsilon}_{F_{WNN}}(\mathbf{x}) \right] \\ & - \left[\widehat{\mathbf{W}}_G^T \widetilde{\Psi}_G(\mathbf{x}, \widetilde{\mathbf{C}}_G) + \widehat{\mathbf{W}}_G^T \Psi_G^*(\mathbf{x}, \mathbf{C}_G^*) + \boldsymbol{\varepsilon}_{G_{WNN}}(\mathbf{x}) \right] \mathbf{u} - \mathbf{u}_r - \mathbf{d} \end{aligned} \quad (2.30)$$

On exploite (2.9)-(2.10) pour aboutir à:

$$\begin{aligned} \dot{\mathbf{s}} = & - \left[\widehat{\mathbf{W}}_F^T \widetilde{\Psi}_F(\mathbf{x}, \widetilde{\mathbf{C}}_F) + \widehat{\mathbf{W}}_F^T \left(\widehat{\Psi}_F(\mathbf{x}, \widehat{\mathbf{C}}_F) + \widetilde{\Psi}_F(\mathbf{x}, \widetilde{\mathbf{C}}_F) \right) + \boldsymbol{\varepsilon}_{F_{WNN}}(\mathbf{x}) \right] \\ & - \left[\widehat{\mathbf{W}}_G^T \widetilde{\Psi}_G(\mathbf{x}, \widetilde{\mathbf{C}}_G) + \widehat{\mathbf{W}}_G^T \left(\widehat{\Psi}_G(\mathbf{x}, \widehat{\mathbf{C}}_G) + \widetilde{\Psi}_G(\mathbf{x}, \widetilde{\mathbf{C}}_G) \right) + \boldsymbol{\varepsilon}_{G_{WNN}}(\mathbf{x}) \right] \mathbf{u} - \mathbf{u}_r - \mathbf{d} \end{aligned} \quad (2.31)$$

En utilisant la linéarisation de Taylor [77] (voir annexe A), on trouve que:

$$\widetilde{\Psi}_F(\mathbf{x}, \widetilde{\mathbf{C}}_F) = \boldsymbol{\Omega}_F^T \widetilde{\mathbf{C}}_F + O_F \quad (2.32)$$

$$\widetilde{\Psi}_G(\mathbf{x}, \widetilde{\mathbf{C}}_G) = \boldsymbol{\Omega}_G^T \widetilde{\mathbf{C}}_G + O_G \quad (2.33)$$

Où $\widetilde{\mathbf{C}}_F = \mathbf{C}_F^* - \widehat{\mathbf{C}}_F$, $\widetilde{\mathbf{C}}_G = \mathbf{C}_G^* - \widehat{\mathbf{C}}_G$ et $O \in R^{n_c}$ est vecteur des terme des ordres plus élevés, aussi:

$$\boldsymbol{\Omega} = \left[\frac{\partial \psi_1}{\partial \mathbf{C}} \quad \frac{\partial \psi_2}{\partial \mathbf{C}} \quad \dots \quad \frac{\partial \psi_{n_c}}{\partial \mathbf{C}} \right]^T \bigg|_{\mathbf{C}=\widehat{\mathbf{C}}} \in R^{n_c \times n_c} \quad (2.34)$$

$$\text{et } \frac{\partial \psi_i}{\partial \mathbf{C}} = \left[\underbrace{0 \quad \dots \quad 0}_{(i-1)n} \quad \frac{\partial \psi_i}{\partial \mathbf{C}_{1i}} \quad \dots \quad \frac{\partial \psi_i}{\partial \mathbf{C}_{ni}} \quad \underbrace{0 \quad \dots \quad 0}_{(n_c-i)n} \right]^T.$$

Alors, on remplace par (2.32)-(2.33) dans (2.31) comme suit :

$$\begin{aligned} \dot{\mathbf{s}} = & - \left[\widehat{\mathbf{W}}_F^T (\boldsymbol{\Omega}_F^T \widetilde{\mathbf{C}}_F + O_F) + \widehat{\mathbf{W}}_F^T \left(\widehat{\Psi}_F(\mathbf{x}, \widehat{\mathbf{C}}_F) + (\boldsymbol{\Omega}_F^T \widetilde{\mathbf{C}}_F + O_F) \right) + \boldsymbol{\varepsilon}_{F_{WNN}}(\mathbf{x}) \right] \\ & - \left[\widehat{\mathbf{W}}_G^T (\boldsymbol{\Omega}_G^T \widetilde{\mathbf{C}}_G + O_G) + \widehat{\mathbf{W}}_G^T \left(\widehat{\Psi}_G(\mathbf{x}, \widehat{\mathbf{C}}_G) + (\boldsymbol{\Omega}_G^T \widetilde{\mathbf{C}}_G + O_G) \right) + \boldsymbol{\varepsilon}_{G_{WNN}}(\mathbf{x}) \right] \mathbf{u} - \mathbf{u}_r - \mathbf{d} \end{aligned} \quad (2.35)$$

On exploitant les relations $\widetilde{\mathbf{C}}_F = \mathbf{C}_F^* - \widehat{\mathbf{C}}_F$ et $\widetilde{\mathbf{C}}_G = \mathbf{C}_G^* - \widehat{\mathbf{C}}_G$ pour réécrire (2.35) comme suit :

$$\begin{aligned} \dot{\mathbf{s}} = & - \left[\widehat{\mathbf{W}}_F^T (\boldsymbol{\Omega}_F^T \widetilde{\mathbf{C}}_F + O_F) + \widehat{\mathbf{W}}_F^T \left(\widehat{\Psi}_F(\mathbf{x}, \widehat{\mathbf{C}}_F) + \left((\boldsymbol{\Omega}_F^T \mathbf{C}_F^* - \boldsymbol{\Omega}_F^T \widehat{\mathbf{C}}_F) + O_F \right) \right) + \boldsymbol{\varepsilon}_{F_{WNN}}(\mathbf{x}) \right] \\ & - \left[\widehat{\mathbf{W}}_G^T (\boldsymbol{\Omega}_G^T \widetilde{\mathbf{C}}_G + O_G) + \widehat{\mathbf{W}}_G^T \left(\widehat{\Psi}_G(\mathbf{x}, \widehat{\mathbf{C}}_G) + \left((\boldsymbol{\Omega}_G^T \mathbf{C}_G^* - \boldsymbol{\Omega}_G^T \widehat{\mathbf{C}}_G) + O_G \right) \right) + \boldsymbol{\varepsilon}_{G_{WNN}}(\mathbf{x}) \right] \mathbf{u} \\ & - \mathbf{u}_r - \mathbf{d} \end{aligned} \quad (2.36)$$

Ce qui équivaut à :

$$\begin{aligned}
 \dot{\mathbf{s}} = & -[\widetilde{\mathbf{W}}_F^T \Omega_F^T \widetilde{\mathbf{C}}_F + \widetilde{\mathbf{W}}_F^T (\widehat{\Psi}_F(\mathbf{x}, \widehat{\mathbf{C}}_F) - \Omega_F^T \widehat{\mathbf{C}}_F) + \widetilde{\mathbf{W}}_F^T \Omega_F^T \mathbf{C}_F^* + (\widetilde{\mathbf{W}}_F^T + \widetilde{\mathbf{W}}_F^T) O_F + \boldsymbol{\varepsilon}_{F_{WNN}}(\mathbf{x})] \\
 & - [\widetilde{\mathbf{W}}_G^T \Omega_G^T \widetilde{\mathbf{C}}_G + \widetilde{\mathbf{W}}_G^T (\widehat{\Psi}_G(\mathbf{x}, \widehat{\mathbf{C}}_G) - \Omega_G^T \widehat{\mathbf{C}}_G) + \widetilde{\mathbf{W}}_G^T \Omega_G^T \mathbf{C}_G^* + (\widetilde{\mathbf{W}}_G^T + \widetilde{\mathbf{W}}_G^T) O_G \\
 & + \boldsymbol{\varepsilon}_{G_{WNN}}(\mathbf{x})] \mathbf{u} - \mathbf{u}_r - \mathbf{d}
 \end{aligned} \tag{2.37}$$

De (2.9)-(2.10), on peut écrire (2.37) comme :

$$\begin{aligned}
 \dot{\mathbf{s}} = & -[\widetilde{\mathbf{W}}_F^T \Omega_F^T \widetilde{\mathbf{C}}_F + \widetilde{\mathbf{W}}_F^T (\widehat{\Psi}_F(\mathbf{x}, \widehat{\mathbf{C}}_F) - \Omega_F^T \widehat{\mathbf{C}}_F) + \widetilde{\mathbf{W}}_F^T \Omega_F^T \mathbf{C}_F^* + \mathbf{W}_F^{*T} O_F + \boldsymbol{\varepsilon}_{F_{WNN}}(\mathbf{x})] \\
 & - [\widetilde{\mathbf{W}}_G^T \Omega_G^T \widetilde{\mathbf{C}}_G + \widetilde{\mathbf{W}}_G^T (\widehat{\Psi}_G(\mathbf{x}, \widehat{\mathbf{C}}_G) - \Omega_G^T \widehat{\mathbf{C}}_G) + \widetilde{\mathbf{W}}_G^T \Omega_G^T \mathbf{C}_G^* + \mathbf{W}_G^{*T} O_G + \boldsymbol{\varepsilon}_{G_{WNN}}(\mathbf{x})] \mathbf{u} \\
 & - \mathbf{u}_r - \mathbf{d}
 \end{aligned} \tag{2.38}$$

Mettant :

$$\begin{aligned}
 \Lambda_{F_{WNN}} & = -(\widetilde{\mathbf{W}}_F^T \Omega_F^T \mathbf{C}_F^* + \mathbf{W}_F^{*T} O_F) \\
 \Lambda_{G_{WNN}} & = -(\widetilde{\mathbf{W}}_G^T \Omega_G^T \mathbf{C}_G^* + \mathbf{W}_G^{*T} O_G) \mathbf{u}
 \end{aligned} \tag{2.39}$$

On obtient:

$$\begin{aligned}
 \dot{\mathbf{s}} = & -[\widetilde{\mathbf{W}}_F^T \Omega_F^T \widetilde{\mathbf{C}}_F + \widetilde{\mathbf{W}}_F^T (\widehat{\Psi}_F(\mathbf{x}, \widehat{\mathbf{C}}_F) - \Omega_F^T \widehat{\mathbf{C}}_F) + \Lambda_F + \boldsymbol{\varepsilon}_{F_{WNN}}(\mathbf{x})] \\
 & - [\widetilde{\mathbf{W}}_G^T \Omega_G^T \widetilde{\mathbf{C}}_G + \widetilde{\mathbf{W}}_G^T (\widehat{\Psi}_G(\mathbf{x}, \widehat{\mathbf{C}}_G) - \Omega_G^T \widehat{\mathbf{C}}_G) + \Lambda_G + \boldsymbol{\varepsilon}_{G_{WNN}}(\mathbf{x})] \mathbf{u} - \mathbf{u}_r - \mathbf{d}
 \end{aligned} \tag{2.40}$$

Réorganisant (2.40) comme:

$$\begin{aligned}
 \dot{\mathbf{s}} = & -[\widetilde{\mathbf{W}}_F^T \Omega_F^T \widetilde{\mathbf{C}}_F + \widetilde{\mathbf{W}}_F^T (\widehat{\Psi}_F(\mathbf{x}, \widehat{\mathbf{C}}_F) - \Omega_F^T \widehat{\mathbf{C}}_F) + \boldsymbol{\varepsilon}_{F_{WNN}}(\mathbf{x})] \\
 & - [\widetilde{\mathbf{W}}_G^T \Omega_G^T \widetilde{\mathbf{C}}_G + \widetilde{\mathbf{W}}_G^T (\widehat{\Psi}_G(\mathbf{x}, \widehat{\mathbf{C}}_G) - \Omega_G^T \widehat{\mathbf{C}}_G) + \boldsymbol{\varepsilon}_{G_{WNN}}(\mathbf{x})] \mathbf{u} - \mathbf{u}_r - \mathbf{d} + \Delta_{WNN}
 \end{aligned} \tag{2.41}$$

Avec $\Delta_{WNN} = -\Lambda_{F_{WNN}} - \Lambda_{G_{WNN}} \mathbf{u}$

Ainsi, (2.41) peut être exprimée sous une forme compacte de la manière suivante :

$$\begin{aligned}
 \dot{\mathbf{s}} = & -[\widetilde{\mathbf{F}}_{WNN}(\mathbf{x}, \widetilde{\mathbf{W}}_F, \widetilde{\mathbf{C}}_F) + \boldsymbol{\varepsilon}_{F_{WNN}}(\mathbf{x})] - [\widetilde{\mathbf{G}}_{WNN}(\mathbf{x}, \widetilde{\mathbf{W}}_G, \widetilde{\mathbf{C}}_G) + \boldsymbol{\varepsilon}_{G_{WNN}}(\mathbf{x})] \mathbf{u} - \mathbf{u}_r - \mathbf{d} \\
 & + \Delta_{WNN}
 \end{aligned} \tag{2.42}$$

Hypothèse 6:

L'erreur d'approximation est supposée être bornée par $\|\Delta_{WNN}\| \leq \Gamma_{WNN}$ [78].

Pour atteindre les objectifs de la commande, il est nécessaire d'ajuster les paramètres des approximateurs WNNs afin d'avoir $e(t) \rightarrow 0$ quand $t \rightarrow \infty$. Ainsi, les lois d'adaptation des paramètres des WNNs sont données comme suivant :

$$\dot{\hat{C}}_F = -\gamma_{C_F} \Omega_F \widehat{W}_F s \quad (2.43)$$

$$\dot{\widehat{W}}_F = -\gamma_{W_F} (\widehat{\Psi}_F(x, \widehat{C}_F) - \Omega_F^T \widehat{C}_F) s^T \quad (2.44)$$

$$\dot{\hat{C}}_G = -\gamma_{C_G} \Omega_G \widehat{W}_G s u^T \quad (2.45)$$

$$\dot{\widehat{W}}_G = -\gamma_{W_G} (\widehat{\Psi}_G(x, \widehat{C}_G) - \Omega_G^T \widehat{C}_G) u s^T \quad (2.46)$$

Avec $\gamma_{C_F}, \gamma_{C_G}, \gamma_{W_F}, \gamma_{W_G}$ sont des gains positifs d'adaptation, et $\tau_{C_F}, \tau_{C_G}, \tau_{W_F}, \tau_{W_G}$ sont des constantes positives.

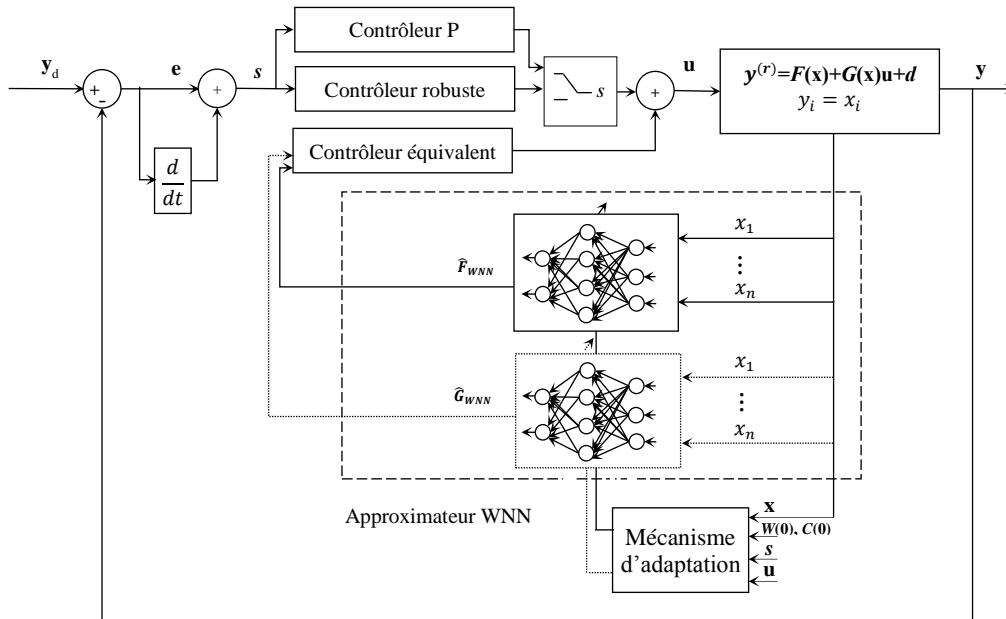


Figure 2.2 : Architecture d'une boucle de commande adaptative indirecte en utilisant des approximateurs WNNs

Théorème 1:

Considérant la boucle fermée de commande de la Figure 2.2, avec le système MIMO non linéaire incertain (2.11). La loi de commande neuronale adaptative à base d'ondelettes est choisie comme (2.24) et les lois d'adaptation des paramètres sont données à partir de (2.43)-(2.46). Le contrôleur à base des approximateurs WNNs adaptatifs garantit que le système de commande en boucle fermée est stable, et que tous les signaux sont bornés.

Preuve du Théorème 1:

Considérant une fonction candidate de Lyapunov comme suit :

$$V = \frac{1}{2} \mathbf{s}^T \mathbf{s} + V_{WNN} \quad (2.47)$$

Avec :

$$V_{WNN} = V_{F_{WNN}} + V_{G_{WNN}} \quad (2.48)$$

Et :

$$\begin{aligned} V_{F_{WNN}} &= \frac{1}{2\gamma_{W_F}} \text{tr}(\widetilde{\mathbf{W}}_F^T \widetilde{\mathbf{W}}_F) + \frac{1}{2\gamma_{C_F}} \text{tr}(\widetilde{\mathbf{C}}_F^T \widetilde{\mathbf{C}}_F) \\ V_{G_{WNN}} &= \frac{1}{2\gamma_{W_G}} \text{tr}(\widetilde{\mathbf{W}}_G^T \widetilde{\mathbf{W}}_G) + \frac{1}{2\gamma_{C_G}} \text{tr}(\widetilde{\mathbf{C}}_G^T \widetilde{\mathbf{C}}_G) \end{aligned} \quad (2.49)$$

La dérivée par rapport au temps la fonction candidate de Lyapunov (2.47) est donnée par:

$$\dot{V} = \mathbf{s}^T \dot{\mathbf{s}} + \dot{V}_{WNN} \quad (2.50)$$

Avec :

$$\dot{V}_{WNN} = \dot{V}_{F_{WNN}} + \dot{V}_{G_{WNN}} \quad (2.51)$$

Et :

$$\begin{aligned} \dot{V}_{F_{WNN}} &= \frac{1}{\gamma_{W_F}} \text{tr}(\widetilde{\mathbf{W}}_F^T \dot{\widetilde{\mathbf{W}}}_F) + \frac{1}{\gamma_{C_F}} \text{tr}(\widetilde{\mathbf{C}}_F^T \dot{\widetilde{\mathbf{C}}}_F) \\ \dot{V}_{G_{WNN}} &= \frac{1}{\gamma_{W_G}} \text{tr}(\widetilde{\mathbf{W}}_G^T \dot{\widetilde{\mathbf{W}}}_G) + \frac{1}{\gamma_{C_G}} \text{tr}(\widetilde{\mathbf{C}}_G^T \dot{\widetilde{\mathbf{C}}}_G) \end{aligned} \quad (2.52)$$

Alors, en utilisant (2.42), on trouve :

$$\dot{V} = -\mathbf{s}^T \left(-[\tilde{\mathbf{F}}_{WNN}(\mathbf{x}, \tilde{\mathbf{W}}_F, \tilde{\mathbf{C}}_F) + \boldsymbol{\varepsilon}_{F_{WNN}}(\mathbf{x})] - [\tilde{\mathbf{G}}_{WNN}(\mathbf{x}, \tilde{\mathbf{W}}_G, \tilde{\mathbf{C}}_G) + \boldsymbol{\varepsilon}_{G_{WNN}}(\mathbf{x})] \mathbf{u} - \mathbf{u}_r - \mathbf{d} + \Delta_{WNN} \right) + \dot{V}_{WNN} \quad (2.53)$$

En prenant en considération la relation (2.51) :

$$\begin{aligned} \dot{V} = & - \left[\mathbf{s}^T \left(\tilde{\mathbf{F}}_{WNN}(\mathbf{x}, \tilde{\mathbf{W}}_F, \tilde{\mathbf{C}}_F) + \boldsymbol{\varepsilon}_{F_{WNN}}(\mathbf{x}) \right) - \dot{V}_{F_{WNN}} \right] \\ & - \left[\mathbf{s}^T \left(\tilde{\mathbf{G}}_{WNN}(\mathbf{x}, \tilde{\mathbf{W}}_G, \tilde{\mathbf{C}}_G) + \boldsymbol{\varepsilon}_{G_{WNN}}(\mathbf{x}) \right) \mathbf{u} - \dot{V}_{G_{WNN}} \right] + \mathbf{s}^T (-\mathbf{u}_r - \mathbf{d} + \Delta_{WNN}) \end{aligned} \quad (2.54)$$

Donc, on reformule (2.54) comme suit:

$$\begin{aligned} \dot{V} = & - \left[\mathbf{s}^T \tilde{\mathbf{F}}_{WNN}(\mathbf{x}, \tilde{\mathbf{W}}_F, \tilde{\mathbf{C}}_F) - \dot{V}_{F_{WNN}} \right] - \left[\mathbf{s}^T \tilde{\mathbf{G}}_{WNN}(\mathbf{x}, \tilde{\mathbf{W}}_G, \tilde{\mathbf{C}}_G) \mathbf{u} - \dot{V}_{G_{WNN}} \right] \\ & + \mathbf{s}^T (-\boldsymbol{\varepsilon}_{F_{WNN}}(\mathbf{x}) - \boldsymbol{\varepsilon}_{G_{WNN}}(\mathbf{x}) \mathbf{u} - \mathbf{u}_r - \mathbf{d} + \Delta_{WNN}) \end{aligned} \quad (2.55)$$

On remplace par (2.41) dans (2.55) et on aura :

$$\begin{aligned} \dot{V} = & - \left[\left(\mathbf{s}^T (\tilde{\mathbf{W}}_F^T (\hat{\Psi}_F(\mathbf{x}, \hat{\mathbf{C}}_F) - \boldsymbol{\Omega}_F^T \hat{\mathbf{C}}_F) + \tilde{\mathbf{W}}_F^T \boldsymbol{\Omega}_F^T \tilde{\mathbf{C}}_F) - \dot{V}_{F_{WNN}} \right) \right] \\ & - \left[\left(\mathbf{s}^T (\tilde{\mathbf{W}}_G^T (\hat{\Psi}_G(\mathbf{x}, \hat{\mathbf{C}}_G) - \boldsymbol{\Omega}_G^T \hat{\mathbf{C}}_G) + \tilde{\mathbf{W}}_G^T \boldsymbol{\Omega}_G^T \tilde{\mathbf{C}}_G) \mathbf{u} - \dot{V}_{G_{WNN}} \right) \right] \\ & + \mathbf{s}^T (-\boldsymbol{\varepsilon}_{F_{WNN}}(\mathbf{x}) - \boldsymbol{\varepsilon}_{G_{WNN}}(\mathbf{x}) \mathbf{u} - \mathbf{u}_r - \mathbf{d} + \Delta_{WNN}) \end{aligned} \quad (2.56)$$

Donc, si on utilise (2.52), on obtient :

$$\begin{aligned} \dot{V} = & - \left[\left(\left(\mathbf{s}^T \tilde{\mathbf{W}}_F^T (\hat{\Psi}_F(\mathbf{x}, \hat{\mathbf{C}}_F) - \boldsymbol{\Omega}_F^T \hat{\mathbf{C}}_F) \right) + \frac{1}{\gamma_{W_F}} \text{tr} \left(\tilde{\mathbf{W}}_F^T \dot{\tilde{\mathbf{W}}}_F \right) \right) \right. \\ & \left. + \left(\mathbf{s}^T \tilde{\mathbf{W}}_F^T \boldsymbol{\Omega}_F^T \tilde{\mathbf{C}}_F + \frac{1}{\gamma_{C_F}} \text{tr} \left(\tilde{\mathbf{C}}_F^T \dot{\tilde{\mathbf{C}}}_F \right) \right) \right] \\ & - \left[\left(\mathbf{s}^T \tilde{\mathbf{W}}_G^T (\hat{\Psi}_G(\mathbf{x}, \hat{\mathbf{C}}_G) - \boldsymbol{\Omega}_G^T \hat{\mathbf{C}}_G) \mathbf{u} + \frac{1}{\gamma_{W_G}} \text{tr} \left(\tilde{\mathbf{W}}_G^T \dot{\tilde{\mathbf{W}}}_G \right) \right) \right. \\ & \left. + \left(\mathbf{s}^T \tilde{\mathbf{W}}_G^T \boldsymbol{\Omega}_G^T \tilde{\mathbf{C}}_G \mathbf{u} + \frac{1}{\gamma_{C_G}} \text{tr} \left(\tilde{\mathbf{C}}_G^T \dot{\tilde{\mathbf{C}}}_G \right) \right) \right] \\ & + \mathbf{s}^T (-\boldsymbol{\varepsilon}_{F_{WNN}}(\mathbf{x}) - \boldsymbol{\varepsilon}_{G_{WNN}}(\mathbf{x}) \mathbf{u} - \mathbf{u}_r - \mathbf{d} + \Delta_{WNN}) \end{aligned} \quad (2.57)$$

Ce qui est équivalent à :

$$\begin{aligned}
 \dot{V} = & \left[tr \left(\widetilde{\mathbf{W}}_F^T \left(\dot{\widetilde{\mathbf{W}}}_F - \gamma_{W_F} (\widehat{\Psi}_F(\mathbf{x}, \widehat{\mathbf{C}}_F) - \mathbf{\Omega}_F^T \widehat{\mathbf{C}}_F) \mathbf{s}^T \right) \right) + tr \left(\widetilde{\mathbf{C}}_F^T \left(\dot{\widetilde{\mathbf{C}}}_F - \gamma_{C_F} \mathbf{\Omega}_F \widehat{\mathbf{W}}_F \mathbf{s} \right) \right) \right] \\
 & + \left[tr \left(\widetilde{\mathbf{W}}_G^T \left(\dot{\widetilde{\mathbf{W}}}_G - \gamma_{W_G} (\widehat{\Psi}_G(\mathbf{x}, \widehat{\mathbf{C}}_G) - \mathbf{\Omega}_G^T \widehat{\mathbf{C}}_G) \mathbf{u} \mathbf{s}^T \right) \right) \right. \\
 & \left. + tr \left(\widetilde{\mathbf{C}}_G^T \left(\dot{\widetilde{\mathbf{C}}}_G - \gamma_{C_G} \mathbf{\Omega}_G \widehat{\mathbf{W}}_G \mathbf{s} \mathbf{u}^T \right) \right) \right] \\
 & + \mathbf{s}^T \left(-\boldsymbol{\varepsilon}_{FWNN}(\mathbf{x}) - \boldsymbol{\varepsilon}_{GWNN}(\mathbf{x}) \mathbf{u} - \mathbf{u}_r - \mathbf{d} + \Delta_{WNN} \right)
 \end{aligned} \tag{2.58}$$

Avec $\dot{\widetilde{\mathbf{C}}}_F = -\dot{\widehat{\mathbf{C}}}_F$, $\dot{\widetilde{\mathbf{C}}}_G = -\dot{\widehat{\mathbf{C}}}_G$, $\dot{\widetilde{\mathbf{W}}}_F = -\dot{\widehat{\mathbf{W}}}_F$, $\dot{\widetilde{\mathbf{W}}}_G = -\dot{\widehat{\mathbf{W}}}_G$. Par conséquent, en utilisant (2.43)-(2.46), on obtient:

$$\dot{V} = \mathbf{s}^T \left(-\boldsymbol{\varepsilon}_{FWNN}(\mathbf{x}) - \boldsymbol{\varepsilon}_{GWNN}(\mathbf{x}) |\mathbf{u}| - \mathbf{u}_r - \mathbf{d} + \Delta_{WNN} \right) \tag{2.59}$$

En utilisant **les hypothèses 1 et 5**, l'expression (2.59) est délimitée par:

$$\dot{V} \leq \mathbf{s}^T \left(\bar{\boldsymbol{\varepsilon}}_{FWNN} + \bar{\boldsymbol{\varepsilon}}_{GWNN} |\mathbf{u}| - K \operatorname{sgn}(\mathbf{s}) + D + \Gamma_{WNN} \right) \tag{2.60}$$

$$\dot{V} \leq \|\mathbf{s}^T\| \left(\bar{\boldsymbol{\varepsilon}}_{FWNN} + \bar{\boldsymbol{\varepsilon}}_{GWNN} |\mathbf{u}| - K + \xi_{WNN} \right) \tag{2.61}$$

Étant donné que K et $\xi_{WNN} = D + \Gamma_{WNN} + \bar{\boldsymbol{\varepsilon}}_{WNN} + \bar{\boldsymbol{\varepsilon}}_{WNN}$ sont sélectionnés par le choix du concepteur, K_{WNN} doit être choisi de telle sorte que $\dot{V} < 0$. Par conséquent, le système en boucle fermée est stable. Ainsi, tous les signaux sont bornés ce qui implique que l'erreur de poursuite \mathbf{e} et tous ses dérivés sont bornés, également, \mathbf{s} est bornée, c'est-à-dire, $\mathbf{s} \in L_\infty$.

L'intégration des deux côtés de (2.61) donne:

$$\int_0^t \|\mathbf{s}\| d\tau \leq \frac{1}{K - \xi_{WNN}} (|V(0)| + |V(t)|) \tag{2.62}$$

Ce qui signifie que $\mathbf{s} \in L_1$, en se basant sur le Lemme de Barbalat [95], on trouve que: $\lim_{t \rightarrow \infty} \|\mathbf{s}(t)\| = 0$, donc $\lim_{t \rightarrow \infty} \|\mathbf{e}(t)\| = 0$. Ainsi, l'erreur de poursuite convergera vers zéro quand $t \rightarrow \infty$.

4. Résultats de simulation

Un robot manipulateur à deux degrés de liberté de la Figure 2.3 se caractérise par les équations dynamiques du mouvement suivantes [79] :

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{c}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{g}(\mathbf{q}) + \boldsymbol{\tau}_c + \boldsymbol{\tau}_d = \mathbf{u} \quad (2.63)$$

$\mathbf{q}^T = [q_1 \quad q_2]$ est le vecteur des positions des articulations et $\dot{\mathbf{q}}^T = [\dot{q}_1 \quad \dot{q}_2]$ est le vecteur des vitesses articulaires.

$\mathbf{u}^T = [u_1 \quad u_2]$ est le vecteur de signaux de commandes appliqués au système.

$$\mathbf{M}(\mathbf{q}) = \begin{bmatrix} (m_1 + m_2)l_1^2 & m_1 l_1 l_2 \cos(q_1 - q_2) \\ m_2 l_1 l_2 \cos(q_1 - q_2) & m_2 l_2^2 \end{bmatrix}, \quad \mathbf{c}(\mathbf{q}, \dot{\mathbf{q}}) = \begin{bmatrix} m_2 l_1 l_2 \sin(q_1 - q_2) \dot{q}_2^2 \\ m_2 l_1 l_2 \sin(q_1 - q_2) \dot{q}_1^2 \end{bmatrix},$$

$$\mathbf{g}(\mathbf{q}) = \begin{bmatrix} -g(m_1 + m_2)l_1 \sin q_1 \\ -g m_2 l_2 \sin q_2 \end{bmatrix}, \quad \boldsymbol{\tau}_c = \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} \text{ and } \boldsymbol{\tau}_d = \begin{bmatrix} \text{sgn}(\dot{q}_1) \\ \text{sgn}(\dot{q}_2) \end{bmatrix}$$

où $\mathbf{M}(\mathbf{q})$ est une matrice positive d'inertie, $\mathbf{c}(\mathbf{q}, \dot{\mathbf{q}})$ et $\mathbf{g}(\mathbf{q})$ sont les effets de coriolis et les effets gravitationnels, respectivement. $\boldsymbol{\tau}_c$ and $\boldsymbol{\tau}_d$ représentent les incertitudes. La procédure détaillée pour la modélisation du robot est donnée selon l'équation de Lagrange dans [80].

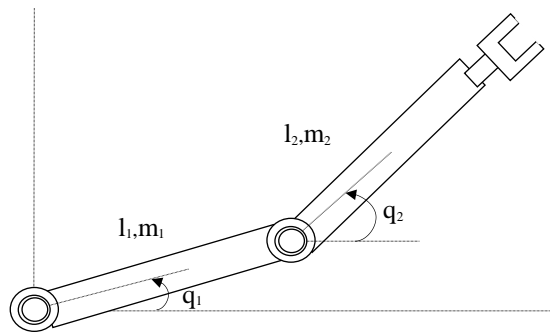


Figure 2.3 : Robot manipulateur

Les paramètres physiques du robot sont: $l_1 = l_2 = 0.432 \text{ m}$, $m_1 = 15.91 \text{ kg}$, $m_2 = 11.36 \text{ kg}$, $g = 9.81 \text{ m/sec}^2$.

Le vecteur d'état est donné par: $\mathbf{x}^T = [x_1 x_2 x_3 x_4] = [q_1 \dot{q}_1 q_2 \dot{q}_2]$ et le vecteur de sortie est donné par: $\mathbf{y}^T = [q_1 \quad q_2]$. On pose:

$$\mathbf{F}(\mathbf{x}) = \begin{bmatrix} f_1 \\ f_2 \end{bmatrix} = -\mathbf{M}^{-1}(\mathbf{q})(\mathbf{c}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{g}(\mathbf{q}) + \boldsymbol{\tau}_c),$$

$$\mathbf{G}(\mathbf{x}) = \mathbf{M}^{-1}(\mathbf{q}) \text{ et } \mathbf{d} = -\mathbf{M}^{-1}(\mathbf{q})\boldsymbol{\tau}_d.$$

Les paramètres de design sont choisis comme : $K_1 = K_2 = \text{diag}(8, 12)$, $K = 20I_2$, les paramètres de la surface de glissement sont sélectionnés à partir de [79] comme $\lambda = I_2$. Les conditions initiales du robot manipulateur sont données comme $(x_1, x_2, x_3, x_4) = (0.3, 1, 0.5, 0)$. Les gains d'adaptation sont choisis comme : $\gamma_{C_F} = 0.008$, $\gamma_{W_F} = 320$, $\gamma_{C_G} = 0.002$, $\gamma_{W_G} = 0.001$. Le WNN a 04 entrées qui sont les états du système représentées par le vecteur \mathbf{x} , deux sorties représentées par le vecteur \mathbf{y} et 81 nœuds cachés. Les paramètres de translation et de dilatation du WNN sont initialisés à l'aide de la technique de partitionnement de grille comme $\mathbf{c} = (-0.5, 0, 0.5)$ pour $k=1, \dots, n$, tandis que les poids de sortie sont choisis arbitrairement dans l'intervalle $[-1, 1]$. Les paramètres de translation et les poids de sortie du WNN sont adaptés en ligne pour assurer la poursuite de la trajectoires désirées $\mathbf{y}_d^T = [\sin(t) \quad \cos(t)]$.

Deux WNNs sont synthétisés pour approximer les fonctions inconnues $\mathbf{F}(\mathbf{x})$ et $\mathbf{G}(\mathbf{x})$ du système. Leurs approximations sont utilisées dans la synthèse du contrôleur adaptatif indirect selon (2.24). Les performances de poursuite dans des conditions nominales pour la première articulation y_1 et la deuxième articulation y_2 sont données aux Figure 2.4 et Figure 2.5, respectivement. Bien que les Figure 2.6 et 2.7 montrent les signaux de commande pour les deux articulations, et les erreurs de poursuite sont illustrées dans la Figure 2.8 pour la première articulation et la Figure 2.9 pour la deuxième articulation.

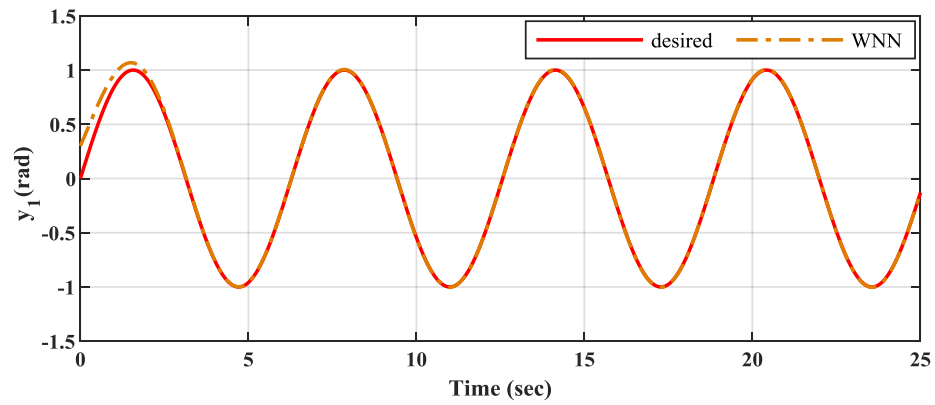


Figure 2.4 : Performance de poursuite de y_1 d'une trajectoire désirée y_{d1} dans le cas de contrôleur WNN

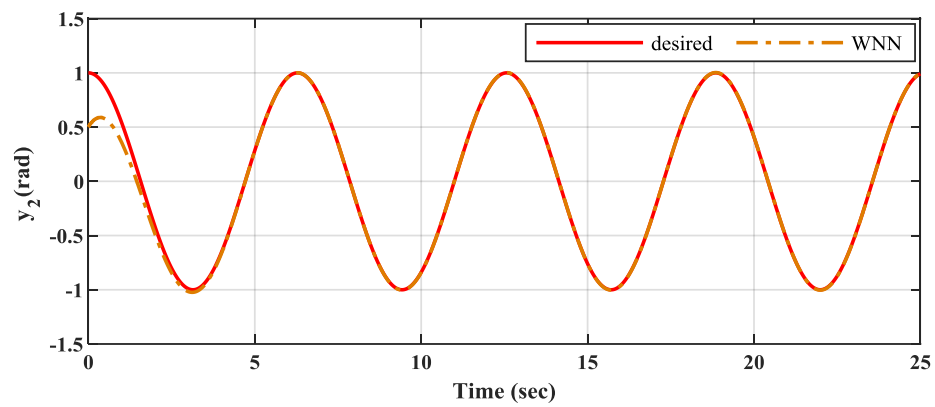


Figure 2.5 : Performance de poursuite de y_2 d'une trajectoire désirée y_{d2} dans le cas de contrôleur WNN

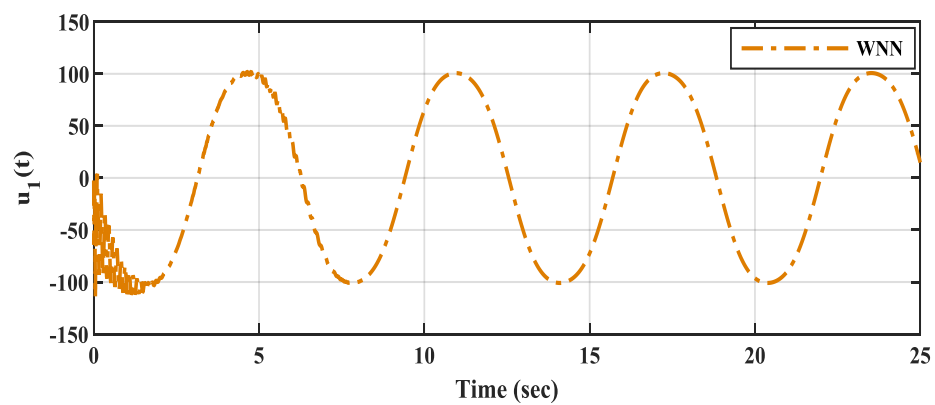
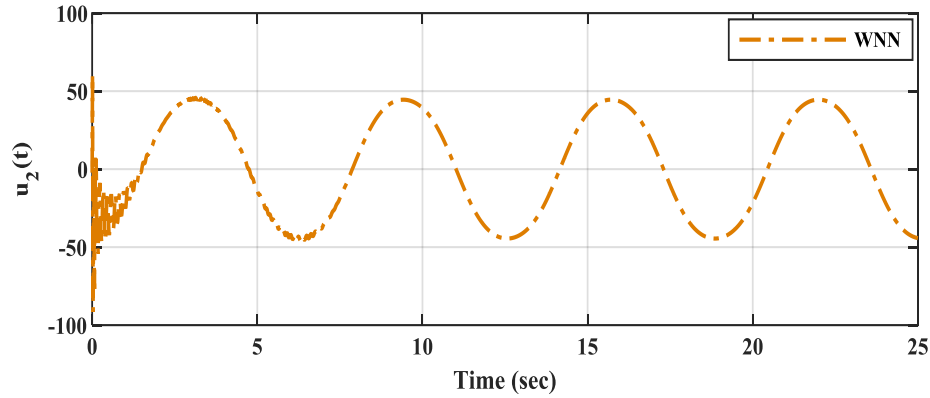
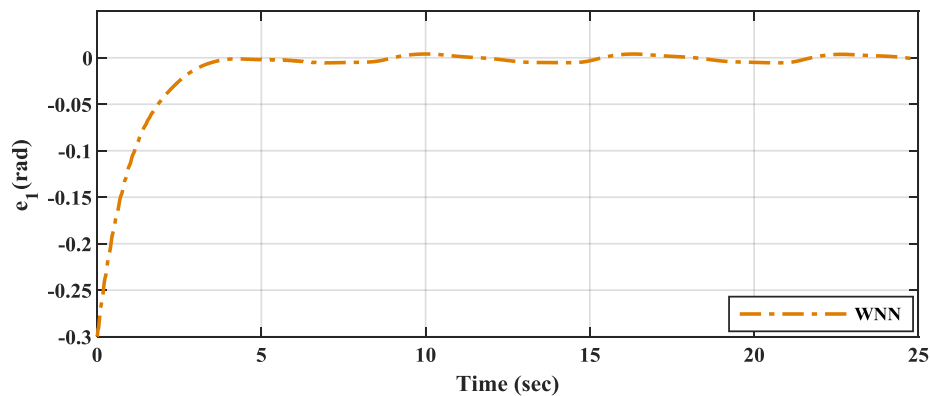


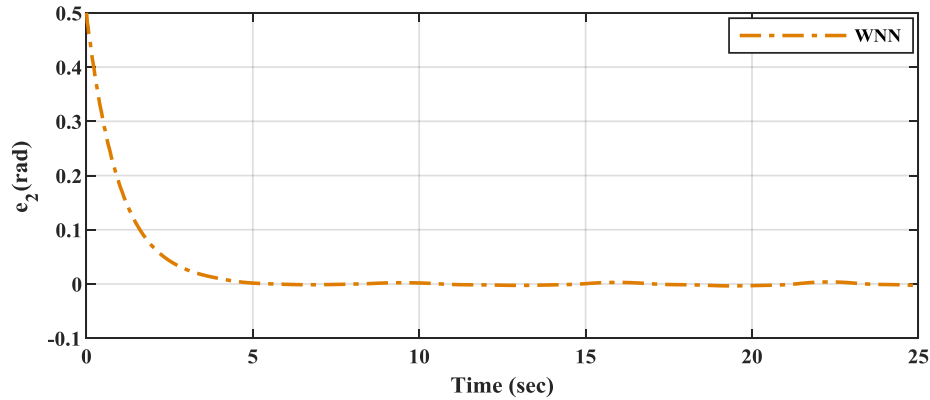
Figure 2.6 : Signal de commande u_1 du contrôleur WNN


 Figure 2.7 : Signal de commande u_2 dans le cas du contrôleur WNN

La Figure 2.4 et la Figure 2.5 montrent les performances de poursuite des sorties du système y_1 et y_2 dans le cas de contrôle d'un robot manipulateur à deux degrés de liberté par un contrôleur WNN adaptatif indirect, c.-à-d., le contrôleur basé sur l'approximation des fonctions du système par des réseaux WNNs. Le contrôleur a pu assurer des bonnes performances de poursuite, où les sorties du système y_1 et y_2 suivaient avec précision les trajectoire désirée y_{d_1} et y_{d_2} , respectivement.

Les signaux de commande du contrôleur WNN sont donnés dans la Figure 2.6 pour la première articulation et dans la Figure 2.7 pour la deuxième articulation. Le contrôleur synthétisé à la base des approximateurs WNNs montre des signaux de commande oscillatoires dans la phase transitoire qui disparaissent progressivement. Ces oscillations sont dues à l'effet de l'adaptation des paramètres des approximateurs WNNs, où le mécanisme d'adaptation essaye de minimiser l'écart entre la référence et la sortie du système en ajustant les centres des fonctions d'activation et les poids de sortie des WNNs qui sont sélectionnées arbitrairement au départ.


 Figure 2.8 : Erreur de poursuite e_1 dans le cas du contrôleur WNN

Figure 2.9 : Erreur de poursuite e_2 dans le cas du contrôleur WNN

La Figure 2.8 et la Figure 2.9 montrent les erreurs de poursuite du contrôleur WNN pour la première et la deuxième articulation, respectivement. D'après ces Figures, on peut remarquer que les erreurs de poursuite sont faibles. Ce qui signifie que les sorties des systèmes suivent avec précision les trajectoires désirées et cela est due à la bonne approximation des fonctions inconnues du système grâce à l'adaptation des centres et des poids de sortie des approximateurs WNNs.

5. Conclusion

Les réseaux de neurones artificielles à base d'ondelettes WNNs ont été souvent appliqués dans l'identification et la commande des systèmes non linéaires incertains. Ils possèdent la propriété d'approximation universelle pour tout fonction non linéaire. On a présenté dans ce chapitre les aspects théorique et numérique du contrôle neuronal adaptatif indirect à base d'ondelettes des systèmes MIMO non linéaires incertains. Où on a commencé par définir qu'est-ce qu'un réseau de neurones MIMO à base d'ondelettes et on a détaillé sa structure. Par la suite, on a exploité ce réseau WNN pour approximer les dynamiques inconnues d'un système MIMO incertain. Puis, on a utilisé les approximations obtenues des dynamiques du système pour développer un contrôleur adaptatif neuronal indirect pour commander le système. Les lois d'adaptation des centres et des poids de sortie du réseau WNN sont obtenus en utilisant la méthode du Lyapunov pour assurer la stabilité en boucle fermée du système de contrôle. Le contrôleur développé a été utilisé pour la commande d'un robot manipulateur incertain qui a deux entrées et deux sorties. Les résultats obtenus par le contrôleur WNNs dans la commande des systèmes MIMO avec ont montré une très bonne poursuite des états de systèmes de la trajectoire désirée. Le contrôleur basé sur le réseau WNN a pu réaliser une erreur de poursuite minimale mais le signal de commande n'est pas assez

lisse spécialement si l'écart entre la référence désirée et la sortie du système est grand, ou on peut avoir des pics considérables et des fortes oscillations. Ces oscillations sont corrigées par l'action rapide de mécanisme d'adaptation à ajuster les différents paramètres du réseau WNN (translation, dilatation, poids de sortie, ... etc.). Cependant, la représentation d'un modèle du système n'est pas unique. Un autre système intelligent peut générer une approximation différente des dynamiques du système. Pour cette raison, on va étudier les systèmes flous et leurs avantages dans l'approximation des systèmes incertains. Et on compare leurs performances à celles des systèmes WNNs dans le chapitre suivant.

Chapitre 3

Commande adaptative floue indirecte à base d'ondelettes des systèmes MIMO incertains

1. Introduction

Au cours des dernières décennies, l'utilisation des systèmes flous dans des problèmes de commande a attiré un grand intérêt. L'intérêt majeur des systèmes flous est leur capacité à gérer l'inexactitude et l'ambiguïté de l'information [81]. Les systèmes flous possèdent l'aptitude de l'être humain à prendre des décisions et d'agir d'une manière corrective malgré la nature floue des connaissances disponibles. Les systèmes flous permettent d'automatiser la prise de décision et la construction des systèmes artificiels effectuant les tâches habituellement prises en charge par les humains [82]. Davantage, Il est facile d'incorporer un système flou dans une boucle d'identification ou de commande.

L'utilisation des contrôleurs flous fondés sur l'expertise humaine présente une alternative au contrôle des systèmes par les réseaux de neurones artificiels. Ils possèdent l'avantage de compenser les incertitudes et les non linéarités négligées lors de la modélisation du système pour l'obtention de meilleurs résultats. L'implémentation de contrôle flou dans des domaines tels que l'environnement, l'automatisme, la robotique, la gestion de la circulation routière, la médecine et le transport a montré son efficacité et sa robustesse et devenu un sujet d'actualité de recherche dans le domaine de la commande intelligente des systèmes.

L'objectif des chercheurs est de rendre systématique la conception du contrôleur flou et le réglage de ses paramètres. Le contrôle adaptatif direct ou indirect associé à la théorie de stabilité de Lyapunov présente une solution promotrice pour résoudre le problème de réglage des paramètres dans la synthèse du contrôleur flou tout en garantissant la stabilité du système de commande. La structure de base d'un système flou est répartie en 03 parties principales [83]:

Fuzzification

Le savoir-faire de l'expert humain s'exprime avec des mots. Les divers modes de fonctionnement doivent être décrits à l'aide des symboles linguistiques. La fuzzification est l'opération de conversion d'une variable physique quantitative en une variable linguistique qualitative. Il s'agit de projeter la variable physique sur les ensembles flous définissant cette variable. Cette variable appartiendra donc, partiellement et avec différents degrés, à plusieurs fonctions d'appartenance qui constituent l'ensemble flou. Cette procédure permet de déterminer avec précision le degré d'appartenance de la variable d'entrée à chaque fonction d'appartenance. Les fonctions d'appartenance sont réparties sur l'univers de discours et leurs paramètres sont soigneusement choisis pour assurer des bonnes performances du système flou. Donc le choix des fonctions d'appartenance et la sélection de leurs paramètres est d'une grande importance dans la conception du système flou et est lié directement aux performances du système flou.

Inférence flou :

C'est la transformation des connaissances de l'expert humain sur le système en un ensemble de règles floues de la forme suivante :

Si prémisse Alors conclusion

La prémisse est un ensemble de conditions reliées par des opérateurs flous de type « et, ou » et représentent les conditions de fonctionnement du procédé. La partie conclusion peut être une action de commande ou une description d'évolution des états du système dans le cas d'identification.

Défuzzification :

La commande nécessite une grandeur numérique, il faudra donc convertir la sortie floue du moteur d'inférence en une valeur précise. Cette opération est appelée défuzzification. Parmi les méthodes de défuzzification les plus utilisées on trouve la méthode du centre de

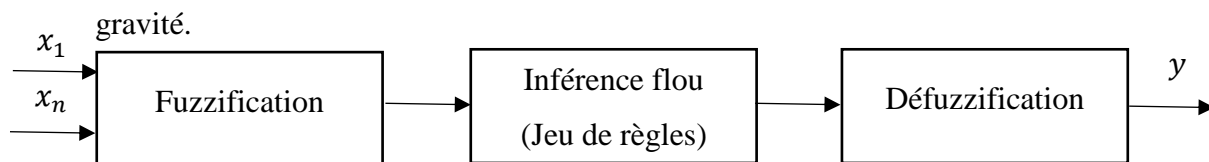


Figure 3.1 : Procédure de conception d'un système flou

Les systèmes flous sont des approximateurs universels et ont prouvé leur capacité dans l'approximation et la commande des systèmes incertains. Dans ce chapitre, on présente la conception d'un contrôleur flou adaptatif indirect à base d'ondelettes pour la commande des systèmes MIMO non linéaires incertains. On construit d'abord un approximateur flou MIMO avec des fonctions d'appartenance ondelettes qu'on a sélectionné dans le chapitre un. On favorise l'utilisation des ondelettes comme des fonctions d'appartenances car elles ont prouvé leur supériorité dans l'identification et le control par rapport aux fonctions d'appartenance non linéaires conventionnelles [84]. On utilise par la suite ce système flou pour approximer les dynamiques incertaines d'un système dynamique MIMO. Les approximations obtenues sont exploitées dans la conception d'un contrôleur flou adaptatif. On applique ce contrôleur flou pour contrôler un Robot manipulateur MIMO incertain par simulation et on compare ses performances aux performances du système WNN présenté dans le chapitre deux. L'objectif de cette étude est d'analyser les avantages de chaque approximateur et d'examiner la lucidité d'associer les deux approximateurs dans un nouveau schéma de commande.

2. Système flou MIMO

Ces dernières années, plusieurs approches systématiques de conception des contrôleurs flous ont été proposées. Dans ce chapitre on va présenter le design et le fonctionnement des contrôleurs flous adaptatifs indirects à base d'ondelettes partant de la théorie de stabilité de Lyapunov pour assurer la compensation en temps réel de l'erreur d'approximation des systèmes MIMO incertains.

Un système flou est un approximateur universel qui approxime les dynamiques d'un système physique en utilisant les connaissances linguistiques disponibles sur le comportement du système. Un système flou MIMO est généralement construit à partir d'un ensemble de règles SI-ALORS de la forme suivante:

R^l : SI x_1 est A_1^l et x_2 est A_2^l et ... et x_n est A_n^l ALORS h_{FS1} est θ_{l1} , h_{FS2} est θ_{l2} , ..., h_{FSm} est θ_{lm} .

Avec $\mathbf{x}^T = [x_1 x_2 \dots x_n] \in R^n$ est le vecteur d'entrée du system flou. h_{FSj} est la $j^{\text{ème}}$ sortie du system flou avec $j=1 \dots m$ et m est le nombre totale des sorties. θ_{lj} est un poids ajustable qui connecte la sortie h_{FSj} avec la règle R_l de $l=1, \dots, L$ et L est le nombre total des règles. A_k^l sont des

fonctions d'appartenance Gaussiennes : $A_k^l(x_k) = e^{-\frac{1}{2} \left(\frac{x_k - p_{kl}^1}{p_{kl}^2} \right)^2}$, p_{kl}^1 et p_{kl}^2 sont les centres et les

dilatations des fonctions d'appartenance A_k^l par rapport à l'entrée x_k . Le degré de validité de la règle R^l est donné par $\mu_l(\mathbf{x}) = \prod_{k=1}^n A_k^l(x_k)$, tant que les fonctions de base floues sont données par la normalisation de degré de validité de chaque règle comme $\varphi_l(\mathbf{x}) = \frac{\mu_l(\mathbf{x})}{\sum_{l=1}^L \mu_l(\mathbf{x})}$. La structure d'un FS MIMO est démontrée dans la Figure 3.2 [85].

La $j^{\text{ème}}$ sortie du FS est calculée comme suit:

$$h_{FS_j}(\mathbf{x}) = \sum_{l=1}^L \theta_{lj} \varphi_l(\mathbf{x}) = \boldsymbol{\theta}_j^T \Phi(\mathbf{x}) \quad (3.1)$$

Avec $\Phi^T = [\varphi_1 \varphi_2 \dots \varphi_L] \in R^L$ est un vecteur des fonctions des bases floues, $\boldsymbol{\theta}_j^T = [\theta_{1j} \theta_{2j} \dots \theta_{Lj}] \in R^L$ est un vecteur des poids θ_{lj} connectant la règle l avec la sortie h_{FS_j} .

La sortie globale du FS MIMO de la Figure 3.2 est donnée par:

$$h_{FS} = \begin{bmatrix} h_{FS_1} \\ h_{FS_2} \\ \vdots \\ h_{FS_m} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\theta}_1^T \Phi \\ \boldsymbol{\theta}_2^T \Phi \\ \vdots \\ \boldsymbol{\theta}_m^T \Phi \end{bmatrix} = \Theta^T \Phi(\mathbf{x}) \quad (3.2)$$

avec $\Theta^T = [\boldsymbol{\theta}_1 \boldsymbol{\theta}_2 \dots \boldsymbol{\theta}_m] \in R^{L \times m}$.

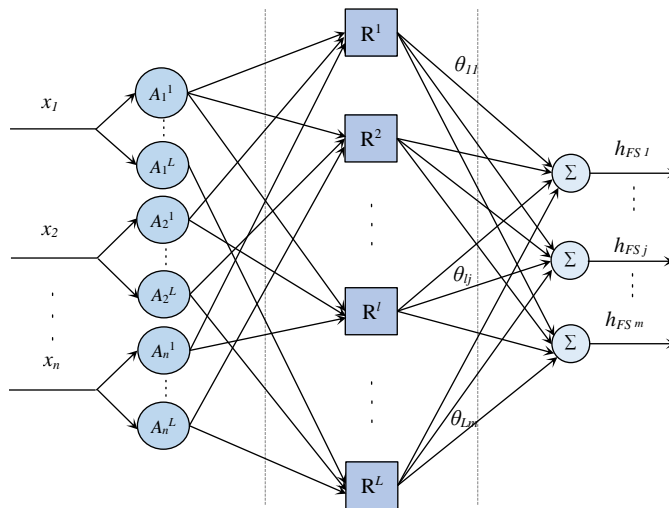


Figure 3.2 :Système flou MIMO

Le théorème d'approximation universel démontre que toute fonction continue peut-être approximée sur un univers compact avec des erreurs d'approximation finies. Se référant au théorème, théoriquement, il existe Θ^* une matrice des paramètres optimaux de Θ pour approximer $h_{FS}(\mathbf{x}) \in R^m$ et minimiser le vecteur d'erreur d'approximation $\boldsymbol{\varepsilon}_{FS}$:

$$h_{FS}(\mathbf{x}) = \Theta^{*T} \Phi(\mathbf{x}) + \boldsymbol{\varepsilon}_{FS} \quad (3.3)$$

Hypothèse 7:

Vecteur d'erreur d'approximation $\boldsymbol{\varepsilon}_{FS}$ est borné par $\boldsymbol{\varepsilon}_{FS} \leq \bar{\boldsymbol{\varepsilon}}_{FS}$, avec $\bar{\boldsymbol{\varepsilon}}_{FS}$ une constante positive inconnue. Cette hypothèse résulte du théorème d'approximation universel.

Les fonctions du système $F(\mathbf{x})$ et $G(\mathbf{x})$ peuvent être approximées à l'aide des FS MIMO comme suit:

$$\begin{aligned} F(\mathbf{x}) &= F_{FS}^*(\mathbf{x}, \Theta_F^*) + \boldsymbol{\varepsilon}_{F_{FS}}(\mathbf{x}) \\ &= \Theta_F^{*T} \Phi_F(\mathbf{x}) + \boldsymbol{\varepsilon}_{F_{FS}}(\mathbf{x}) \end{aligned} \quad (3.4)$$

$$\begin{aligned} G(\mathbf{x}) &= G_{FS}^*(\mathbf{x}, \Theta_G^*) + \boldsymbol{\varepsilon}_{G_{FS}}(\mathbf{x}) \\ &= \Theta_G^{*T} \Phi_G(\mathbf{x}) + \boldsymbol{\varepsilon}_{G_{FS}}(\mathbf{x}) \end{aligned} \quad (3.5)$$

Pratiquement, les paramètres idéaux, Θ_F^* et Θ_G^* , ne peuvent pas être déterminés. Leurs estimations $\hat{\Theta}_F$ et $\hat{\Theta}_G$ sont utilisées. Par conséquent, les fonctions inconnues du système sont approximées comme suit:

$$\hat{F}_{FS}(\mathbf{x}, \hat{\Theta}_F) = \hat{\Theta}_F^T \Phi_F(\mathbf{x}) \quad (3.6)$$

$$\hat{G}_{FS}(\mathbf{x}, \hat{\Theta}_G) = \hat{\Theta}_G^T \Phi_G(\mathbf{x}) \quad (3.7)$$

Avec :

$$\begin{aligned} F_{FS}^*(\mathbf{x}, \Theta_F^*) &= \hat{F}_{FS}(\mathbf{x}, \hat{\Theta}_F) + \tilde{F}_{FS}(\mathbf{x}, \tilde{\Theta}_F) \\ &= \hat{\Theta}_F^T \Phi_F(\mathbf{x}) + \tilde{\Theta}_F^T \Phi_F(\mathbf{x}) \end{aligned} \quad (3.8)$$

$$\begin{aligned} G_{FS}^*(\mathbf{x}, \Theta_G^*) &= \hat{G}_{FS}(\mathbf{x}, \hat{\Theta}_G) + \tilde{G}_{FS}(\mathbf{x}, \tilde{\Theta}_G) \\ &= \hat{\Theta}_G^T \Phi_G(\mathbf{x}) + \tilde{\Theta}_G^T \Phi_G(\mathbf{x}) \end{aligned} \quad (3.9)$$

Donc, les erreurs d'approximation des paramètres sont données comme suit: $\tilde{\Theta}_F = \Theta_F^* - \hat{\Theta}_F$ and $\tilde{\Theta}_G = \Theta_G^* - \hat{\Theta}_G$.

3. Commande adaptative indirecte par systèmes flous MIMO:

Considérant la classe des systèmes MIMO non linéaires incertains représentés par l'équation différentielle (2,11). En admettant toujours que les fonctions non linéaires du système $F(\mathbf{x})$ et $G(\mathbf{x})$ sont inconnues, $F(\mathbf{x})$ et $G(\mathbf{x})$ sont approximées par des approximateurs universel flous FSs de la Figure 3.1, leurs expressions sont données par les équations (3.6) et (3.7). Donc la loi de commande neuronale adaptative à base d'ondelettes est synthétisée comme suit :

On reprend l'équation (2.17) et on rajoute et on soustraire $\hat{G}_{FS}(\mathbf{x}, \hat{\Theta}_G)\mathbf{u}$, on aura:

$$\dot{\mathbf{s}} = \mathbf{v} - F(\mathbf{x}) - \left(G(\mathbf{x}) - \hat{G}_{FS}(\mathbf{x}, \hat{\Theta}_G) \right) \mathbf{u} - \hat{G}_{FS}(\mathbf{x}, \hat{\Theta}_G)\mathbf{u} - \mathbf{d} \quad (3.10)$$

Avec $\mathbf{v} = \mathbf{y}_d^{(r)} + \lambda_1 \mathbf{e}^{(r-1)} + \dots + \lambda_{r-2} \mathbf{e}^{(2)} + \lambda_{r-1} \dot{\mathbf{e}}$, on choisit la loi de commande comme suit:

$$\mathbf{u} = \hat{G}_{FS}(\mathbf{x}, \hat{\Theta}_G)^{-1} \left(\mathbf{v} + K \operatorname{sgn}(\mathbf{s}) - \hat{F}_{FS}(\mathbf{x}, \hat{\Theta}_F) \right) \quad (3.11)$$

Pour éviter le phénomène de chattering engendré par le terme glissant dans (3.11), une loi de commande par PI (proportionel-intégrateur) [32] donnée par (2.23) est employée. Donc, la loi de commande (3.11) devient:

$$\mathbf{u} = \hat{G}_{FS}(\mathbf{x}, \hat{\Theta}_G)^{-1} \left(\mathbf{v} + \mathbf{u}_r - \hat{F}_{FS}(\mathbf{x}, \hat{\Theta}_F) \right) \quad (3.12)$$

En remplaçant par (3.12) dans (3.10), on obtient:

$$\dot{\mathbf{s}} = - \left(F(\mathbf{x}) - \hat{F}_{FS}(\mathbf{x}, \hat{\Theta}_F) \right) - \left(G(\mathbf{x}) - \hat{G}_{FS}(\mathbf{x}, \hat{\Theta}_G) \right) \mathbf{u} - \mathbf{u}_r - \mathbf{d} \quad (3.13)$$

De même, en utilisant (3.4)-(3.5), on trouve :

$$\begin{aligned} \dot{\mathbf{s}} = & - \left(F_{FS}^*(\mathbf{x}, \Theta_F^*) + \varepsilon_{F_{FS}}(\mathbf{x}) - \hat{F}_{FS}(\mathbf{x}, \hat{\Theta}_F) \right) \\ & - \left(G_{FS}^*(\mathbf{x}, \Theta_G^*) + \varepsilon_{G_{FS}}(\mathbf{x}) - \hat{G}_{FS}(\mathbf{x}, \hat{\Theta}_G) \right) \mathbf{u} - \mathbf{u}_r - \mathbf{d} \end{aligned} \quad (3.14)$$

De (3.8)-(3.9), on trouve que:

$$\dot{s} = -\tilde{F}_{FS}(\mathbf{x}, \tilde{\Theta}_F) - \tilde{G}_{FS}(\mathbf{x}, \tilde{\Theta}_G)\mathbf{u} - \mathbf{u}_r - \mathbf{d} - \boldsymbol{\varepsilon}_{F_{FS}}(\mathbf{x}) - \boldsymbol{\varepsilon}_{G_{FS}}(\mathbf{x})\mathbf{u} \quad (3.15)$$

Pour atteindre les objectifs de contrôle, il est nécessaire d'ajuster les paramètres des approximateurs flous FSs pour avoir $\mathbf{e}(t) \rightarrow 0$ lorsque $t \rightarrow \infty$. Pour cela, les lois d'adaptation des paramètres des systèmes flous FSs sont données comme suit:

$$\dot{\hat{\Theta}}_F = -\gamma_{\Theta_F} \Phi_F(\mathbf{x}) \mathbf{s}^T \quad (3.16)$$

$$\dot{\hat{\Theta}}_G = -\gamma_{\Theta_G} \Phi_G(\mathbf{x}) \mathbf{u} \mathbf{s}^T \quad (3.17)$$

Avec $\gamma_{\Theta_F}, \gamma_{\Theta_G}$ sont des gains positives d'adaptation.

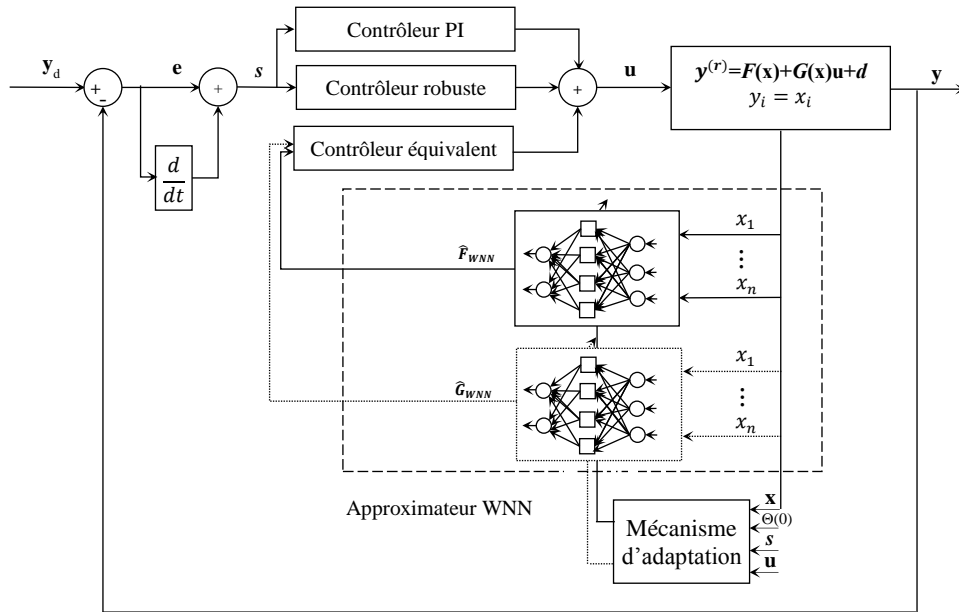


Figure 3.3 : Architecture d'une boucle de commande adaptative indirecte en utilisant des approximateurs FSs

Théorème 3:

Considérant la boucle fermée de commande de la Figure 3.2, avec le système MIMO non linéaire incertain (2.11). Le loi de commande floue adaptative est choisie comme (3.12) et les lois d'adaptation des paramètres sont données par (3.16)-(3.17). Le contrôleur à base des

approximateurs FSs adaptatifs garantit que le système de commande en boucle fermée est stable, et que tous les signaux sont bornés.

Preuve de Théorème 3:

On considère la fonction candidate de Lyapunov suivante:

$$V = \frac{1}{2} \mathbf{s}^T \mathbf{s} + V_{FS} \quad (3.18)$$

Avec :

$$V_{FS} = V_{FFS} + V_{GFS} \quad (3.19)$$

Et :

$$\begin{aligned} V_{FFS} &= \frac{1}{2\gamma_{\Theta_F}} \text{tr} \left(\tilde{\Theta}_F^T \tilde{\Theta}_F \right) \\ V_{GFS} &= \frac{1}{2\gamma_{\Theta_G}} \text{tr} \left(\tilde{\Theta}_G^T \tilde{\Theta}_G \right) \end{aligned} \quad (3.20)$$

La dérivée par rapport au temps la fonction candidate de Lyapunov (3.18) est donnée par:

$$\dot{V} = \mathbf{s}^T \dot{\mathbf{s}} + \dot{V}_{FS} \quad (3.21)$$

Avec :

$$\dot{V}_{FS} = \dot{V}_{FFS} + \dot{V}_{GFS} \quad (3.22)$$

Et :

$$\begin{aligned} \dot{V}_{FFS} &= \frac{1}{\gamma_{\Theta_F}} \text{tr} \left(\tilde{\Theta}_F^T \dot{\tilde{\Theta}}_F \right) \\ \dot{V}_{GFS} &= \frac{1}{\gamma_{\Theta_G}} \text{tr} \left(\tilde{\Theta}_G^T \dot{\tilde{\Theta}}_G \right) \end{aligned} \quad (3.23)$$

En substituant (3.15) dans (3.21), on aura:

$$\dot{V} = \mathbf{s}^T \left(-\tilde{\mathbf{F}}_{FS}(\mathbf{x}, \tilde{\Theta}_F) - \tilde{\mathbf{G}}_{FS}(\mathbf{x}, \tilde{\Theta}_G) \mathbf{u} - \mathbf{u}_r - \mathbf{d} - \boldsymbol{\varepsilon}_{FFS}(\mathbf{x}) - \boldsymbol{\varepsilon}_{GFS}(\mathbf{x}) \mathbf{u} \right) + \dot{V}_{FS} \quad (3.24)$$

Donc, en utilisant (3.22), (3.24) devient :

$$\dot{V} = \mathbf{s}^T \left(-\tilde{\mathbf{F}}_{FS}(\mathbf{x}, \tilde{\Theta}_F) - \tilde{\mathbf{G}}_{FS}(\mathbf{x}, \tilde{\Theta}_G) \mathbf{u} - \mathbf{u}_r - \mathbf{d} - \boldsymbol{\varepsilon}_{F_{FS}}(\mathbf{x}) - \boldsymbol{\varepsilon}_{G_{FS}}(\mathbf{x}) \mathbf{u} \right) + \dot{V}_{F_{FS}} + \dot{V}_{G_{FS}} \quad (3.25)$$

Réorganisant (3.25) comme suit :

$$\begin{aligned} \dot{V} = & -\left(\mathbf{s}^T \tilde{\mathbf{F}}_{FS}(\mathbf{x}, \tilde{\Theta}_F) - \dot{V}_{F_{FS}} \right) - \left(\mathbf{s}^T \tilde{\mathbf{G}}_{FS}(\mathbf{x}, \tilde{\Theta}_G) \mathbf{u} - \dot{V}_{G_{FS}} \right) \\ & + \mathbf{s}^T \left(-\mathbf{u}_r - \mathbf{d} - \boldsymbol{\varepsilon}_{F_{FS}}(\mathbf{x}) - \boldsymbol{\varepsilon}_{G_{FS}}(\mathbf{x}) \mathbf{u} \right) \end{aligned} \quad (3.26)$$

Alors, en utilisant (3.8)-(3.9), on aura :

$$\begin{aligned} \dot{V} = & -\left(\mathbf{s}^T \tilde{\Theta}_F^T \Phi_F(\mathbf{x}) - \dot{V}_{F_{FS}} \right) - \left(\mathbf{s}^T \tilde{\Theta}_G^T \Phi_G(\mathbf{x}) \mathbf{u} - \dot{V}_{G_{FS}} \right) \\ & + \mathbf{s}^T \left(-\mathbf{u}_r - \mathbf{d} - \boldsymbol{\varepsilon}_{F_{FS}}(\mathbf{x}) - \boldsymbol{\varepsilon}_{G_{FS}}(\mathbf{x}) \mathbf{u} \right) \end{aligned} \quad (3.27)$$

Aussi, en utilisant (3.23), on trouve :

$$\begin{aligned} \dot{V} = & -\left(\mathbf{s}^T \tilde{\Theta}_F^T \Phi_F(\mathbf{x}) - \frac{1}{\gamma_{\Theta_F}} \text{tr} \left(\tilde{\Theta}_F^T \dot{\tilde{\Theta}}_F \right) \right) - \left(\mathbf{s}^T \tilde{\Theta}_G^T \Phi_G(\mathbf{x}) \mathbf{u} - \frac{1}{\gamma_{\Theta_G}} \text{tr} \left(\tilde{\Theta}_G^T \dot{\tilde{\Theta}}_G \right) \right) \\ & + \mathbf{s}^T \left(-\mathbf{u}_r - \mathbf{d} - \boldsymbol{\varepsilon}_{F_{FS}}(\mathbf{x}) - \boldsymbol{\varepsilon}_{G_{FS}}(\mathbf{x}) \mathbf{u} \right) \end{aligned} \quad (3.28)$$

A la fin, on obtient:

$$\begin{aligned} \dot{V} = & \text{tr} \left(\tilde{\Theta}_F^T \left(\dot{\tilde{\Theta}}_F - \gamma_{\Theta_F} \Phi_F(\mathbf{x}) \mathbf{s}^T \right) \right) + \text{tr} \left(\tilde{\Theta}_G^T \left(\dot{\tilde{\Theta}}_G - \gamma_{\Theta_G} \Phi_G(\mathbf{x}) \mathbf{u} \mathbf{s}^T \right) \right) \\ & + \mathbf{s}^T \left(-\mathbf{u}_r - \mathbf{d} - \boldsymbol{\varepsilon}_{F_{FS}}(\mathbf{x}) - \boldsymbol{\varepsilon}_{G_{FS}}(\mathbf{x}) \mathbf{u} \right) \end{aligned} \quad (3.29)$$

Avec $\dot{\tilde{\Theta}}_F = -\dot{\tilde{\Theta}}_F$, $\dot{\tilde{\Theta}}_G = -\dot{\tilde{\Theta}}_G$.

De même, en utilisant (3.16)-(3.17), on obtient :

$$\dot{V} = \mathbf{s}^T \left(-\mathbf{u}_r - \mathbf{d} - \boldsymbol{\varepsilon}_{F_{FS}}(\mathbf{x}) - \boldsymbol{\varepsilon}_{G_{FS}}(\mathbf{x}) \mathbf{u} \right) \quad (3.30)$$

Et si on prend en considération l'hypothèse7:

$$\dot{V} \leq \mathbf{s}^T \left(-K \text{sgn}(\mathbf{s}) + D + \bar{\boldsymbol{\varepsilon}}_{F_{FS}} + \bar{\boldsymbol{\varepsilon}}_{G_{FS}} |\mathbf{u}| \right) \quad (3.31)$$

$$\dot{V} \leq \|\mathbf{s}^T\| \left(-K + \xi_{FS} \right) \quad (3.32)$$

Comme K et $\xi_{FS} = D + \bar{\mathbf{e}}_{FFS} + \bar{\mathbf{e}}_{GFS}|\mathbf{u}|$ sont sélectionnés par le concepteur, K doit être choisit de telle façon que $\dot{V} < 0$. Donc, le système en boucle fermée est stable et tous les signaux sont bornés ce qui implique que l'erreur de poursuite \mathbf{e} et tous ses dérivées sont bornées, de même, s'est bornée, c. à. d., $\mathbf{s} \in L_\infty$. Si on prend l'intégral des de côtés de (3.32), on trouve:

$$\int_0^t \|\mathbf{s}\| d\tau \leq \frac{1}{K - \xi_{FS}} (|V(0)| + |V(t)|) \quad (3.33)$$

Ce qui signifie que $\mathbf{s} \in L_1$, en se basant sur le lemme de Barbalat, on conclut que : $\lim_{t \rightarrow \infty} \|\mathbf{s}(t)\| = 0$, alors $\lim_{t \rightarrow \infty} \|\mathbf{e}(t)\| = 0$. Donc, l'erreur de poursuite convergera vers 0 lorsquet $t \rightarrow \infty$.

4. Résultats de simulation

On utilise le modèle du robot manipulateur à deux degrés de liberté de l'équation (2,63) et la Figure 2.3. Les conditions initiales du robot manipulateur sont également choisies comme $(x_1, x_2, x_3, x_4) = (0.3, 1, 0.5, 0)$. Dans cette partie, Deux FSs sont synthétisés pour approximer les fonctions inconnues $\mathbf{F}(\mathbf{x})$ et $\mathbf{G}(\mathbf{x})$ du système. Leurs approximations sont utilisées dans la synthèse de la loi de commande du contrôleur flou adaptatif indirect données par (3.12). Les paramètres de design de la loi de commande (3.12) sont sélectionnés comme dans le cas du contrôleur WNN avec: $K_1 = K_2 = \text{diag}(8, 12)$, $K = 20I_2$. Les paramètres de la surface de glissement sont aussi sélectionnés à partir de [79] comme $\lambda = I_2$. Les gains d'adaptation des réseaux FSs sont choisis comme : $\gamma_{\Theta_F} = 600, \gamma_{\Theta_G} = 0.1$. Chaque système FS à 04 entrées, deux sorties et 81 règles. Pour chaque entrée x_i nous associons trois fonctions d'appartenance Gaussiennes, Leurs paramètres de translation et de dilatation sont initialisés avec la technique de partitionnement de grille comme $p_{ik}^1 = (-1, 0, 1)$ et $p_{ik}^2 = 1.05$ for $k=1, \dots, n$, tandis que les poids de sortie des FSs sont choisis arbitrairement dans l'intervalle $[-1, 1]$ et adaptés en ligne pour assurer la poursuite de la trajectoires désirées $\mathbf{y}_d^T = [\sin(t) \quad \cos(t)]$.

Les performances de poursuite dans des conditions nominales pour la première articulation y_1 et la deuxième articulation y_2 sont données aux Figure 3.3 et Figure 3.4, respectivement. Bien que les Figure 3.5 et 3.6 montrent les signaux de commande pour les deux articulations, et les erreurs

de poursuite sont illustrées dans la Figure 3.7 pour la première articulation et la Figure 3.8 pour la deuxième articulation.

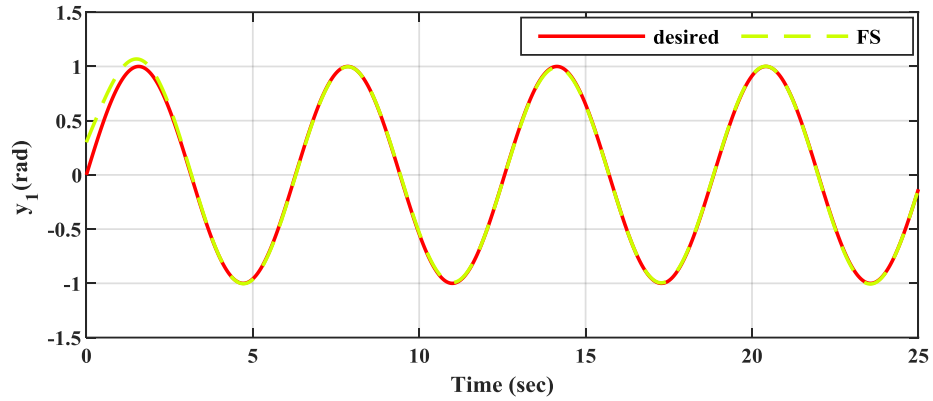


Figure 3.4 : Performance de poursuite de y_1 d'une trajectoire désirée y_{d1} dans le cas du contrôleur FS

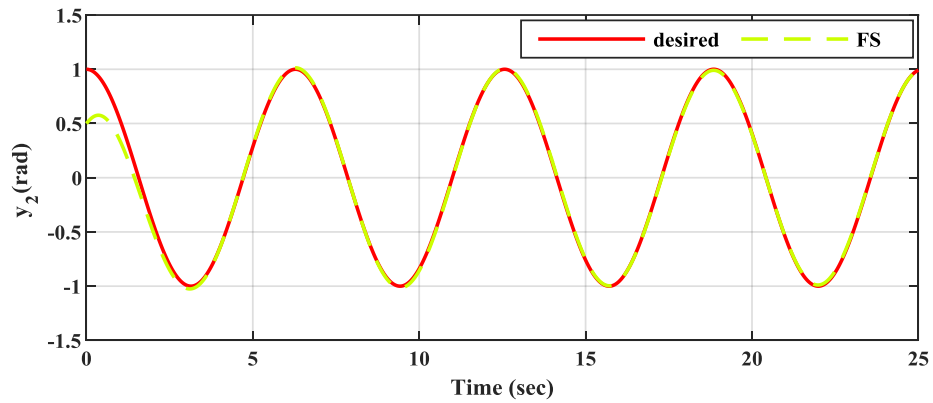


Figure 3.5 : Performance de poursuite de y_2 d'une trajectoire désirée y_{d2} dans le cas de contrôleur FS

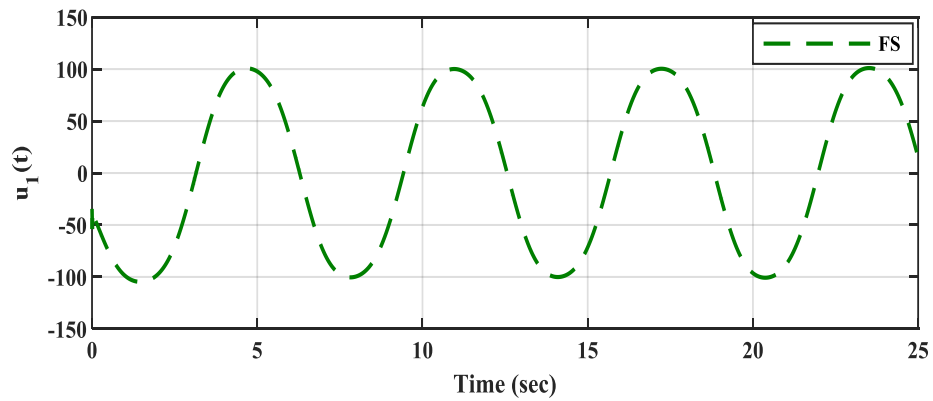
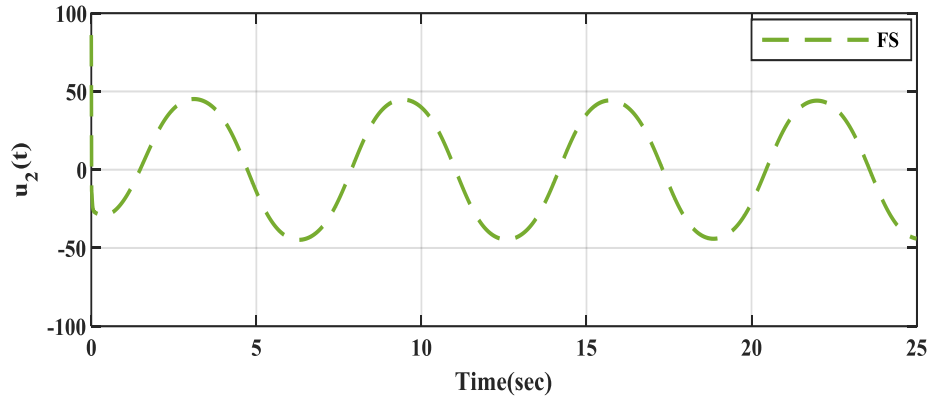
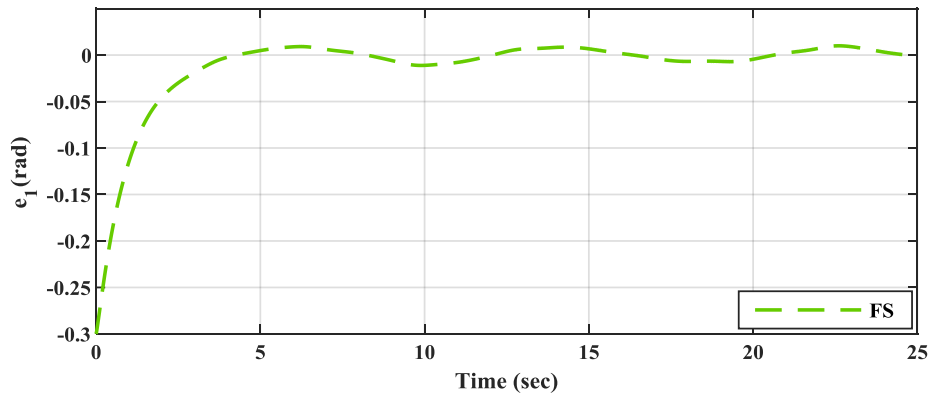


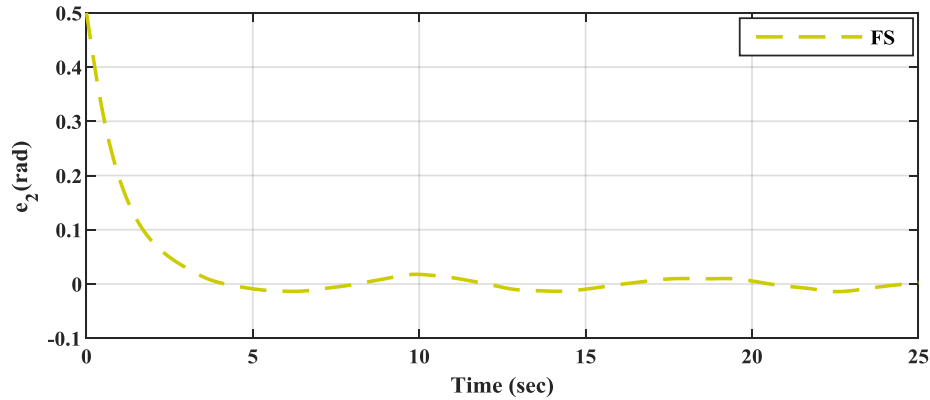
Figure 3.6 : Signal de commande u_1 du contrôleur FS

Figure 3.7 : Signal de commande u_2 dans le cas du contrôleur FS

La Figure 3.3 et la Figure 3.4 montrent les performances de poursuite des sorties du système y_1 et y_2 dans le cas de contrôle du robot manipulateur à deux degrés de liberté par un contrôleur FS adaptatif indirect, c.-à-d., le contrôleur basé sur l'approximation des fonctions du système par des réseaux flous FSs. Le contrôleur a assuré une bonne poursuite des sorties du système y_1 et y_2 des trajectoires de référence y_{d1} et y_{d2} , respectivement.

Les signaux de commande du contrôleur FS sont donnés dans la Figure 3.5 pour la première articulation et dans la Figure 3.6 pour la deuxième articulation. Le contrôleur synthétisé à la base des approximateurs FSs montre des signaux de commande lisses. Malgré que les paramètres de design des approximateurs FSs ont été sélectionné arbitrairement. Contrairement au cas du contrôleur WNN développé dans le chapitre précédent.

Figure 3.8 : Erreur de poursuite e_1 dans le cas du contrôleur FS

Figure 3.9 : : Erreur de poursuite e_2 dans le cas du contrôleur FS

La Figure 3.7 et la Figure 3.8 montrent les erreurs de poursuite du contrôleur FS pour la première et la deuxième articulation, respectivement. D'après ces Figures, on peut remarquer que les erreurs de poursuite sont un peu plus importantes au départ et se diminuent lentement et progressivement. Cela est justifié par le fait que le mécanisme d'adaptation essaye de corriger l'écart entre la référence et la sortie du système doucement et délicatement en jouant sur le degré d'activation des règles floues et leurs contributions dans la prise de décision de contrôle (inférence floue). Ce qui est bénéfique pour le contrôle des systèmes réels car il évite les changements brusques et les pics dans le signal de commande qui peuvent endommager l'actionneur.

5. Conclusion

Dans ce chapitre, on a présenté la technique de contrôle flou adaptatif indirect des systèmes MIMO non linéaires incertains. D'abord, on a défini le système flou MIMO et on a donné une présentation détaillée de son architecture. Par la suite, on a présenté la procédure de synthèse du contrôleur flou adaptatif indirect en partant de l'étape de l'approximation des dynamiques inconnues du système par des approximateurs FSs adaptatifs jusqu'à l'étape de synthèse du contrôleur. Les lois d'adaptation des approximateurs FSs sont obtenues en utilisant la méthode de Lyapunov pour assurer la stabilité en boucle fermée du système de contrôle. Les performances du contrôleur flou ont été testées sur un robot manipulateur incertain à deux degrés de liberté par simulation. Les résultats obtenus ont démontré la capacité du contrôleur flou adaptatif indirect dans la commande des systèmes MIMO avec une bonne poursuite d'une trajectoire désirée. Le contrôleur flou a présenté une petite erreur de poursuite mais il a pu assurer un signal de commande lisse même en présence d'un grand écart entre la référence désirée et la sortie du

système au départ, ou on peut observer l'absence des pics et des oscillations contrairement aux WNNs. D'où on peut conclure la robustesse des systèmes flous.

En conclusion, les systèmes flous disposent de la propriété d'approximation universelle pour tout système dynamique incertain. En plus, les FSs sont plus robustes et moins sensibles aux variations paramétriques contrairement aux WNNs. Cependant, les WNNs ont plus de puissance de réajustement grâce à l'adaptation des paramètres de dilatation et de translation des fonctions d'activations dans les neurones cachés des réseaux WNNs . La complémentarité des deux approches a donc naturellement été étudiée dans la littérature. Une nouvelle structure hiérarchique pour l'intégration des deux approches différentes d'approximation sera présentée dans le chapitre suivant.

Dans le chapitre suivant, un nouveau approximateur FS-WNN sera développé et utilisé pour la commande des systèmes MIMO non linéaires incertains. Nous proposons alors une nouvelle hybridation entre l'approximateur WNN présenté dans le chapitre 02 avec l'approximateur FS présenté dans le chapitre 03 pour combiner les avantages des deux approximateurs.

Chapitre 4

Commande adaptative indirecte par FS-WNN hybride des systèmes MIMO incertains

1. Introduction

Dans les chapitres précédents, on a investigué les performances des systèmes flous et des réseaux de neurones à base d'ondelettes dans la commande adaptative des systèmes incertains. Les caractéristiques inhérentes des systèmes flous et des réseaux WNNs dans l'approximation des fonctions non linéaires ont été analysées et prouvées. Chacun de ces approximateurs possède des avantages individuels et aussi des limitations dans l'approximation et la commande des systèmes incertains. Néanmoins, une hybridation entre les deux approximateurs permet de combiner leurs avantages et surmonter leurs limitations. Les chercheurs ont proposé de combiner les systèmes flous et les réseaux de neurones à base d'ondelettes pour les raisons suivantes [86]:

- Les réseaux de neurones ont une grande capacité d'adaptation des paramètres (translation et dilatations des fonctions d'activation, poids d'entrée et de sortie, ... etc.).
- La qualité d'approximation est considérablement améliorée par cette intégration.
- Les limitations individuelles des deux systèmes intelligents sont contournées, par exemple, cette technique est efficace pour surmonter le problème des minimums locaux connus dans les applications des réseaux de neurones et des systèmes flous.

Ces différentes raisons ont conduit à l'élaboration d'un domaine de recherche désigné par "neuro-flou", cherchant à combiner les deux approches [87]. De nombreux travaux de recherches ont été effectués et diverses solutions ont été proposées dans ce domaine.

Dans la littérature, différents schémas de combinaisons des systèmes flous et des réseaux de neurones ont été proposés. La combinaison des systèmes intelligents la plus courante est le Adaptive Neuro-Fuzzy Inference System (ANFIS). ANFIS est basé sur le modèle flou de Takagi-Sugeno (T-S). Contrairement au modèle flou de T-S, ANFIS utilise les méthodes d'apprentissage,

développées à l'origine pour le design des NNs, afin d'ajuster les fonctions d'appartenances dans le but d'atteindre les performances désirées [88]. Cependant, ANFIS est construit avec un ensemble de règles floues qui ont des parties conséquences linéaires ou constantes [89], [90]. Chairez [91] a proposé de remplacer la partie linéaire ou constante de ANFIS par une combinaison de sous-NNs. Où, la partie résultante de chaque règle est composée d'une somme pondérée des sous-NNs et le résultat global est calculé en sommant les résultats de toutes les règles.

Les chercheurs comme Tsai et al. [92] ont tenté d'améliorer les performances des systèmes Neuro-flous en intégrant la théorie des ondelettes dans les systèmes Neuro-flous, et ils ont conçu un approximateur neuronal flou à base d'ondelettes (FWNN). Le réseau FWNN attribue à la partie conséquence de chaque règle floue de ANFIS un sous-WNN au lieu d'un sous-NN conventionnel comme dans ANFIS. L'intérêt de cette combinaison est de regrouper les avantages des systèmes flous avec les avantages des réseaux NNs et les caractéristiques inhérentes des ondelettes dans l'approximation des signaux [93]–[95] pour obtenir une meilleure identification des systèmes incertains.

En résumé Walia et al. [88], Qin et al. [89] et Chairez [91] ont fusionné un approximation FS avec des approximateurs NN/WNN dans un même réseau pour concevoir un seul approximateur qui combine leurs avantages comme le montre la Figure 4.1.

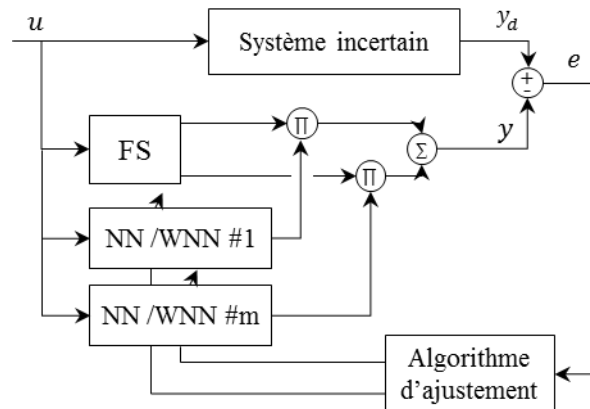


Figure 4.1 : Approximateur FWNN

D'autres travaux ont hybridé plusieurs approximateurs NNs ont été utilisés pour identifier les systèmes dynamiques afin d'obtenir un ensemble d'approximateurs intelligents plus précis, le résultat d'identification est calculé en prenant la moyenne des sorties des approximateurs NNs comme le montre la Figure 4.2.

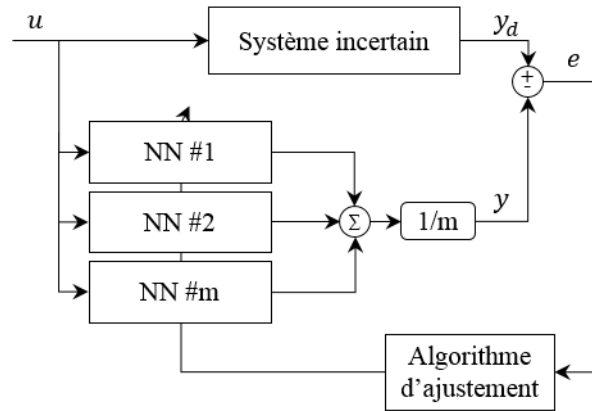


Figure 4.2 : Approximateur à base de NNs Multiples

Dans ce chapitre, on utilise une hybridation de deux systèmes intelligents différents pour approximer les fonctions inconnues du système. Le premier approximateur est un système flou FS et le deuxième approximateur est un approximateur WNN, et on combine les sorties des deux approximateurs en utilisant la technique de modulation qui a été proposée par Dian et al. [96]. Le principe de cette hybridation repose sur l'approximation des fonctions inconnues du système par les deux approximateurs (WNN et FS) séparément et simultanément. Ensuite, les deux approximations différentes de la même fonction sont combinées par un facteur de modulation pour améliorer la qualité d'approximation des dynamiques inconnues du système. La méthode directe de Lyapunov est utilisée pour dériver les lois d'adaptation des paramètres du FS-WNN et pour prouver la stabilité asymptotique du système de contrôle. Les résultats de la simulation sont donnés et une étude comparative est réalisée entre l'approximateur FS, l'approximateur WNN et l'approximateur FS-WNN hybride.

2. Approximateur FS-WNN hybride

Le nouveau approximateur FS-WNN hybride est le résultat de l'hybridation du système FS présenté dans le chapitre 03 avec le système WNN présenté dans le chapitre 02 comme le montre la Figure 4.3. Le FS et le WNN approximent indépendamment chaque fonction inconnue du système dynamique. Ensuite, les deux approximations de la même fonction sont combinées à l'aide de la technique de modulation. La technique de modulation attribue un degré d'importance α à l'approximation du FS et attribue son complément $(1-\alpha)$ à l'approximation du WNN de la même fonction. La somme de ces deux approximations pondérées donne une approximation hybride

optimale des fonctions inconnues. Les approximations obtenues des fonctions du système par le nouveau approximateur FS-WNN hybride rassemblent les avantages des deux approximateurs intelligents, le FS et le WNN.

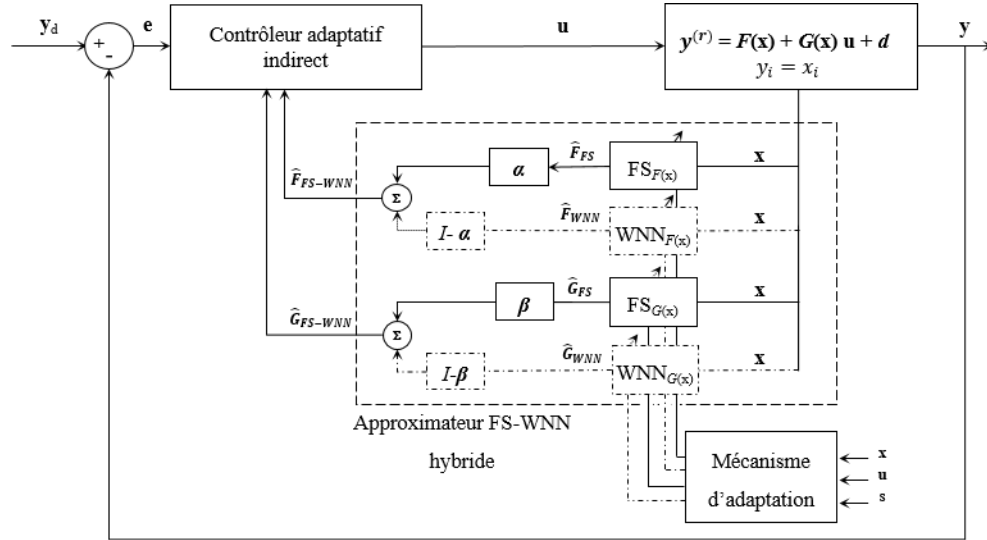


Figure 4.3 : Architecture de l'approximateur FS-WNN hybride dans une boucle de commande

L'approximation donnée par le FS-WNN hybride combine les avantages de FS et de WNN, elle rassemble l'intégration des connaissances de l'expert humain et les données d'entrée-sortie dans un seul schéma, de plus, elle combine deux approximateurs pour obtenir une approximation hybride optimale qui surpasse les minimums locaux et surmonte les limitations individuelles de chaque approximateur.

Le principe de l'hybridation des deux systèmes intelligents adaptatifs de la Figure 2.1 et de la Figure 3.1 est le suivant : chacun des deux systèmes intelligents génère une approximation distincte de chaque fonction inconnue du système non linéaire. Les deux approximations d'une même fonction sont combinées à l'aide des paramètres de modulation $(\alpha, I-\alpha)$ et $(\beta, I-\beta)$, où α et β sont des matrices carrées diagonales avec des valeurs constantes sélectionnées par le concepteur, et I est la matrice d'identité. Attribuer un coefficient de modulation différent à l'approximation de chaque approximateur permet de balancer entre les deux approximateurs en ajustant les éléments de α et β dans l'intervalle $[0, 1]$.

Les fonctions du système $F(x)$ et $G(x)$ sont supposées inconnues et sont approximées à l'aide de l'approximateur FS-WNN hybride comme suit :

$$\begin{aligned} \mathbf{F}(\mathbf{x}) &= \alpha \mathbf{F}_{FS} + (I - \alpha) \mathbf{F}_{WNN} \\ \mathbf{G}(\mathbf{x}) &= \beta \mathbf{G}_{FS} + (I - \beta) \mathbf{G}_{WNN} \end{aligned} \quad (4.1)$$

En détaillant plus l'équation (4,1) en utilisant les équations (2.5)-(2.6) et (3.4)-(3.5), on obtient :

$$\begin{aligned} \mathbf{F}(\mathbf{x}) &= \alpha \left(\mathbf{F}_{FS}^*(\mathbf{x}, \Theta_F^*) + \varepsilon_{F_{FS}}(\mathbf{x}) \right) + (I - \alpha) \left(\mathbf{F}_{WNN}^*(\mathbf{x}, \mathbf{W}_F^*, \mathbf{C}_F^*) + \varepsilon_{F_{WNN}}(\mathbf{x}) \right) \\ &= \mathbf{F}_{FS-WNN}^*(\mathbf{x}, \alpha, \Theta_F^*, \mathbf{W}_F^*, \mathbf{C}_F^*) + \varepsilon_F(\mathbf{x}, \alpha) \end{aligned} \quad (4.2)$$

$$\begin{aligned} \mathbf{G}(\mathbf{x}) &= \beta \left(\mathbf{G}_{FS}^*(\mathbf{x}, \Theta_G^*) + \varepsilon_{G_{FS}}(\mathbf{x}) \right) + (I - \beta) \left(\mathbf{G}_{WNN}^*(\mathbf{x}, \mathbf{W}_G^*, \mathbf{C}_G^*) + \varepsilon_{G_{WNN}}(\mathbf{x}) \right) \\ &= \mathbf{G}_{FS-WNN}^*(\mathbf{x}, \beta, \Theta_G^*, \mathbf{W}_G^*, \mathbf{C}_G^*) + \varepsilon_G(\mathbf{x}, \beta) \end{aligned} \quad (4.3)$$

Avec :

$$\begin{aligned} \varepsilon_F(\mathbf{x}, \alpha) &= \alpha \varepsilon_{F_{FS}}(\mathbf{x}) + (I - \alpha) \varepsilon_{F_{WNN}}(\mathbf{x}) \\ \varepsilon_G(\mathbf{x}, \beta) &= \beta \varepsilon_{G_{FS}}(\mathbf{x}) + (I - \beta) \varepsilon_{G_{WNN}}(\mathbf{x}) \end{aligned}$$

Pratiquement, les paramètres optimaux Θ^*, \mathbf{W}^* et \mathbf{C}^* ne peuvent pas être déterminés, leurs estimations $\hat{\Theta}$, $\hat{\mathbf{W}}$ et $\hat{\mathbf{C}}$ sont utilisées. Par conséquent, en utilisant (2.7)-(2.8) et (3.6)-(3.7), les fonctions inconnues du système sont approximées comme suit:

$$\hat{\mathbf{F}}_{FS-WNN}(\mathbf{x}, \alpha, \hat{\Theta}_F, \hat{\mathbf{W}}_F, \hat{\mathbf{C}}_F) = \alpha \hat{\mathbf{F}}_{FS}(\mathbf{x}, \hat{\Theta}_F) + (I - \alpha) \hat{\mathbf{F}}_{WNN}(\mathbf{x}, \hat{\mathbf{W}}_F, \hat{\mathbf{C}}_F) \quad (4.4)$$

$$\hat{\mathbf{G}}_{FS-WNN}(\mathbf{x}, \beta, \hat{\Theta}_G, \hat{\mathbf{W}}_G, \hat{\mathbf{C}}_G) = \beta \hat{\mathbf{G}}_{FS}(\mathbf{x}, \hat{\Theta}_G) + (I - \beta) \hat{\mathbf{G}}_{WNN}(\mathbf{x}, \hat{\mathbf{W}}_G, \hat{\mathbf{C}}_G) \quad (4.5)$$

Les paramètres de modulation $(\alpha, I - \alpha)$ et $(\beta, I - \beta)$ ont une relation de dépendance. Lorsque la contribution du premier approximateur augmente d'une valeur spécifique, la contribution du deuxième approximateur diminue de la même valeur et vice versa. Ainsi, la somme des deux facteurs complémentaires reste égale à la matrice identité (I).

3. Commande adaptative par FS-WNN hybride :

Considérant la classe des systèmes MIMO non linéaires incertains représentés par l'équation différentielle (2.11). En admettant toujours que les fonctions non linéaires du système $\mathbf{F}(\mathbf{x})$ et $\mathbf{G}(\mathbf{x})$

sont inconnues, $\mathbf{F}(\mathbf{x})$ et $\mathbf{G}(\mathbf{x})$ sont approximées par le nouveau approximateur universel FS-WNN hybride de la Figure 4.3, leurs expressions sont données par les équations (4.4) et (4.5). Donc la loi de commande hybride à base des systèmes flous et réseaux d'ondelettes adaptatifs est synthétisée comme suit :

On reprend l'équation de la dérivée de l'erreur filtrée et on rajoute et on soustrait le terme $\widehat{\mathbf{G}}_{FS-WNN}(\mathbf{x}, \boldsymbol{\beta}, \widehat{\Theta}_G, \widehat{\mathbf{W}}_G, \widehat{\mathbf{C}}_G)\mathbf{u}$ à (2.17), on aura:

$$\begin{aligned} \dot{\mathbf{s}} = & \mathbf{v} - \mathbf{F}(\mathbf{x}) - \left(\mathbf{G}(\mathbf{x}) - \widehat{\mathbf{G}}_{FS-WNN}(\mathbf{x}, \boldsymbol{\beta}, \widehat{\Theta}_G, \widehat{\mathbf{W}}_G, \widehat{\mathbf{C}}_G) \right) \mathbf{u} - \mathbf{d} \\ & - \widehat{\mathbf{G}}_{FS-WNN}(\mathbf{x}, \boldsymbol{\beta}, \widehat{\Theta}_G, \widehat{\mathbf{W}}_G, \widehat{\mathbf{C}}_G) \mathbf{u} \end{aligned} \quad (4.6)$$

Avec $\mathbf{v} = \mathbf{y}_d^{(r)} + \lambda_1 \mathbf{e}^{(r-1)} + \dots + \lambda_{r-2} \mathbf{e}^{(2)} + \lambda_{r-1} \dot{\mathbf{e}}$, on choisit la loi de commande comme suit

$$\begin{aligned} \mathbf{u} = & \widehat{\mathbf{G}}_{FS-WNN}(\mathbf{x}, \boldsymbol{\beta}, \widehat{\Theta}_G, \widehat{\mathbf{W}}_G, \widehat{\mathbf{C}}_G)^{-1} \left(\mathbf{v} + K_{FS-WNN} \operatorname{sgn}(\mathbf{s}) \right. \\ & \left. - \widehat{\mathbf{F}}_{FS-WNN}(\mathbf{x}, \boldsymbol{\alpha}, \widehat{\Theta}_F, \widehat{\mathbf{W}}_F, \widehat{\mathbf{C}}_F) \right) \end{aligned} \quad (4.7)$$

Pour éviter le phénomène de chattering causé par la fonction $\operatorname{sgn}()$ dans la loi de commande (4.7), la fonction $\operatorname{sgn}()$ est substituée par un terme proportionnel intégral PI donné dans (2.23). Ainsi, la loi de contrôle (4.7) est réécrite comme :

$$\mathbf{u} = \widehat{\mathbf{G}}_{FS-WNN}(\mathbf{x}, \boldsymbol{\beta}, \widehat{\Theta}_G, \widehat{\mathbf{W}}_G, \widehat{\mathbf{C}}_G)^{-1} \left(\mathbf{v} + \mathbf{u}_r - \widehat{\mathbf{F}}_{FS-WNN}(\mathbf{x}, \boldsymbol{\alpha}, \widehat{\Theta}_F, \widehat{\mathbf{W}}_F, \widehat{\mathbf{C}}_F) \right) \quad (4.8)$$

Substituant (4.8) dans (4.6), on obtient :

$$\begin{aligned} \dot{\mathbf{s}} = & - \left(\mathbf{F}(\mathbf{x}) - \widehat{\mathbf{F}}_{FS-WNN}(\mathbf{x}, \boldsymbol{\alpha}, \widehat{\Theta}_F, \widehat{\mathbf{W}}_F, \widehat{\mathbf{C}}_F) \right) - \left(\mathbf{G}(\mathbf{x}) - \widehat{\mathbf{G}}_{FS-WNN}(\mathbf{x}, \boldsymbol{\beta}, \widehat{\Theta}_G, \widehat{\mathbf{W}}_G, \widehat{\mathbf{C}}_G) \right) \mathbf{u} \\ & - \mathbf{u}_r - \mathbf{d} \end{aligned} \quad (4.9)$$

En utilisant les expressions (4.2)-(4.5), on obtient :

$$\begin{aligned}
 \dot{\mathbf{s}} = & - \left[\alpha \mathbf{F}_{FS}^*(\mathbf{x}, \Theta_F^*) + (I - \alpha) \mathbf{F}_{WNN}^*(\mathbf{x}, \mathbf{W}_F^*, \mathbf{C}_F^*) + \varepsilon_F(\mathbf{x}, \alpha) \right. \\
 & \left. - \left(\alpha \widehat{\mathbf{F}}_{FS}(\mathbf{x}, \widehat{\Theta}_F) + (I - \alpha) \widehat{\mathbf{F}}_{WNN}(\mathbf{x}, \widehat{\mathbf{W}}_F, \widehat{\mathbf{C}}_F) \right) \right] \\
 & - \left[\beta \mathbf{G}_{FS}^*(\mathbf{x}, \Theta_G^*) + (I - \beta) \mathbf{G}_{WNN}^*(\mathbf{x}, \mathbf{W}_G^*, \mathbf{C}_G^*) + \varepsilon_G(\mathbf{x}, \beta) \right. \\
 & \left. - \left(\beta \widehat{\mathbf{G}}_{FS}(\mathbf{x}, \widehat{\Theta}_G) + (I - \beta) \widehat{\mathbf{G}}_{WNN}(\mathbf{x}, \widehat{\mathbf{W}}_G, \widehat{\mathbf{C}}_G) \right) \right] \mathbf{u} - \mathbf{u}_r - \mathbf{d}
 \end{aligned} \tag{4.10}$$

Réorganisant (4.10) comme suit :

$$\begin{aligned}
 \dot{\mathbf{s}} = & - \left[\alpha \mathbf{F}_{FS}^*(\mathbf{x}, \Theta_F^*) - \alpha \widehat{\mathbf{F}}_{FS}(\mathbf{x}, \widehat{\Theta}_F) + (I - \alpha) \mathbf{F}_{WNN}^*(\mathbf{x}, \mathbf{W}_F^*, \mathbf{C}_F^*) \right. \\
 & \left. - (I - \alpha) \widehat{\mathbf{F}}_{WNN}(\mathbf{x}, \widehat{\mathbf{W}}_F, \widehat{\mathbf{C}}_F) + \varepsilon_F(\mathbf{x}, \alpha) \right] \\
 & - \left[\beta \mathbf{G}_{FS}^*(\mathbf{x}, \Theta_G^*) - \beta \widehat{\mathbf{G}}_{FS}(\mathbf{x}, \widehat{\Theta}_G) + (I - \beta) \mathbf{G}_{WNN}^*(\mathbf{x}, \mathbf{W}_G^*, \mathbf{C}_G^*) \right. \\
 & \left. - (I - \beta) \widehat{\mathbf{G}}_{WNN}(\mathbf{x}, \widehat{\mathbf{W}}_G, \widehat{\mathbf{C}}_G) + \varepsilon_G(\mathbf{x}, \beta) \right] \mathbf{u} - \mathbf{u}_r - \mathbf{d}
 \end{aligned} \tag{4.11}$$

On peut ainsi remettre (4.11) sous la forme suivante :

$$\begin{aligned}
 \dot{\mathbf{s}} = & - \left[\alpha \left(\mathbf{F}_{FS}^*(\mathbf{x}, \Theta_F^*) - \widehat{\mathbf{F}}_{FS}(\mathbf{x}, \widehat{\Theta}_F) \right) \right. \\
 & \left. + (I - \alpha) \left(\mathbf{F}_{WNN}^*(\mathbf{x}, \mathbf{W}_F^*, \mathbf{C}_F^*) - \widehat{\mathbf{F}}_{WNN}(\mathbf{x}, \widehat{\mathbf{W}}_F, \widehat{\mathbf{C}}_F) + \varepsilon_F(\mathbf{x}, \alpha) \right) \right] \\
 & - \left[\beta \left(\mathbf{G}_{FS}^*(\mathbf{x}, \Theta_G^*) - \widehat{\mathbf{G}}_{FS}(\mathbf{x}, \widehat{\Theta}_G) \right) \right. \\
 & \left. + (I - \beta) \left(\mathbf{G}_{WNN}^*(\mathbf{x}, \mathbf{W}_G^*, \mathbf{C}_G^*) - \widehat{\mathbf{G}}_{WNN}(\mathbf{x}, \widehat{\mathbf{W}}_G, \widehat{\mathbf{C}}_G) + \varepsilon_G(\mathbf{x}, \beta) \right) \right] \mathbf{u} - \mathbf{u}_r - \mathbf{d}
 \end{aligned} \tag{4.12}$$

Développant l'expression de $\dot{\mathbf{s}}$ sous forme détaillée comme suit :

$$\begin{aligned}
 \dot{\mathbf{s}} = & - \left[\boldsymbol{\alpha} \left(\Theta_F^{*T} \Phi_F(\mathbf{x}) - \widehat{\Theta}_F^T \Phi_F(\mathbf{x}) \right) + (I - \boldsymbol{\alpha}) \left(\mathbf{W}_F^{*T} \Psi_F^*(\mathbf{x}, \mathbf{C}_F^*) - \widehat{\mathbf{W}}_F^T \widehat{\Psi}_F(\mathbf{x}, \widehat{\mathbf{C}}_F) \right) \right. \\
 & \left. + \boldsymbol{\varepsilon}_F(\mathbf{x}, \boldsymbol{\alpha}) \right] \\
 & - \left[\boldsymbol{\beta} \left(\Theta_G^{*T} \Phi_G(\mathbf{x}) - \widehat{\Theta}_G^T \Phi_G(\mathbf{x}) \right) + (I - \boldsymbol{\beta}) \left(\mathbf{W}_G^{*T} \Psi_G^*(\mathbf{x}, \mathbf{C}_G^*) - \widehat{\mathbf{W}}_G^T \widehat{\Psi}_G(\mathbf{x}, \widehat{\mathbf{C}}_G) \right) \right. \\
 & \left. + \boldsymbol{\varepsilon}_G(\mathbf{x}, \boldsymbol{\beta}) \right] \mathbf{u} - \mathbf{u}_r - \mathbf{d}
 \end{aligned} \tag{4.13}$$

En utilisant la relation $\mathbf{W}^{*T} = \widehat{\mathbf{W}}^T + \widetilde{\mathbf{W}}^T$, on peut écrire $\dot{\mathbf{s}}$ comme suit :

$$\begin{aligned}
 \dot{\mathbf{s}} = & - \left[\boldsymbol{\alpha} \left(\Theta_F^{*T} \Phi_F(\mathbf{x}) - \widehat{\Theta}_F^T \Phi_F(\mathbf{x}) \right) \right. \\
 & \left. + (I - \boldsymbol{\alpha}) \left[\left(\widehat{\mathbf{W}}_F^T \Psi_F^*(\mathbf{x}, \mathbf{C}_F^*) + \widetilde{\mathbf{W}}_F^T \Psi_F^*(\mathbf{x}, \mathbf{C}_F^*) \right) - \widehat{\mathbf{W}}_F^T \widehat{\Psi}_F(\mathbf{x}, \widehat{\mathbf{C}}_F) \right] + \boldsymbol{\varepsilon}_F(\mathbf{x}, \boldsymbol{\alpha}) \right] \\
 & - \left[\boldsymbol{\beta} \left(\Theta_G^{*T} \Phi_G(\mathbf{x}) - \widehat{\Theta}_G^T \Phi_G(\mathbf{x}) \right) \right. \\
 & \left. + (I - \boldsymbol{\beta}) \left[\left(\widehat{\mathbf{W}}_G^T \Psi_G^*(\mathbf{x}, \mathbf{C}_G^*) + \widetilde{\mathbf{W}}_G^T \Psi_G^*(\mathbf{x}, \mathbf{C}_G^*) \right) - \widehat{\mathbf{W}}_G^T \widehat{\Psi}_G(\mathbf{x}, \widehat{\mathbf{C}}_G) \right] + \boldsymbol{\varepsilon}_G(\mathbf{x}, \boldsymbol{\beta}) \right] \mathbf{u} - \mathbf{u}_r \\
 & - \mathbf{d}
 \end{aligned} \tag{4.14}$$

Ce qui permet d'éliminer les termes $\widehat{\mathbf{W}}_F^T \widehat{\Psi}_F(\mathbf{x}, \widehat{\mathbf{C}}_F)$ et $\widehat{\mathbf{W}}_G^T \widehat{\Psi}_G(\mathbf{x}, \widehat{\mathbf{C}}_G)$, donc $\dot{\mathbf{s}}$ devient :

$$\begin{aligned}
 \dot{\mathbf{s}} = & - \left[\boldsymbol{\alpha} \left(\Theta_F^{*T} \Phi_F(\mathbf{x}) - \widehat{\Theta}_F^T \Phi_F(\mathbf{x}) \right) + (I - \boldsymbol{\alpha}) \left(\widetilde{\mathbf{W}}_F^T \widetilde{\Psi}_F(\mathbf{x}, \widetilde{\mathbf{C}}_F) + \widehat{\mathbf{W}}_F^T \Psi_F^*(\mathbf{x}, \mathbf{C}_F^*) \right) \right. \\
 & \left. + \boldsymbol{\varepsilon}_F(\mathbf{x}, \boldsymbol{\alpha}) \right] \\
 & - \left[\boldsymbol{\beta} \left(\Theta_G^{*T} \Phi_G(\mathbf{x}) - \widehat{\Theta}_G^T \Phi_G(\mathbf{x}) \right) + (I - \boldsymbol{\beta}) \left(\widetilde{\mathbf{W}}_G^T \widetilde{\Psi}_G(\mathbf{x}, \widetilde{\mathbf{C}}_G) + \widehat{\mathbf{W}}_G^T \Psi_G^*(\mathbf{x}, \mathbf{C}_G^*) \right) \right. \\
 & \left. + \boldsymbol{\varepsilon}_G(\mathbf{x}, \boldsymbol{\beta}) \right] \mathbf{u} - \mathbf{u}_r - \mathbf{d}
 \end{aligned} \tag{4.15}$$

On développe $\widetilde{\mathbf{W}}_F^T \Psi_F^*(\mathbf{x}, \mathbf{C}_F^*)$ et $\widetilde{\mathbf{W}}_G^T \Psi_G^*(\mathbf{x}, \mathbf{C}_G^*)$ comme suit :

$$\begin{aligned}
 \dot{\mathbf{s}} = & - \left[\boldsymbol{\alpha} \left(\Theta_F^{*T} \Phi_F(\mathbf{x}) - \widehat{\Theta}_F^T \Phi_F(\mathbf{x}) \right) \right. \\
 & + (I - \boldsymbol{\alpha}) \left[\widetilde{\mathbf{W}}_F^T \widetilde{\Psi}_F(\mathbf{x}, \widetilde{\mathcal{C}}_F) + \widetilde{\mathbf{W}}_F^T \left(\widehat{\Psi}_F(\mathbf{x}, \widehat{\mathcal{C}}_F) + \widetilde{\Psi}_F(\mathbf{x}, \widetilde{\mathcal{C}}_F) \right) \right] + \boldsymbol{\varepsilon}_F(\mathbf{x}, \boldsymbol{\alpha}) \left. \right] \\
 & - \left[\boldsymbol{\beta} \left(\Theta_G^{*T} \Phi_G(\mathbf{x}) - \widehat{\Theta}_G^T \Phi_G(\mathbf{x}) \right) \right. \\
 & + (I - \boldsymbol{\beta}) \left[\widetilde{\mathbf{W}}_G^T \widetilde{\Psi}_G(\mathbf{x}, \widetilde{\mathcal{C}}_G) + \widetilde{\mathbf{W}}_G^T \left(\widehat{\Psi}_G(\mathbf{x}, \widehat{\mathcal{C}}_G) + \widetilde{\Psi}_G(\mathbf{x}, \widetilde{\mathcal{C}}_G) \right) \right] + \boldsymbol{\varepsilon}_G(\mathbf{x}, \boldsymbol{\beta}) \left. \right] \mathbf{u} - \mathbf{u}_r \\
 & - \mathbf{d}
 \end{aligned} \tag{4.16}$$

En utilisant la linéarisation de Taylor (voir annexe A), on obtient:

$$\widetilde{\Psi}_F(\mathbf{x}, \widetilde{\mathcal{C}}_F) = \boldsymbol{\Omega}_F^T \widetilde{\mathcal{C}}_F + O_F \tag{4.17}$$

$$\widetilde{\Psi}_G(\mathbf{x}, \widetilde{\mathcal{C}}_G) = \boldsymbol{\Omega}_G^T \widetilde{\mathcal{C}}_G + O_G \tag{4.18}$$

On substitue par (4.17)-(4.18) dans l'expression (4.16), pour avoir :

$$\begin{aligned}
 \dot{\mathbf{s}} = & - \left[\boldsymbol{\alpha} \left(\Theta_F^{*T} \Phi_F(\mathbf{x}) - \widehat{\Theta}_F^T \Phi_F(\mathbf{x}) \right) \right. \\
 & + (I - \boldsymbol{\alpha}) \left[\widetilde{\mathbf{W}}_F^T (\boldsymbol{\Omega}_F^T \widetilde{\mathcal{C}}_F + O_F) + \widetilde{\mathbf{W}}_F^T \left(\widehat{\Psi}_F(\mathbf{x}, \widehat{\mathcal{C}}_F) + \boldsymbol{\Omega}_F^T (\mathbf{C}_F^* - \widehat{\mathcal{C}}_F) + O_F \right) \right] + \boldsymbol{\varepsilon}_F(\mathbf{x}, \boldsymbol{\alpha}) \left. \right] \\
 & - \left[\boldsymbol{\beta} \left(\Theta_G^{*T} \Phi_G(\mathbf{x}) - \widehat{\Theta}_G^T \Phi_G(\mathbf{x}) \right) \right. \\
 & + (I - \boldsymbol{\beta}) \left[\widetilde{\mathbf{W}}_G^T (\boldsymbol{\Omega}_G^T \widetilde{\mathcal{C}}_G + O_G) + \widetilde{\mathbf{W}}_G^T \left(\widehat{\Psi}_G(\mathbf{x}, \widehat{\mathcal{C}}_G) + \boldsymbol{\Omega}_G^T (\mathbf{C}_G^* - \widehat{\mathcal{C}}_G) + O_G \right) \right] \\
 & + \boldsymbol{\varepsilon}_G(\mathbf{x}, \boldsymbol{\beta}) \left. \right] \mathbf{u} - \mathbf{u}_r - \mathbf{d}
 \end{aligned} \tag{4.19}$$

On développe l'expression de $\dot{\mathbf{s}}$ comme suit :

$$\begin{aligned}
 \dot{\mathbf{s}} = & - \left[\boldsymbol{\alpha} \left(\Theta_F^{*T} \Phi_F(\mathbf{x}) - \widehat{\Theta}_F^T \Phi_F(\mathbf{x}) \right) \right. \\
 & + (I - \boldsymbol{\alpha}) \left[\widetilde{\mathbf{W}}_F^T \boldsymbol{\Omega}_F^T \widetilde{\mathbf{C}}_F + \widetilde{\mathbf{W}}_F^T (\widehat{\Psi}_F(\mathbf{x}, \widehat{\mathbf{C}}_F) - \boldsymbol{\Omega}_F^T \widehat{\mathbf{C}}_F) + \widetilde{\mathbf{W}}_F^T \boldsymbol{\Omega}_F^T \mathbf{C}_F^* + \mathbf{W}_F^{*T} O_F \right] \\
 & \left. + \boldsymbol{\varepsilon}_F(\mathbf{x}, \boldsymbol{\alpha}) \right] \\
 & - \left[\boldsymbol{\beta} \left(\Theta_G^{*T} \Phi_G(\mathbf{x}) - \widehat{\Theta}_G^T \Phi_G(\mathbf{x}) \right) \right. \\
 & + (I - \boldsymbol{\beta}) \left[\widetilde{\mathbf{W}}_G^T \boldsymbol{\Omega}_G^T \widetilde{\mathbf{C}}_G + \widetilde{\mathbf{W}}_G^T (\widehat{\Psi}_G(\mathbf{x}, \widehat{\mathbf{C}}_G) - \boldsymbol{\Omega}_G^T \widehat{\mathbf{C}}_G) + \widetilde{\mathbf{W}}_G^T \boldsymbol{\Omega}_G^T \mathbf{C}_G^* + \mathbf{W}_G^{*T} O_G \right] \\
 & \left. + \boldsymbol{\varepsilon}_G(\mathbf{x}, \boldsymbol{\beta}) \right] \mathbf{u} - \mathbf{u}_r - \mathbf{d}
 \end{aligned} \tag{4.20}$$

Si on introduit les termes Λ_F et Λ_G dans (4.20), on obtient :

$$\begin{aligned}
 \dot{\mathbf{s}} = & - \left[\boldsymbol{\alpha} \left(\Theta_F^{*T} \Phi_F(\mathbf{x}) - \widehat{\Theta}_F^T \Phi_F(\mathbf{x}) \right) \right. \\
 & + (I - \boldsymbol{\alpha}) \left[\widetilde{\mathbf{W}}_F^T \boldsymbol{\Omega}_F^T \widetilde{\mathbf{C}}_F + \widetilde{\mathbf{W}}_F^T (\widehat{\Psi}_F(\mathbf{x}, \widehat{\mathbf{C}}_F) - \boldsymbol{\Omega}_F^T \widehat{\mathbf{C}}_F) + \Lambda_F \right] + \boldsymbol{\varepsilon}_F(\mathbf{x}, \boldsymbol{\alpha}) \left. \right] \\
 & - \left[\boldsymbol{\beta} \left(\Theta_G^{*T} \Phi_G(\mathbf{x}) - \widehat{\Theta}_G^T \Phi_G(\mathbf{x}) \right) \right. \\
 & + (I - \boldsymbol{\beta}) \left[\widetilde{\mathbf{W}}_G^T \boldsymbol{\Omega}_G^T \widetilde{\mathbf{C}}_G + \widetilde{\mathbf{W}}_G^T (\widehat{\Psi}_G(\mathbf{x}, \widehat{\mathbf{C}}_G) - \boldsymbol{\Omega}_G^T \widehat{\mathbf{C}}_G) + \Lambda_G \right] + \boldsymbol{\varepsilon}_G(\mathbf{x}, \boldsymbol{\beta}) \left. \right] \mathbf{u} - \mathbf{u}_r - \mathbf{d}
 \end{aligned} \tag{4.21}$$

Avec :

$$\begin{aligned}
 \Lambda_F &= \widetilde{\mathbf{W}}_F^T \boldsymbol{\Omega}_F^T \mathbf{C}_F^* + \mathbf{W}_F^{*T} O_F \\
 \Lambda_G &= \left(\widetilde{\mathbf{W}}_G^T \boldsymbol{\Omega}_G^T \mathbf{C}_G^* + \mathbf{W}_G^{*T} O_G \right) \mathbf{u}
 \end{aligned}$$

Réorganisant $\dot{\mathbf{s}}$ comme suit:

$$\begin{aligned}
 \dot{\mathbf{s}} = & - \left[\alpha \tilde{\Theta}_F^T \Phi_F(\mathbf{x}) + (I - \alpha) [\widehat{\mathbf{W}}_F^T \Omega_F^T \tilde{\mathbf{C}}_F + \widehat{\mathbf{W}}_F^T (\widehat{\Psi}_F(\mathbf{x}, \widehat{\mathbf{C}}_F) - \Omega_F^T \widehat{\mathbf{C}}_F)] \right] \\
 & - \left[\beta \tilde{\Theta}_G^T \Phi_G(\mathbf{x}) \mathbf{u} + (I - \beta) [\widehat{\mathbf{W}}_G^T \Omega_G^T \tilde{\mathbf{C}}_G + \widehat{\mathbf{W}}_G^T (\widehat{\Psi}_G(\mathbf{x}, \widehat{\mathbf{C}}_G) - \Omega_G^T \widehat{\mathbf{C}}_G)] \mathbf{u} \right] - \mathbf{u}_r - \mathbf{d} \quad (4.22) \\
 & + \Delta_{FS-WNN}
 \end{aligned}$$

Cette expression de $\dot{\mathbf{s}}$ nous facilite l'étude de stabilité avec $\Delta_{FS-WNN} = -(I - \alpha)\Lambda_F - (I - \beta)\Lambda_G$.

Hypothèse 8:

L'erreur d'approximation est supposée être bornée par $\|\Delta_{FS-WNN}\| \leq \Gamma_{FS-WNN}$ [78].

Pour atteindre les objectifs de commande, il est nécessaire d'ajuster les paramètres de l'approximateur FS-WNN hybride afin d'avoir $\mathbf{e}(t) \rightarrow 0$ quand $t \rightarrow \infty$. Ainsi, les lois d'adaptation des paramètres du FS-WNN hybride sont données comme suit :

$$\dot{\hat{\Theta}}_{FS-WNN} = \alpha \dot{\hat{\Theta}}_{FFS} \quad (4.23)$$

$$\dot{\hat{\Theta}}_{GFS-WNN} = \beta \dot{\hat{\Theta}}_{GFS} \quad (4.24)$$

$$\dot{\hat{\mathbf{C}}}_{FS-WNN} = -\gamma_{C_F} \Omega_F \widehat{\mathbf{W}}_{FS-WNN} (I - \alpha)^T \mathbf{s} \quad (4.25)$$

$$\dot{\widehat{\mathbf{W}}}_{FS-WNN} = -\gamma_{W_F} (\widehat{\Psi}_F(\mathbf{x}, \widehat{\mathbf{C}}_{FS-WNN}) - \Omega_F^T \widehat{\mathbf{C}}_{FS-WNN}) \mathbf{s}^T (I - \alpha) \quad (4.26)$$

$$\dot{\hat{\mathbf{C}}}_{GFS-WNN} = -\gamma_{C_G} (I - \beta) \Omega_G \widehat{\mathbf{W}}_{GFS-WNN} \mathbf{s} \mathbf{u}^T \quad (4.27)$$

$$\dot{\widehat{\mathbf{W}}}_{GFS-WNN} = -\gamma_{W_G} (I - \beta) (\widehat{\Psi}_G(\mathbf{x}, \widehat{\mathbf{C}}_{GFS-WNN}) - \Omega_G^T \widehat{\mathbf{C}}_{GFS-WNN}) \mathbf{u} \mathbf{s}^T \quad (4.28)$$

Théorème 4:

Considérant la boucle fermée de commande de la Figure 4.3, avec le système MIMO non linéaire incertain (2.11). La loi de commande hybride est choisie comme (4.8) et les lois d'adaptation des paramètres sont donné par (4.14)-(4.19). Le contrôleur à base des approximateurs (FS-WNN)s

hybrides garantit que le système de commande en boucle fermée est stable, et que tous les signaux sont bornés.

Preuve du théorème 4:

Considérant une fonction candidate de Lyapunov comme suit :

$$V = \frac{1}{2} \mathbf{s}^T \mathbf{s} + V_{FS} + V_{WNN} \quad (4.29)$$

La dérivée de (4.20) par rapport au temps est donnée par :

$$\dot{V} = \mathbf{s}^T \dot{\mathbf{s}} + \dot{V}_{FS} + \dot{V}_{WNN} \quad (4.30)$$

En substituant (4.13) dans (4.21), on obtient :

$$\begin{aligned} \dot{V} = & -\mathbf{s}^T \left([\alpha \tilde{\mathbf{F}}_{FS}(\mathbf{x}, \tilde{\Theta}_F) + (I - \alpha) \tilde{\mathbf{F}}_{WNN}(\mathbf{x}, \tilde{\mathbf{W}}_F, \tilde{\mathbf{C}}_F) + \boldsymbol{\varepsilon}_F(\mathbf{x}, \alpha)] \right. \\ & + \left. [\beta \tilde{\mathbf{G}}_{FS}(\mathbf{x}, \tilde{\Theta}_G) + (I - \beta) (\tilde{\mathbf{G}}_{WNN}(\mathbf{x}, \tilde{\mathbf{W}}_G, \tilde{\mathbf{C}}_G)) + \boldsymbol{\varepsilon}_G(\mathbf{x}, \beta)] \mathbf{u} + \mathbf{u}_r + \mathbf{d} \right) + \dot{V}_{FS} \\ & + \dot{V}_{WNN} \end{aligned} \quad (4.31)$$

Sachant que :

$$\begin{aligned} \dot{V}_{FS} &= \dot{V}_{F_{FS}} + \dot{V}_{G_{FS}} \\ \dot{V}_{WNN} &= \dot{V}_{F_{WNN}} + \dot{V}_{G_{WNN}} \end{aligned} \quad (4.32)$$

Donc, l'expression (4.22) devient:

$$\begin{aligned} \dot{V} = & - \left[\mathbf{s}^T \left(\alpha \tilde{\mathbf{F}}_{FS}(\mathbf{x}, \tilde{\Theta}_F) + (I - \alpha) \tilde{\mathbf{F}}_{WNN}(\mathbf{x}, \tilde{\mathbf{W}}_F, \tilde{\mathbf{C}}_F) \right) + \dot{V}_{F_{FS}} + \dot{V}_{F_{WNN}} \right] \\ & - \left[\mathbf{s}^T \left(\beta \tilde{\mathbf{G}}_{FS}(\mathbf{x}, \tilde{\Theta}_G) + (I - \beta) (\tilde{\mathbf{G}}_{WNN}(\mathbf{x}, \tilde{\mathbf{W}}_G, \tilde{\mathbf{C}}_G)) \right) \mathbf{u} + \dot{V}_{G_{FS}} + \dot{V}_{G_{WNN}} \right] \\ & + \mathbf{s}^T (-\boldsymbol{\varepsilon}_F(\mathbf{x}, \alpha) - \boldsymbol{\varepsilon}_G(\mathbf{x}, \beta) \mathbf{u} - \mathbf{u}_r - \mathbf{d}) \end{aligned} \quad (4.33)$$

De même, (4.24) est arrangée comme suit :

$$\begin{aligned} \dot{V} = & - \left[(\mathbf{s}^T \alpha \tilde{\mathbf{F}}_{FS}(\mathbf{x}, \tilde{\Theta}_F) + \dot{V}_{F_{FS}}) + (\mathbf{s}^T (I - \alpha) \tilde{\mathbf{F}}_{WNN}(\mathbf{x}, \tilde{\mathbf{W}}_F, \tilde{\mathbf{C}}_F) + \dot{V}_{F_{WNN}}) \right] \\ & - \left[(\mathbf{s}^T \beta \tilde{\mathbf{G}}_{FS}(\mathbf{x}, \tilde{\Theta}_G) \mathbf{u} + \dot{V}_{G_{FS}}) + (\mathbf{s}^T (I - \beta) \tilde{\mathbf{G}}_{WNN}(\mathbf{x}, \tilde{\mathbf{W}}_G, \tilde{\mathbf{C}}_G) \mathbf{u} + \dot{V}_{G_{WNN}}) \right] \\ & + \mathbf{s}^T (-\boldsymbol{\varepsilon}_F(\mathbf{x}, \alpha) - \boldsymbol{\varepsilon}_G(\mathbf{x}, \beta) \mathbf{u} - \mathbf{u}_r - \mathbf{d}) \end{aligned} \quad (4.34)$$

En substituant par (2.9)-(2.10) et (3.8)-(3.9) dans (4.25), on peut réécrire (4.25) comme suit (voir Annexe B):

$$\begin{aligned}
 \dot{V} = & - \left[\left(\mathbf{s}^T \boldsymbol{\alpha} \tilde{\Theta}_F^T \Phi_F(\mathbf{x}) + \frac{1}{\gamma_{\Theta_F}} \text{tr} \left(\tilde{\Theta}_F^T \dot{\tilde{\Theta}}_F \right) \right) \right. \\
 & + \left(\left(\mathbf{s}^T (I - \boldsymbol{\alpha}) \tilde{\mathbf{W}}_F^T (\hat{\Psi}_F(\mathbf{x}, \hat{\mathbf{C}}_F) - \boldsymbol{\Omega}_F^T \hat{\mathbf{C}}_F) \right) + \frac{1}{\gamma_{W_F}} \text{tr} \left(\tilde{\mathbf{W}}_F^T \dot{\tilde{\mathbf{W}}}_F \right) \right) \\
 & + \left. \left(\mathbf{s}^T (I - \boldsymbol{\alpha}) \tilde{\mathbf{W}}_F^T \boldsymbol{\Omega}_F^T \tilde{\mathbf{C}}_F + \frac{1}{\gamma_{C_F}} \text{tr} \left(\tilde{\mathbf{C}}_F^T \dot{\tilde{\mathbf{C}}}_F \right) \right) \right] \\
 & - \left[\left(\mathbf{s}^T \boldsymbol{\beta} \tilde{\Theta}_G^T \Phi_G(\mathbf{x}) \mathbf{u} + \frac{1}{\gamma_{\Theta_G}} \text{tr} \left(\tilde{\Theta}_G^T \dot{\tilde{\Theta}}_G \right) \right) \right. \\
 & + \left(\mathbf{s}^T (I - \boldsymbol{\beta}) \tilde{\mathbf{W}}_G^T (\hat{\Psi}_G(\mathbf{x}, \hat{\mathbf{C}}_G) - \boldsymbol{\Omega}_G^T \hat{\mathbf{C}}_G) \mathbf{u} + \frac{1}{\gamma_{W_G}} \text{tr} \left(\tilde{\mathbf{W}}_G^T \dot{\tilde{\mathbf{W}}}_G \right) \right) \\
 & + \left. \left(\mathbf{s}^T (I - \boldsymbol{\beta}) \tilde{\mathbf{W}}_G^T \boldsymbol{\Omega}_G^T \tilde{\mathbf{C}}_G \mathbf{u} + \frac{1}{\gamma_{C_G}} \text{tr} \left(\tilde{\mathbf{C}}_G^T \dot{\tilde{\mathbf{C}}}_G \right) \right) \right] \\
 & + \mathbf{s}^T (-\boldsymbol{\varepsilon}_F(\mathbf{x}, \boldsymbol{\alpha}) - \boldsymbol{\varepsilon}_G(\mathbf{x}, \boldsymbol{\beta}) \mathbf{u} - \mathbf{u}_r - \mathbf{d} + \Delta_{FS-WNN})
 \end{aligned} \tag{4.35}$$

Finalement, on obtient :

$$\begin{aligned}
 \dot{V} = & \left[\text{tr} \left(\tilde{\Theta}_F^T \left(\dot{\tilde{\Theta}}_F - \gamma_{\Theta_F} \boldsymbol{\alpha} \Phi_F(\mathbf{x}) \mathbf{s}^T \right) \right) \right. \\
 & + \text{tr} \left(\tilde{\mathbf{W}}_F^T \left(\dot{\tilde{\mathbf{W}}}_F - \gamma_{W_F} (\hat{\Psi}_F(\mathbf{x}, \hat{\mathbf{C}}_F) - \boldsymbol{\Omega}_F^T \hat{\mathbf{C}}_F) \mathbf{s}^T (I - \boldsymbol{\alpha}) \right) \right) \\
 & + \text{tr} \left(\tilde{\mathbf{C}}_F^T \left(\dot{\tilde{\mathbf{C}}}_F - \gamma_{C_F} \boldsymbol{\Omega}_F^T \tilde{\mathbf{W}}_F (I - \boldsymbol{\alpha})^T \mathbf{s} \right) \right) \\
 & + \left[\text{tr} \left(\tilde{\Theta}_G^T \left(\dot{\tilde{\Theta}}_G - \gamma_{\Theta_G} \boldsymbol{\beta} \Phi_G(\mathbf{x}) \mathbf{u} \mathbf{s}^T \right) \right) \right. \\
 & + \text{tr} \left(\tilde{\mathbf{W}}_G^T \left(\dot{\tilde{\mathbf{W}}}_G - \gamma_{W_G} (I - \boldsymbol{\beta}) (\hat{\Psi}_G(\mathbf{x}, \hat{\mathbf{C}}_G) - \boldsymbol{\Omega}_G^T \hat{\mathbf{C}}_G) \mathbf{u} \mathbf{s}^T \right) \right) \\
 & + \left. \text{tr} \left(\tilde{\mathbf{C}}_G^T \left(\dot{\tilde{\mathbf{C}}}_G - \gamma_{C_G} (I - \boldsymbol{\beta}) \boldsymbol{\Omega}_G^T \tilde{\mathbf{W}}_G \mathbf{u} \mathbf{s}^T \right) \right) \right] + \mathbf{s}^T (-\mathbf{u}_r - \mathbf{d} + \Delta_{FS-WNN})
 \end{aligned} \tag{4.36}$$

Par conséquent, en utilisant (4.14)-(4.19), on obtient:

$$\dot{V} = \mathbf{s}^T (-\mathbf{u}_r - \mathbf{d} + \Delta_{FS-WNN}) \tag{4.37}$$

En utilisant les **hypothèses 4 et 8**:

$$\dot{V} \leq \mathbf{s}^T (-K_{FS-WNN} \operatorname{sgn}(\mathbf{s}) + D + \Gamma_{FS-WNN}) \quad (4.38)$$

$$\dot{V} \leq \|\mathbf{s}^T\| (-K_{FS-WNN} + \xi_{FS-WNN}) \quad (4.39)$$

Étant donné que K_{FS-WNN} et $\xi_{FS-WNN} = D + \Gamma_{FS-WNN}$ sont sélectionnés par le choix du concepteur, K_{FS-WNN} doit être choisi de telle sorte que $\dot{V} < 0$. Par conséquent, le système en boucle fermée est stable. Ainsi, tous les signaux sont bornés ce qui implique que l'erreur de poursuite \mathbf{e} et tous ses dérivés sont bornés, également \mathbf{s} est bornée, c'est-à-dire, $\mathbf{s} \in L_\infty$.

L'intégration des deux côtés de (4.30) donne:

$$\int_0^t \|\mathbf{s}\| d\tau \leq \frac{1}{K_{FS-WNN} - \xi_{FS-WNN}} (|V(0)| + |V(t)|) \quad (4.40)$$

Ce qui signifie que $\mathbf{s} \in L_1$, en se basant sur le Lemme de Barbalat [95], on trouve que: $\lim_{t \rightarrow \infty} \|\mathbf{s}(t)\| = 0$, donc $\lim_{t \rightarrow \infty} \|\mathbf{e}(t)\| = 0$. Ainsi, l'erreur de poursuite convergera vers zéro quand $t \rightarrow \infty$. En observant (4.31), nous constatons que la stabilité du système de la boucle fermée est indépendante des paramètres de modulation α et β et quelle que soit la valeur de α et β la stabilité et les performances du système de contrôle en boucle fermée sont toujours garanties, même dans les cas extrêmes, c'est-à-dire lorsque $\alpha=0$ et $\beta=0$ ou bien quand $\alpha=I$ et $\beta=I$.

4. Résultats de simulation

On utilise le modèle du robot manipulateur à deux degrés de liberté de l'équation (2.63) comme un système MIMO non linéaire incertains pour vérifier l'efficacité du schéma de commande développée. Pour pouvoir établir une étude comparative avec les schémas de commande présentés dans les chapitres 02 et 03, on garde les mêmes conditions initiales du robot manipulateur comme $(x_1, x_2, x_3, x_4) = (0.3, 1, 0.5, 0)$. Cependant, dans le schéma de commande développé dans ce chapitre, deux FS-WNNs hybrides sont synthétisés pour approximer les fonctions inconnues $\mathbf{F}(\mathbf{x})$ et $\mathbf{G}(\mathbf{x})$ du système. Leurs approximations sont utilisées dans la synthèse de la loi de commande du contrôleur adaptatif indirect donnée par (4.8). Les paramètres de design de la loi de commande (4.8) sont sélectionnés comme dans les cas des contrôleurs FS et WNN

avec: $K_1=K_2=\text{diag}(8,12)$, $K=20I_2$, les paramètres de la surface de glissement sont sélectionnés à partir de [79] comme $\lambda=I_2$.

Chaque FS-WNN a été synthétisé en combinant le FS MIMO développé dans chapitre 3 avec le WNN MIMO développé dans le chapitre 2 en utilisant la technique de modulation, où l'approximation de la fonction $F(\mathbf{x})$ générée par le FS a été multipliée par le facteur α , et l'approximation de la fonction $F(\mathbf{x})$ générée par le WNN a été multipliée par le facteur complémentaire de α qui est $(I-\alpha)$. La somme pondérée de ces approximations constitue l'approximation hybride de la fonction $F(\mathbf{x})$ inconnue du système. La même procédure a été utilisée pour approximer $G(\mathbf{x})$, où les paramètres de modulation ($\beta, I-\beta$) ont été utilisés pour associer les différentes approximations de $G(\mathbf{x})$. Les valeurs de α et β ont été déterminées d'une manière heuristique comme étant $\alpha=\text{diag}(0.01,0.4)$ et $\beta=\text{diag}(0.7,0.8)$. Les trajectoires désirées sont sélectionnées comme $\mathbf{y}_d^T = [\sin(t) \quad \cos(t)]$.

Pour mettre au point l'intérêt de l'hybridation de deux approximateurs différents en utilisant la technique de modulation pour l'approximation des dynamiques inconnues du système. On reprend l'approximateur WNN développé dans le chapitre 2 et l'approximateur FS développé dans le chapitre 03 avec les mêmes paramètres. En rappelant que le système flou développé dans le chapitre 03 à 04 entrées, deux sorties et 81 règles ou on associe pour chaque entrée x_i trois fonctions d'appartenances Gaussiennes, Leurs paramètres de translation et de dilatation sont initialisés comme $p_{ik}^1=(-1,0,1)$ et $p_{ik}^2=1.05$ pour $k=1,\dots,n$, tandis que les poids de sortie de FS sont choisis arbitrairement dans l'intervalle $[-1,1]$ et adaptés en ligne avec les mêmes gains d'adaptation: $\gamma_{\Theta_F} = 600, \gamma_{\Theta_G} = 0.1$. Le WNN développé dans le chapitre 2 à 04 entrées qui sont les états du système, deux sorties et 81 nœuds cachés. Les paramètres de translation et de dilatation du WNN sont initialisés comme $\mathbf{c}=(-0.5,0,0.5)$ et $\sigma_{ki}=1.2$ for $k=1,\dots,n$, tandis que les poids de sortie sont choisis arbitrairement dans l'intervalle $[-1,1]$. Les paramètres de translation et les poids de sortie du WNN sont adaptés en ligne. Les gains d'adaptation sont repris de chapitre 3 comme : $\gamma_{C_F} = 0.008$, $\gamma_{W_F} = 320$, $\gamma_{C_G} = 0.002$, $\gamma_{W_G} = 0.001$. Le nouveau approximateur FS-WNN hybride sera synthétisé en utilisant ces deux approximateurs tel qu'ils sont et ses performances seront comparés avec celles obtenues dans le cas de contrôle adaptatif indirect en utilisant les FS et les WNNs.

Pour évaluer les performances de FS-WNN hybride par rapport au FS et au WNN, trois contrôleurs adaptatifs indirects ont été synthétisés en fonction des trois approximateurs. Les trois contrôleurs subissent trois tests différents comme dans la littérature [79], le premier test concerne

le système robotisant dans des conditions nominales, c'est-à-d. en l'absence de perturbations et de variations des paramètres du système. Le deuxième et le troisième test visent à explorer la robustesse des trois contrôleurs, où le deuxième test étudie les effets de la friction, introduit à $t=15$ sec. Le troisième test étudie les effets de la friction ajoutée au changement paramétrique lorsque le robot prend une charge de 5 kg, introduites à 35 sec. Les performances de poursuite dans des conditions nominales pour la première articulation y_1 et la deuxième articulation y_2 sont données dans la Figure 4.4 et la Figure 4.5, respectivement. Bien que les Figure 4.6 et 4.7 montrent les signaux de commande pour les deux articulations, et les erreurs de poursuite sont illustrées dans la Figure 4.8 pour la première articulation et la Figure 4.9 pour la deuxième articulation.

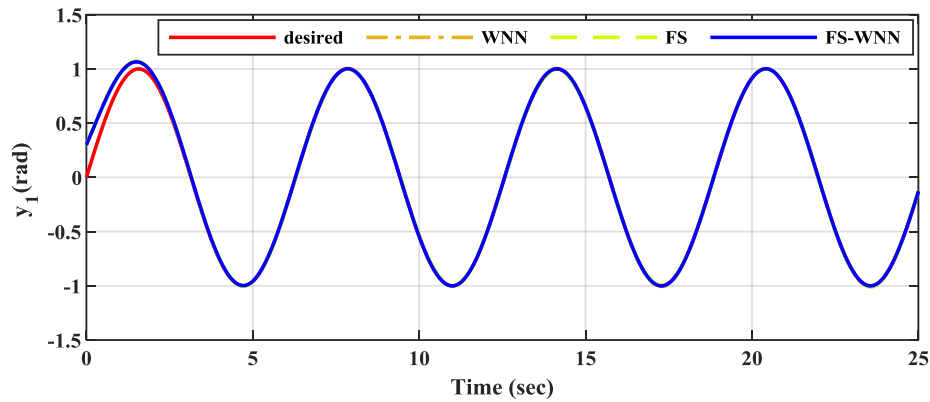


Figure 4.4 : Performance de poursuite de y_1 d'une trajectoire désirée y_{d1} pour les trois contrôleurs (WNN, FS, FS-WNN) en absence des perturbations et des variations paramétriques.

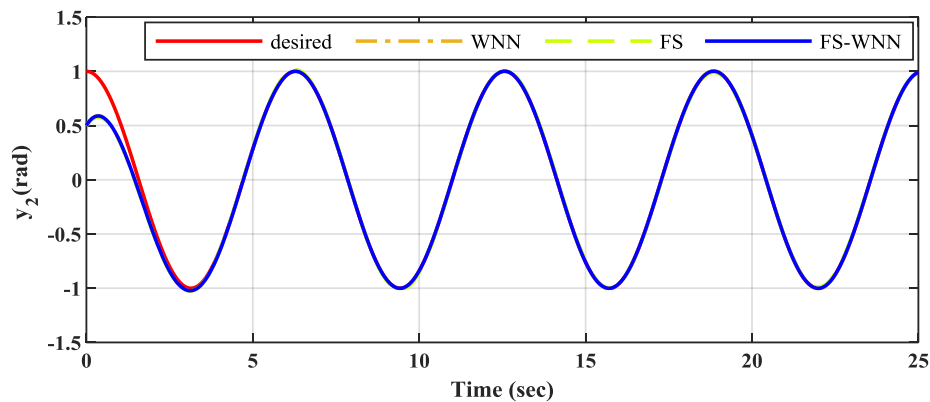


Figure 4.5 : Performance de poursuite de y_2 d'une trajectoire désirée y_{d2} pour les trois contrôleurs (WNN, FS, FS-WNN) en absence des perturbations et des variations paramétriques.

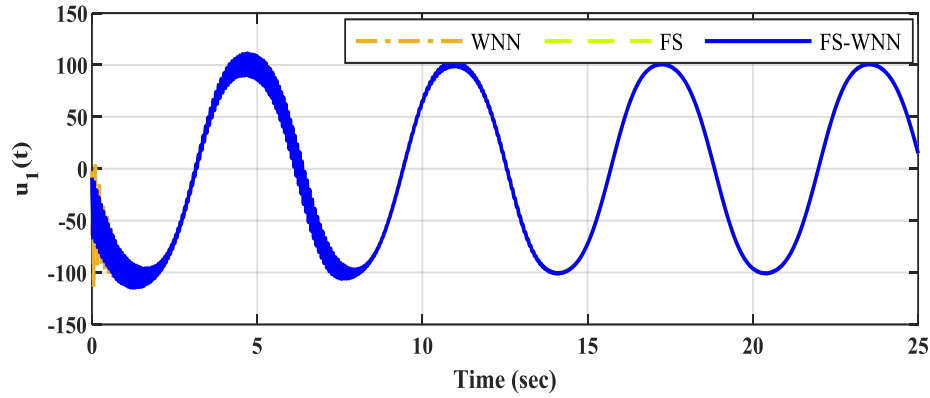


Figure 4.6 : Signal de commande u_1 pour les trois contrôleurs (WNN, FS, FS-WNN) en absence des perturbations et des variations paramétriques.

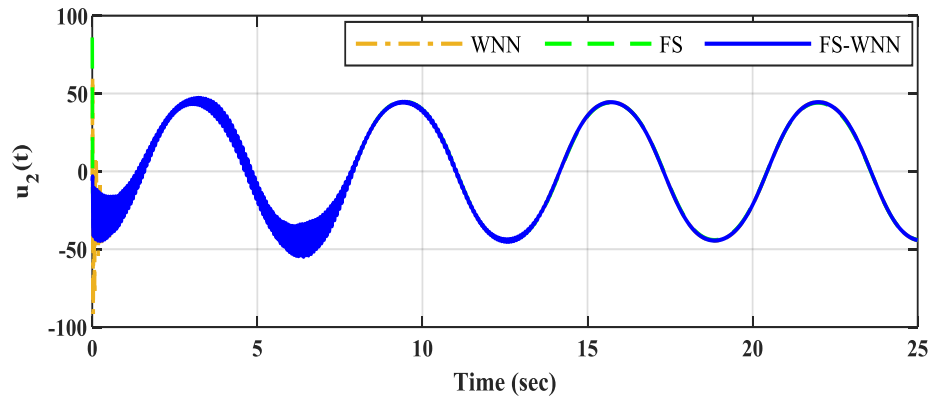


Figure 4.7 : Signal de commande u_2 pour les trois contrôleurs (WNN, FS, FS-WNN) en absence des perturbations et des variations paramétriques.

La Figure 4.4 et la Figure 4.5 montrent les performances de poursuite des sorties du système y_1 et y_2 pour les trois contrôleurs, c.-à-d., le contrôleur basé sur l'approximateur WNN, le contrôleur basé sur l'approximateur FS et le contrôleur proposé basé sur l'approximateur hybride FS-WNN. Les trois contrôleurs pourraient assurer des bonnes performances, où les sorties du système y_1 et y_2 suivaient avec précision la trajectoire désirée y_d .

Les signaux de commande des trois contrôleurs en absence des perturbations et des variations paramétriques sont donnés dans la Figure 4.6 pour la première articulation et dans la Figure 4.7 pour la deuxième articulation. Le contrôleur synthétisé à la base de WNN montre un signal de commande oscillatoire dans la phase transitoire. Dans le cas du contrôleur synthétisé à la base de FS, le signal de commande est lisse. Cependant, les signaux de commande des contrôleurs FS et WNN commencent par des grandes oscillations comme le montre la Figure 4.7. Par contre, Le

contrôleur à base de FS-WNN hybride montre un signal de commande avec de petites oscillations dans la phase transitoire qui disparaissent rapidement.

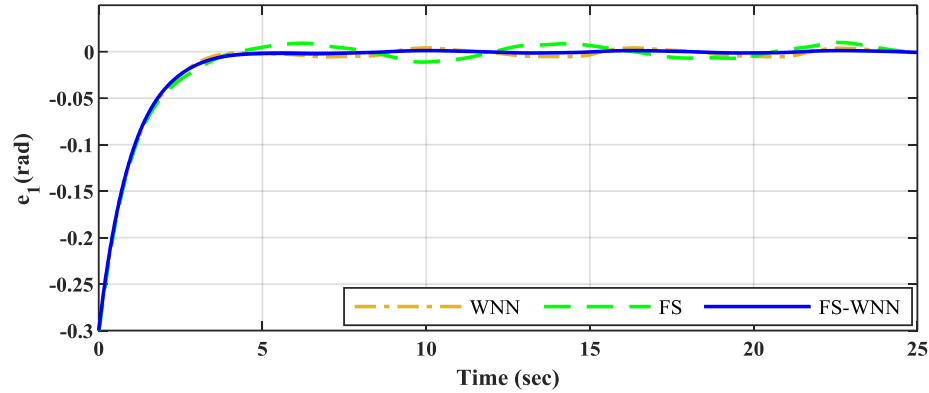


Figure 4.8 : Erreur de poursuite e_1 pour les trois contrôleurs (WNN, FS, FS-WNN) en absence des perturbations et des variations paramétriques.

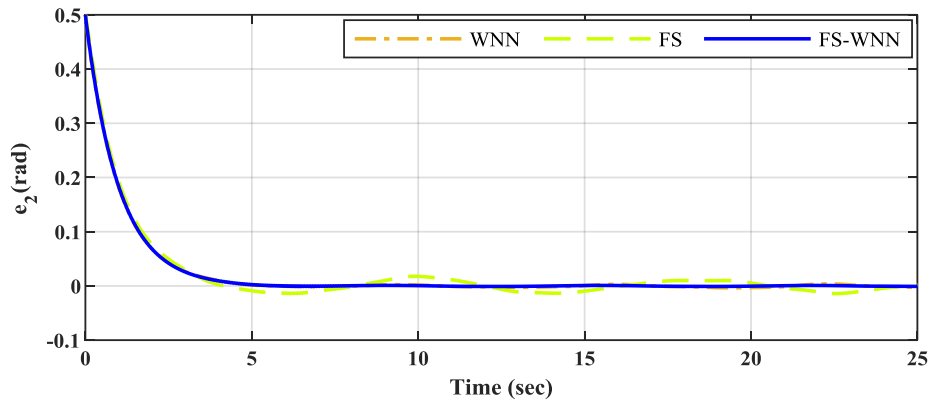


Figure 4.9 : Erreur de poursuite e_2 pour les trois contrôleurs (WNN, FS, FS-WNN) en absence des perturbations et des variations paramétriques.

La Figure 4.8 et la Figure 4.9 montrent les erreurs de poursuite des trois contrôleurs en l'absence des perturbations externes et des variations paramétriques. D'après la Figure 4.8 et la Figure 4.9, on peut voir que l'erreur de poursuite dans le cas du contrôleur à base de FS est plus importante que le cas du contrôleur à base de WNN. En outre, ces Figures montrent que l'erreur de poursuite la plus petite et la plus lisse est obtenue dans le cas du contrôleur à base de FS-WNN adaptatif hybride.

Pour analyser précisément les performances des trois contrôleurs en l'absence des perturbations externes et des variations paramétriques, leurs performances sont comparées en

termes de leurs précisions à l'aide de la fonction MSE et à l'aide de la valeur maximale de signal de commande. La fonction MSE est donnée comme :

$$MSE = \frac{1}{N} \sum_{i=1}^N e_i^2$$

où N est la longueur du vecteur d'erreur e_i . La valeur maximale du signal de commande est donnée comme : $\text{Max}(|u|)$. Ces critères sont considérés dans deux cas, le premier cas est la période transitoire et le second cas comprend que la période permanente à partir de $t= 8$ sec.

La comparaison des performances des trois contrôleurs en fonction de MSE de leurs erreurs de poursuite est donnée dans le Tableau 4.1 pour les deux articulations, en période transitoire et permanente, en l'absence des perturbations externes et des variations paramétriques, tandis que la comparaison des performances de ces contrôleurs en fonction de la valeur maximale de signaux de commandes est donnée dans le Tableau 4.2 pour les deux articulations dans les mêmes conditions.

Tableau 4.1 : Comparaison de MSE de l'erreur de poursuite des trois contrôleurs (WNN, FS, FS-WNN) en l'absence des perturbations et des variations paramétriques.

	Période transitoire		Période permanente	
	Link 1	Link 2	Link 1	Link 2
WNN	0.0059	0.0156	1.11×10^{-5}	4.19×10^{-6}
FS	0.0061	0.0169	3.84×10^{-5}	8.83×10^{-5}
FS-WNN	0.0057	0.0157	7.82×10^{-7}	4.38×10^{-7}

Tableau 4.2 : Comparaison de la valeur maximale de signal de commande des trois contrôleurs (WNN, FS, FS-WNN) en l'absence des perturbations et des variations paramétriques.

	Période transitoire		Période permanente	
	Max u ₁	Max u ₂	Max u ₁	Max u ₂
WNN	114.91	91.48	100.92	44.54
FS	104.73	86.00	100.99	44.82
FS-WNN	114.63	54.87	102.73	45.32

Le Tableau 4.1 compare les performances des trois contrôleurs en fonction de MSE de leurs erreurs de poursuite, tandis que le Tableau 4.2 compare les performances de ces contrôleurs en

fonction de la valeur maximale de leurs signaux de commande. En observant le MSE dans la période permanente pour les deux articulations, nous voyons que le WNN a présenté une bonne poursuite avec une erreur de poursuite au voisinage de 10^{-6} par rapport au FS qui est au voisinage de 10^{-5} . Cependant, en combinant les deux approximateurs et par un choix approprié de α et β , le FS-WNN hybride a minimisé l'erreur de poursuite jusqu'au 10^{-7} .

A partir du Tableau 4.1, il est clair que l'hybridation de FS avec le WNN en utilisant la technique de modulation donne un approximateur hybride optimale. L'approximateur hybride a exploité les avantages des deux approximateurs pour achever une poursuite optimale. De plus, en observant la période transitoire dans le Tableau 4.2, on remarque des oscillations de grandes amplitudes dans les signaux de commande de WNN et de FS. Cela pourrait endommager l'équipement dans des applications du monde réel [98]. Au contraire, le contrôleur FS-WNN hybride n'a pas présenté cet inconvénient. Ainsi, le FS-WNN hybride a été plus performant que le WNN et le FS par rapport aux deux critères.

Pour évaluer la performance de FS-WNN hybride en présence de perturbations et de variations paramétriques par rapport au WNN et au FS, les trois contrôleurs subissent un test des effets de frottement ajoutés à $t=15$ sec, en plus un changement des paramètres de système ajouté à $t=35$ sec. Les performances de poursuite de la position de la première articulation y_1 et de la deuxième articulation y_2 sont données aux Figure 4.10 et Figure 4.11, respectivement. Les Figures 4.12 et 4.13 montrent les signaux de commande pour les deux articulations.

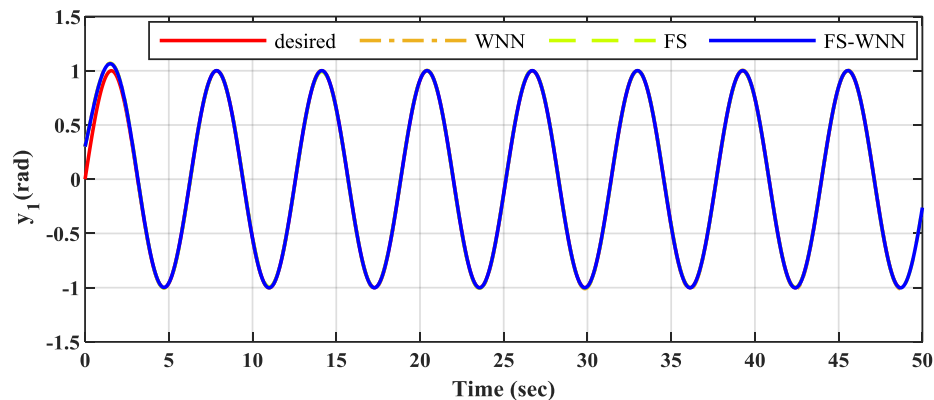


Figure 4.10 : Performance de poursuite de y_1 pour les trois contrôleurs (WNN, FS, FS-WNN) dans le cas de présence des perturbations et des variations paramétriques.

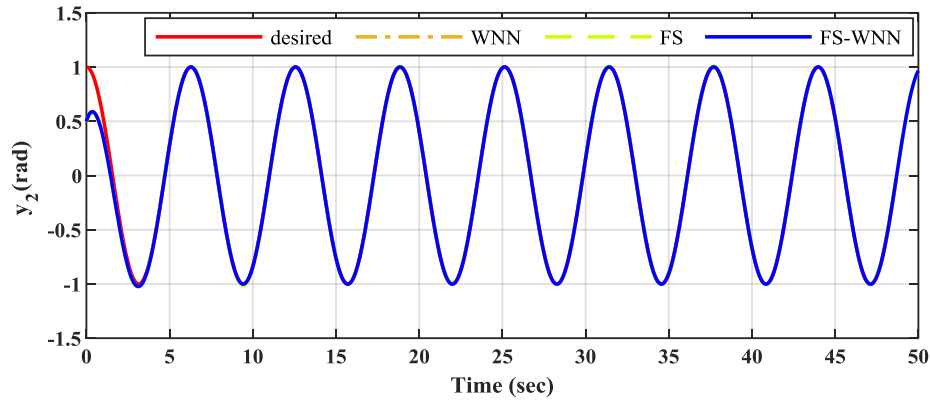


Figure 4.11 : Performance de poursuite de y_2 pour les trois contrôleurs (WNN, FS, FS-WNN) dans le cas de présence des perturbations et des variations paramétriques

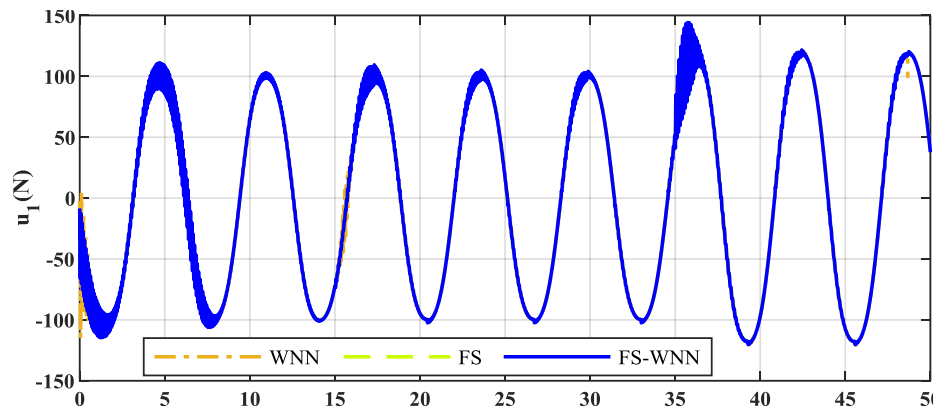


Figure 4.12 : Signal de commande u_1 pour les trois contrôleurs (WNN, FS, FS-WNN) dans le cas de présence des perturbations et des variations paramétriques

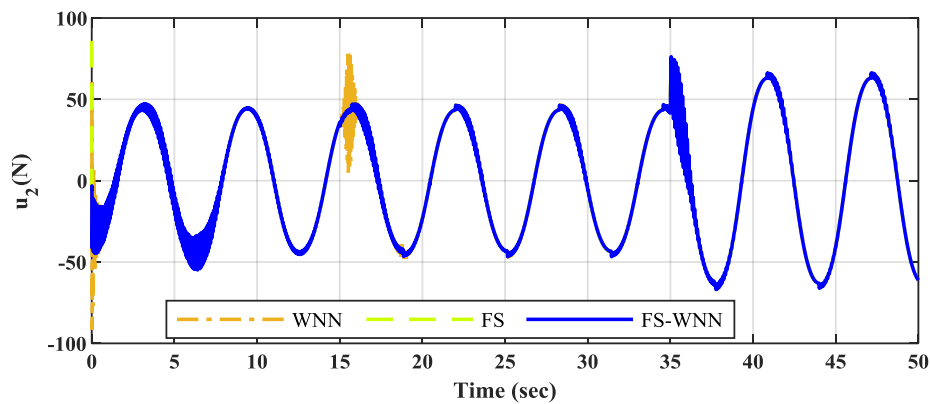


Figure 4.13 : Signal de commande u_2 pour les trois contrôleurs (WNN, FS, FS-WNN) dans le cas de présence des perturbations et des variations paramétriques

D'après les résultats de simulation de la Figure 4.10 et la Figure 4.11, on peut voir que la poursuite des trajectoires désirées peut encore être réalisée par les trois contrôleurs même en présence de perturbations et des variations paramétriques. Toutefois, la Figure 4.13 montre que le signal de commande généré par le contrôleur basé sur le WNN a été extrêmement influencé par l'introduction de perturbations externes à l'instant $t=15$ sec, également lorsqu'un paramètre de système a changé à l'instant $t=35$ sec. D'autre part, le FS a présenté une grande robustesse contre les perturbations externes et a maintenu un signal de contrôle lisse. Et vu que le FS-WNN hybride est une combinaison des deux systèmes intelligents, il a bénéficié de la robustesse du FS pour corriger la dégradation du signal de commande engendré par le WNN. Par conséquent, aucun changement brusque n'a été noté dans le signal de commande de FS-WNN lors de l'ajout de perturbations à $t=15$ sec, et seulement de faibles oscillations sont apparues. De plus, lorsqu'on a associé un changement paramétrique aux perturbations à $t=35$ sec, le signal de commande a été rapidement perturbé. Mais, par le fort influence d'adaptation des centres et des poids de sortie du WNN, l'effet des incertitudes a été rapidement compensé.

Afin d'analyser avec précision les performances des trois contrôleurs en cas de variations paramétriques et de perturbations externes, leurs performances sont comparées en termes de leurs précisions de poursuite à l'aide de la fonction MSE et le Max ($|u|$). Les résultats obtenus sont donnés dans le Tableau 4.3 pour les deux articulations dans le régime permanent.

Tableau 4.3 : Comparaison des performances des trois contrôleurs (WNN, FS, FS-WNN) dans le cas de présence des perturbations externes et des variations paramétriques.

	Link 1		Link 2	
	MSE(y_1)	Max $ u_1 $	MSE(y_2)	Max $ u_2 $
WNN	2.88×10^{-5}	121.04	9.34×10^{-6}	78.93
FS	2.59×10^{-5}	119.97	5.98×10^{-5}	65.35
FS-WNN	1.13×10^{-6}	144.16	5.78×10^{-7}	76.20

Le Tableau 4.3 compare les performances des trois contrôleurs dans le cas des variations des paramètres du système et des perturbations externes pour la première et la deuxième articulation dans le régime permanent. Le MSE de l'erreur de poursuite et le Max ($|u|$) sont fixés comme critères d'évaluation. En observant le MSE pendant le régime permanent pour les deux articulations, nous remarquons que le contrôleur basé sur le WNN a présenté une bonne poursuite avec une erreur de poursuite de l'ordre de 10^{-6} , même en présence de perturbations. En parallèle, le

MSE de FS s'est amélioré. Cependant, le contrôleur basé sur FS-WNN a donné le meilleur MSE dans l'ordre de 10^{-7} .

En observant le Max ($|u|$) dans le Tableau 5.3 pour la deuxième articulation, nous voyons que le WNN était sensible aux perturbations et présentait des changements brusques dans le signal de commande avec des grandes oscillations. En différence avec le WNN, le FS et le FS-WNN étaient robustes aux perturbations, et pourraient compenser les effets des changements des paramètres de système par une augmentation de signal de commande.

Du Tableau 4.3, l'aspect intéressant de l'hybridation de deux systèmes intelligents différents apparaît clairement. Le FS-WNN hybride a combiné la robustesse et l'insensibilité de FS face aux variations paramétriques et aux perturbations avec les bonnes performances de poursuite et la forte influence de l'adaptation des centres et des poids de WNN pour minimiser les effets des incertitudes des perturbations. Tout en évitant les changements brusques et les grandes oscillations dans le signal de commande.

Le choix des paramètres α et β du FS-WNN hybride comme étant $\alpha = \text{diag}(0.01, 0.4)$ et $\beta = \text{diag}(0.7, 0.8)$, a été fait heuristiquement. Lors de l'observation des valeurs des éléments diagonales de α , on remarque que plus d'importance est accordée au WNN par $I - \alpha = \text{diag}(0.99, 0.6)$. Et en observant les valeurs des éléments diagonales de β , on remarque que plus d'importance est accordée au FS par $\beta = \text{diag}(0.7, 0.8)$. Ce qui peut être expliqué comme suit: le WNN approxime mieux la fonction non linéaire $F(\mathbf{x})$ du système. En outre, la sortie du WNN est fortement non linéaire ce qui convient à l'approximation de $F(\mathbf{x})$. Par conséquent, il obtient le degré de contribution le plus élevé. D'un autre côté, le FS approxime mieux la fonction non linéaire $G(\mathbf{x})$ du système. En outre, le FS a montré une grande robustesse et insensibilité aux perturbations externes et variations paramétriques, ce qui le rend plus adéquat pour estimer la matrice des gains de commande.

Afin d'analyser les performances du système de contrôle en régime nominale pour d'autres valeurs des paramètres de modulation, on établit une étude comparative entre plusieurs valeurs de α et β dans l'intervalle $[0, 1]$. En commençant par $\alpha=0$ et $\beta=0$, dans ce cas-là, le système flou opère seul et le contrôleur neuronal à base d'ondelettes est désactivé. On incrémente α et β d'un pas de 0.2 à partir de 0.1 jusqu'à 0.9. Ensuite, on étudie le cas où $\alpha=I$ et $\beta=I$, dans ce cas-là, le réseau de neurones à base d'ondelettes opère seul et le contrôleur flou est désactivé. Les résultats de simulation sont donnés dans la Figures 4.14 pour $\alpha=\beta=0$ jusqu'à la Figures 4.20 où $\alpha=\beta=I$.

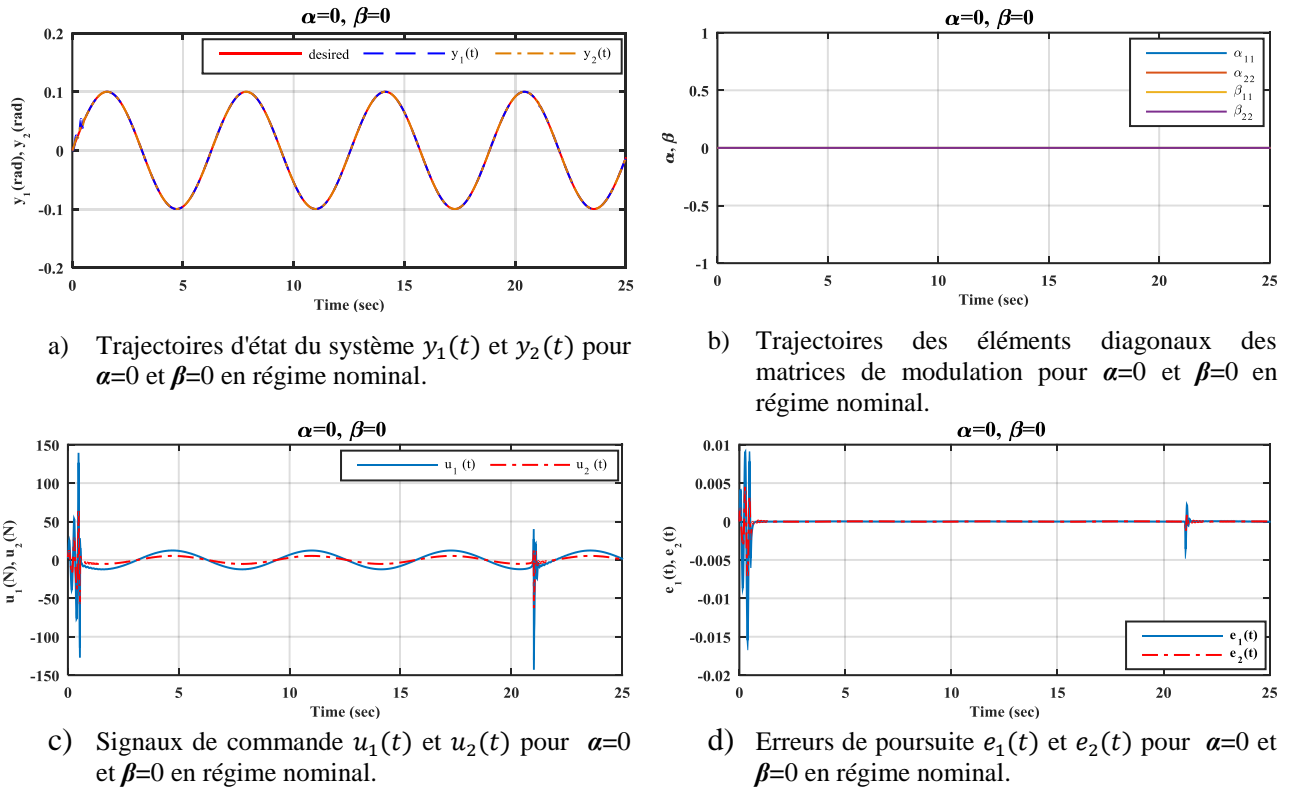


Figure 4.14 : Performance du contrôleur FS-WNN hybride pour $\alpha=0$ et $\beta=0$.

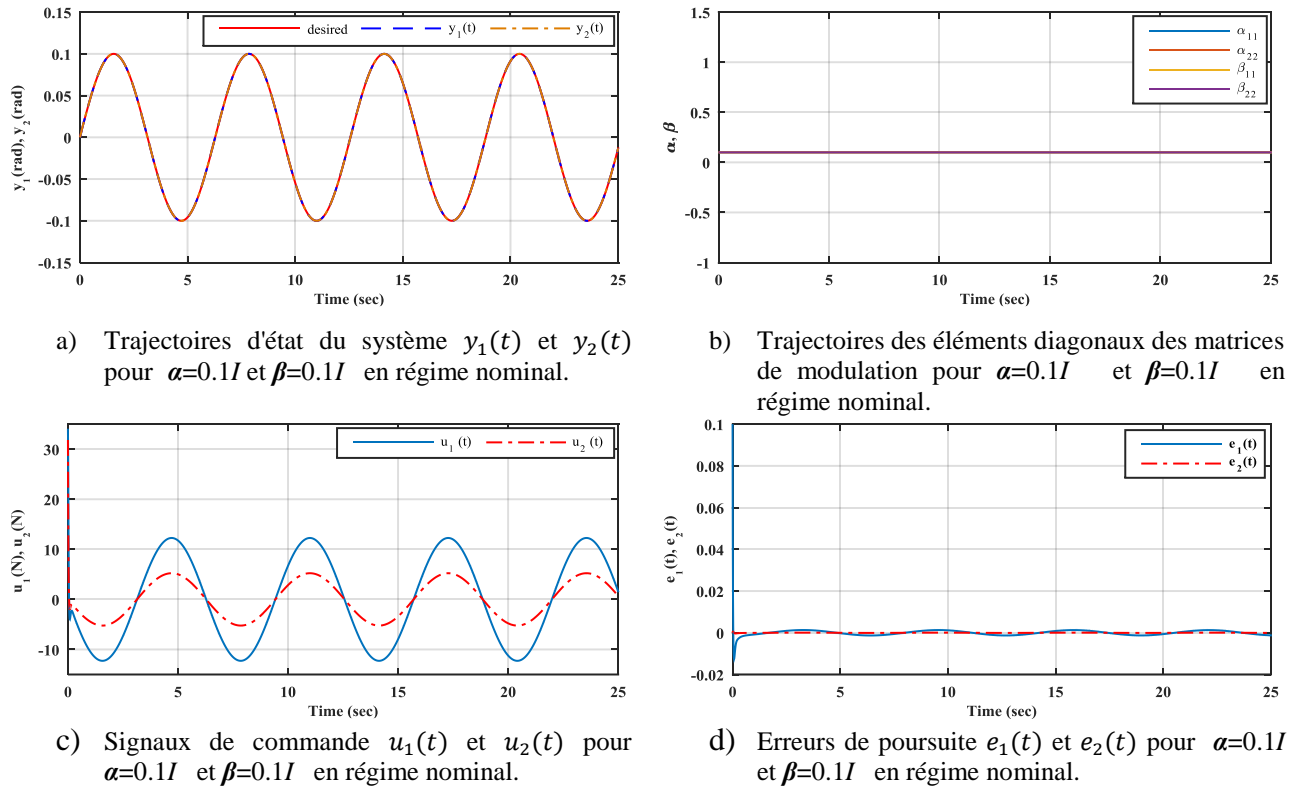


Figure 4.15 : Performances du contrôleur FS-WNN hybride pour $\alpha=0.11$ et $\beta=0.11$.

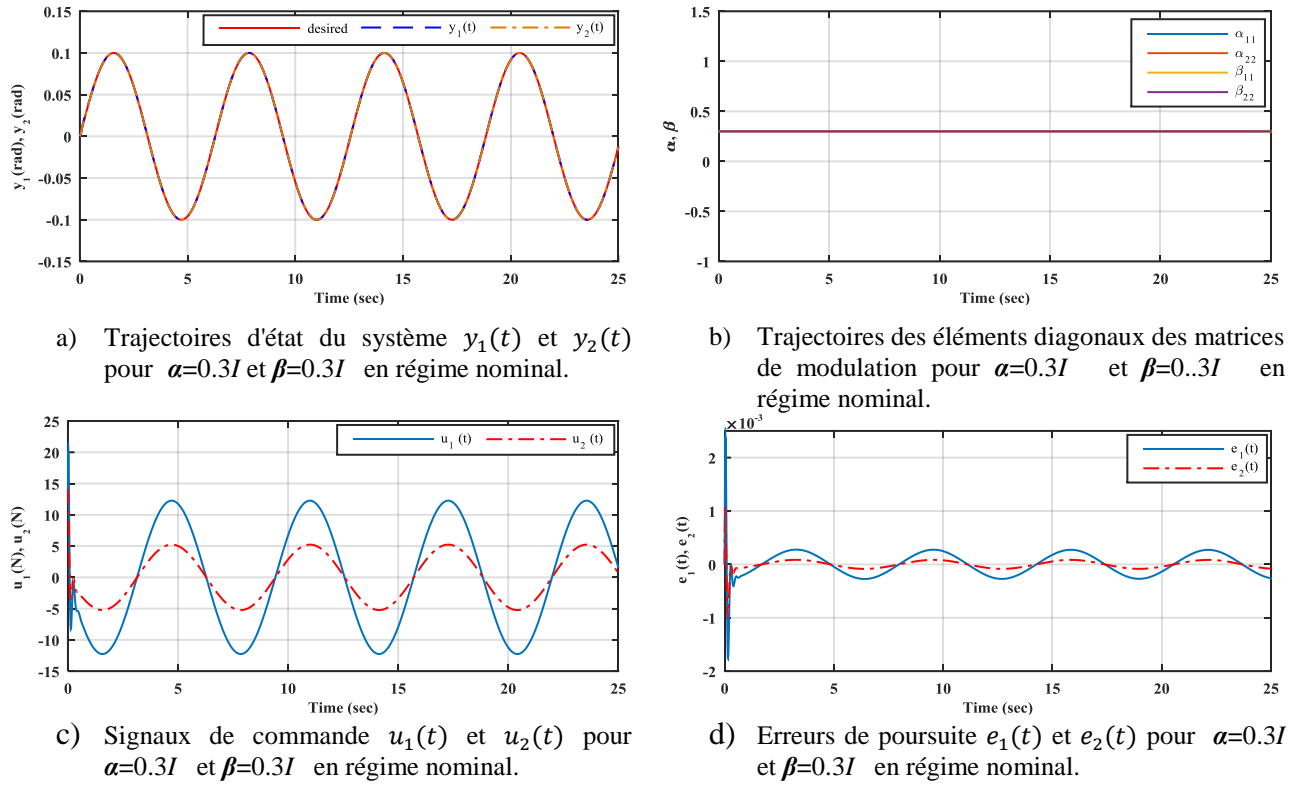


Figure 4.16 : Performances du contrôleur FS-WNN hybride pour $\alpha=0.3I$ et $\beta=0.3I$

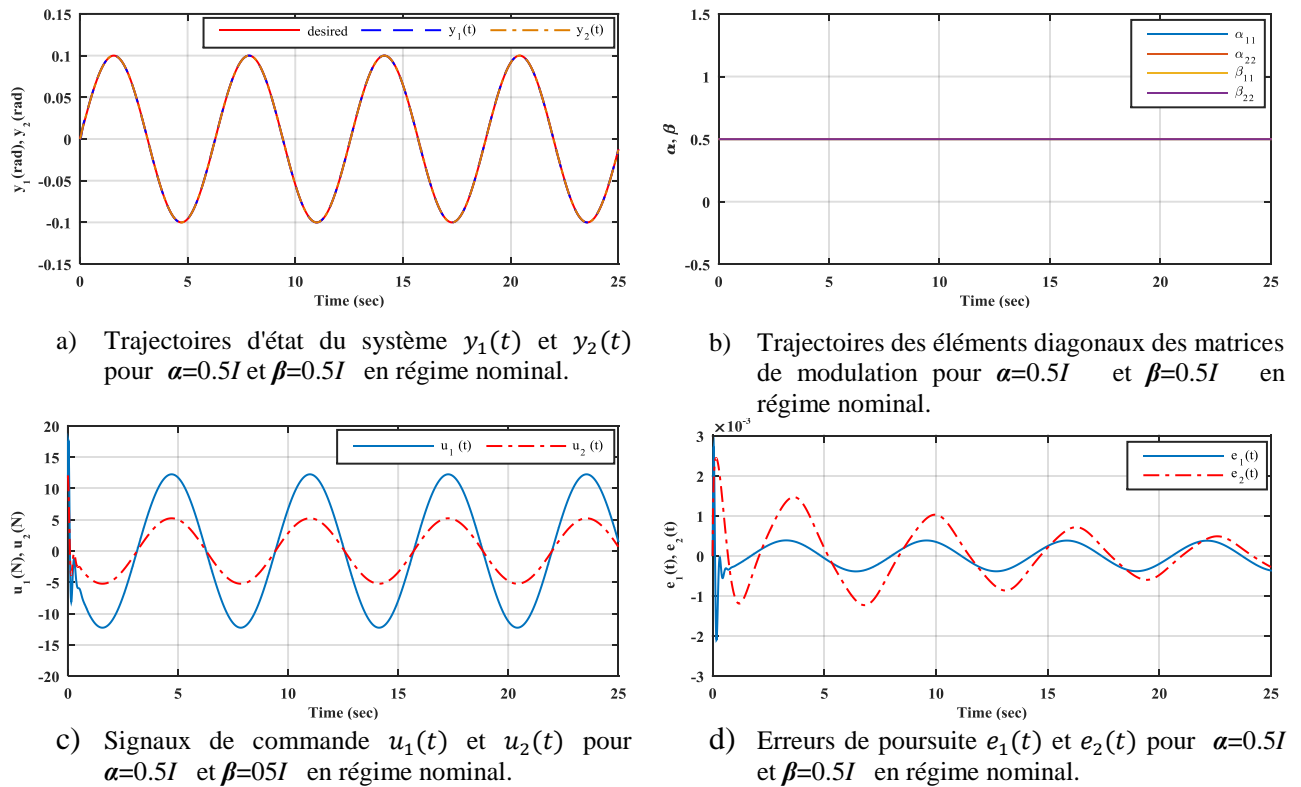


Figure 4.17 : Performances du contrôleur FS-WNN hybride pour $\alpha=0.5I$ et $\beta=0.5I$

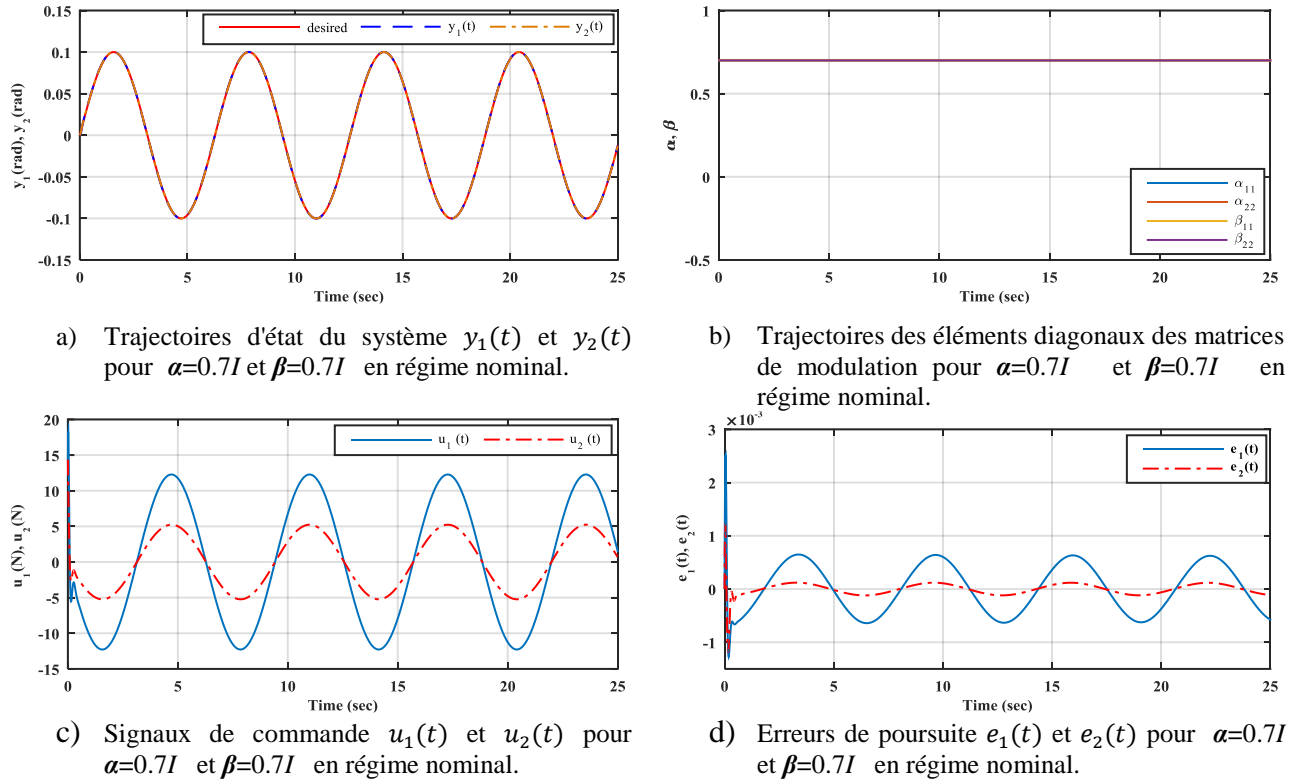


Figure 4.18 : Performances du contrôleur FS-WNN hybride pour $\alpha=0.7I$ et $\beta=0.7I$

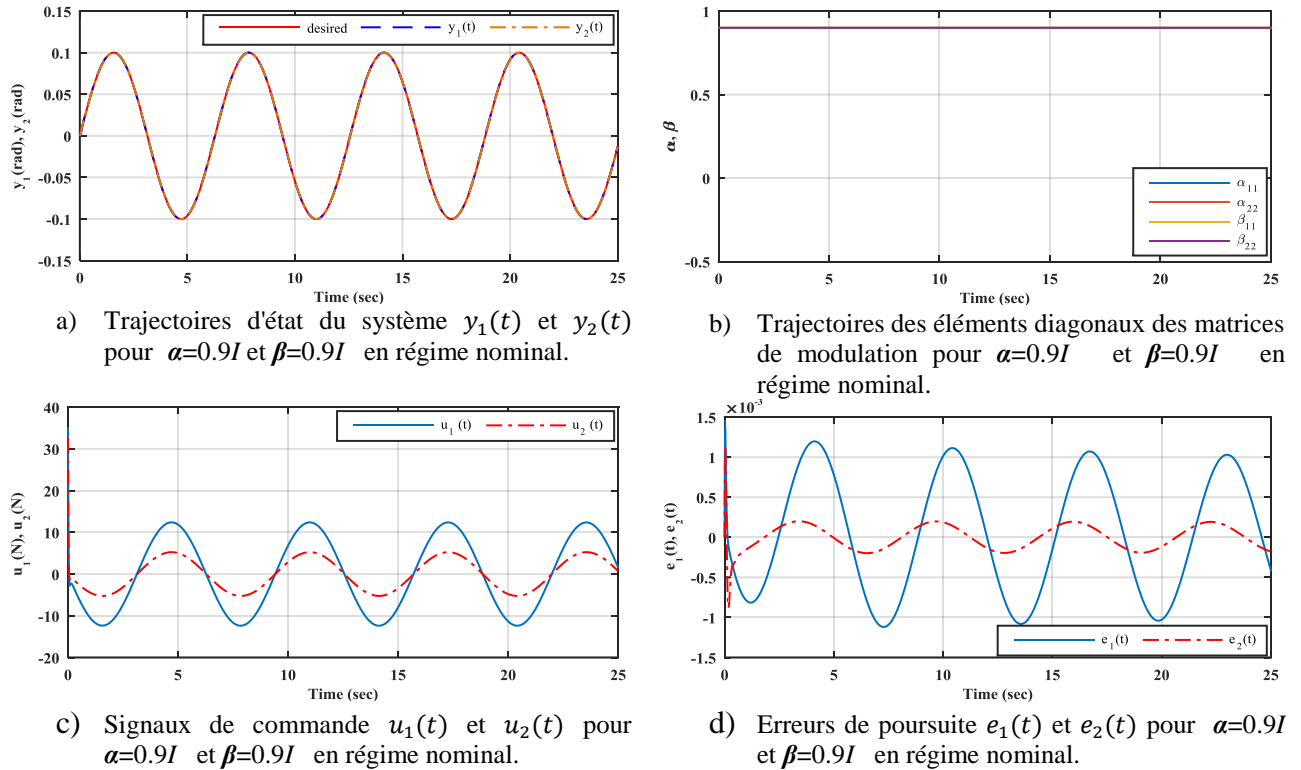
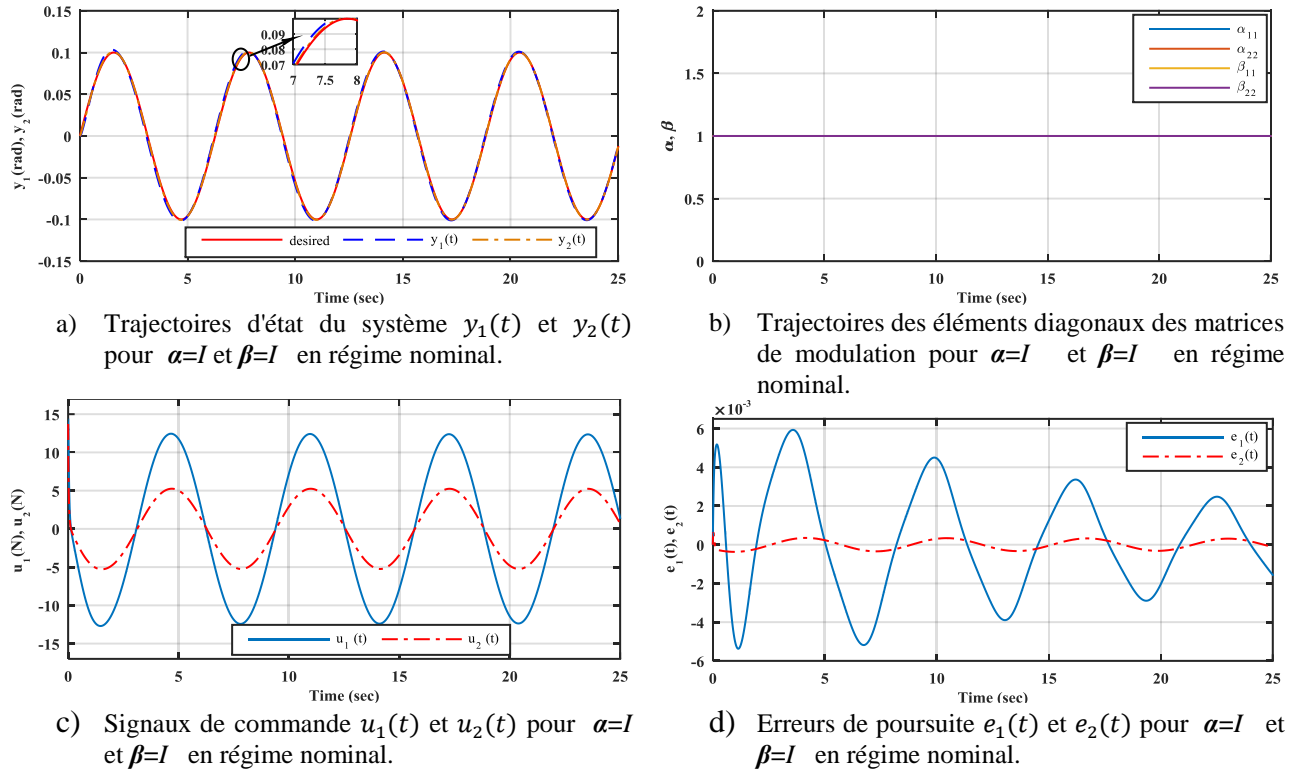


Figure 4.19 : Performances du contrôleur FS-WNN hybride pour $\alpha=0.9I$ et $\beta=0.9I$


 Figure 4.20 : Performances du contrôleur FS-WNN hybride pour $\alpha=1$ et $\beta=1$

Les performances du contrôleur FS-WNN hybride ont été présentés dans les Figures de 4.14 à 4.20 pour différentes valeurs des paramètres de modulation α et β en régime nominal. A partir de la Figure 4.14 (a) jusqu'à la Figure 4.20 (a), on remarque que le contrôleur FS-WNN hybride a pu assurer des bonnes performances de poursuite quelle que soit la valeur de α et β . Sauf pour le cas $\alpha=1$ et $\beta=1$, le contrôleur flou contrôle seul le système dynamique, on remarque qu'il maintient une petite erreur statique dans la Figure 4.20 (a). D'autre part, le signal de commande est suffisamment lisse et borné sans changement brusques pour $\alpha=[0.1 \ 1]$ et $\beta=[0.1 \ 1]$. Pour le cas $\alpha=0$ et $\beta=0$, le contrôleur WNN contrôle seul le système dynamique, on remarque des pics importants dans le signal de commande au départ et à l'instant $t=21$ sec. Ce qui confirme notre conclusion à propos de la forte non linéarité et la sensibilité des réseaux de neurones à base d'ondelettes. Chose qu'on peut remarquer à partir de la Figure 4.14(d) jusqu'à la Figure 4.20(d) qui montrent le signal d'erreur de poursuite.

A partir de cette analyse qualitative, on constate que pour avoir plus de précision il faut utiliser les deux systèmes intelligents simultanément en donnant plus d'importance au système WNN, et pour avoir un signal de commande lisse avec plus de robustesse plus d'importance doit être accordée au système flou. Cela crée un compromis entre les deux systèmes intelligents. Cette

explication nous aide à comprendre le fonctionnement du système hybride, mais ça ne résout pas le problème de détermination des paramètres de modulation α et β . Une analyse quantitative est nécessaire dans ce cas-là.

Dans le but de choisir les bons paramètres de modulation α et β . Une analyse des performances du contrôleur FS-WNN hybride est présentée dans le Tableau 4.4. Cette analyse concerne le fonctionnement du contrôleur FS-WNN hybride dans des conditions nominales pour différentes valeurs des paramètres de modulation α et β . Les critères de sélection sont la moyenne de l'erreur quadratique MSE et la valeur absolue de la moyenne de l'erreur de poursuite MAE et pour le signal de commande on prend l'énergie E du signal de commande comme critère de sélection pour les deux articulations. L'expression de la valeur absolue de la moyenne de l'erreur de poursuite MAE est donnée par :

$$MAE = \frac{|\sum_{i=1}^N y_{di} - y_i|}{N}$$

Tableau 4.4 : Comparaison des performances du contrôleur FS-WNN adaptatif indirect en régime nominale pour différentes valeurs des paramètres de modulation.

Paramètres de modulation	Link 1			Link 2		
	MSE	MAE	$E(u_1)$	MSE	MAE	$E(u_2)$
0	1.0795x10 ⁻⁶	3.8158x10 ⁻⁵	3.3923x10 ⁵	2.2001x10 ⁻⁷	1.9111x10 ⁻⁵	7.1986x10 ⁴
0.1	6.2772x10 ⁻⁶	1.9221x10 ⁻⁵	1.9184x10 ⁵	3.1937x10 ⁻⁹	5.1767x10 ⁻⁷	3.5958x10 ⁴
0.3	5.5839x10⁻⁸	3.4841x10⁻⁶	1.9127x10⁵	8.7400x10⁻⁹	9.6967x10⁻⁷	3.5191x10⁴
0.5	1.0125x10 ⁻⁷	5.3465x10 ⁻⁶	1.9117x10 ⁵	4.9786x10 ⁻⁷	1.1610x10 ⁻⁷	3.5106x10 ⁴
0.7	2.1470x10 ⁻⁷	1.2085x10 ⁻⁵	1.9123x10 ⁵	1.4741x10 ⁻⁸	7.512 x10 ⁻⁷	3.5100x10 ⁴
0.9	5.6487x10 ⁻⁷	4.8522x10 ⁻⁵	1.9483x10 ⁵	2.5569x10 ⁻⁸	1.6783x10 ⁻⁶	3.6183x10 ⁴
1	8.0671x10 ⁻⁶	1.9447x10 ⁻⁴	1.9567x10 ⁵	5.7001x10 ⁻⁸	7.2898x10 ⁻⁵	3.5096x10 ⁴

En observant le tableau 4.4 pour la première articulation, on remarque que l'erreur de poursuite MSE diminue au fur et à mesure qu'on augmente α et β jusqu'à ce qu'elle atteigne un pic minimal pour $\alpha=\beta=0.3I$. Après, elle re-augmente progressivement pour des valeurs de α et β supérieures à $0.3I$. idem pour l'erreur MAE , la meilleure valeur est obtenue lorsque $\alpha=\beta=0.3I$ pour la première articulation. L'énergie de signal de commande est presque identique pour toutes les valeurs de α et β . Cependant, l'effort de contrôle minimal est obtenu dans le cas où $\alpha=\beta=0.3I$ également.

En ce qui concerne la deuxième articulation, les meilleures valeurs de l'erreur de poursuite sont obtenues dans le cas où $\alpha=\beta=0.1I$ pour le critère de MSE , dans le cas où $\alpha=\beta=0.5I$ pour le critère de MAE et de l'énergie de signal de commande. Donc les meilleures performances sont obtenues pour des valeurs de α et β dans l'intervalle $[0.1 \ 0.5]$, ce qui serait judicieux de prendre la moyenne qui est 0.3 .

5. Conclusion

Dans ce chapitre, on a proposé un nouveau schéma de commande pour la modélisation et le contrôle des systèmes MIMO non linéaires incertain en utilisant un nouvel approximateur FS-WNN hybride. Dans le schéma de commande proposé dans ce chapitre, nous avons modulé l'approximateur WNN présenté dans le chapitre 02 avec l'approximateur FS présenté dans le chapitre 03 pour obtenir un nouveau approximateur hybride qui permet de bénéficier des avantages des deux approximateurs. Chaque approximateur fournit une approximation différente pour chaque fonction inconnue du modèle du système. Ensuite, les différentes approximations d'une même fonction sont combinées à l'aide de la technique de modulation. La technique de modulation nous a permis de choisir le degré de contribution de chaque approximateur par le concepteur. Les paramètres de modulation ont été fixés à priori par le concepteur par tâtonnement. La méthode directe de Lyapunov a été utilisée pour dériver les lois d'adaptation des paramètres du FS-WNN et pour prouver la stabilité asymptotique du système de contrôle. Des tests de simulation ont été établis et une étude comparative est réalisée entre le FS, le WNN et le nouveau approximateur FS-WNN hybride. Les résultats de simulation ont montré que les contrôleurs synthétisés à l'aide de l'approximateur FS-WNN hybride et de l'approximateur WNN ont donné la meilleure poursuite. Toutefois, le contrôleur basé sur le FS-WNN a surpassé le contrôleur basé sur le WNN. En outre, les contrôleurs basés sur le FS et le FS-WNN ont démontré une bonne robustesse contre les perturbations et les variations paramétriques du système, sans présenter de changement brusque

dans le signal de commande. Contrairement au WNN qui a engendré des changements brusques dans le signal de contrôle après l'application des perturbations.

Par conséquent, il a été démontré par cette étude que l'approximateur FS-WNN hybride est plus performant que les approximateurs WNN et FS. La raison est que le FS-WNN hybride a bénéficié des avantages du système FS et du système WNN en même temps, c'est-à-dire:

- La robustesse et l'insensibilité du FS face aux variations paramétriques et perturbations externes,
- L'auto-ajustement en ligne des paramètres du WNN (les centres et les paramètres de dilatation des fonctions d'activation ondelettes et les poids de sortie).

Ces avantages ont permis au nouveau approximateur FS-WNN d'améliorer la précision de poursuite et de garantir un signal de commande lisse même en présence d'incertitudes paramétriques et de perturbations externes.

A partir de l'étude comparative effectuée à la fin de ce chapitre, nous avons constaté que pour ce système et ces conditions de fonctionnement, la meilleure valeur des paramètres de modulation α et β est 0.31. cette valeur doit être recalculée si les conditions de fonctionnements ou le cahier de charge change. Ce qui nous incite à considérer l'optimisation de ces paramètres de modulation en ligne. Il est nécessaire dans ce cas d'automatiser le choix des paramètres de modulation pour assurer l'optimalité de ce choix et pour atteindre les meilleures performances du contrôleur FS-WNN hybride.

Chapitre 5

Commande adaptative indirecte par FS-WNN hybride-variable des systèmes MIMO incertains

1. Introduction

L'hybridation de plusieurs système intelligent est une méthode qui a prouvé son efficacité dans l'amélioration de la qualité d'approximation et de contrôle intelligent des systèmes incertains. Cette hybridation permet de regrouper les avantages de différents approximateurs intelligents et de surmonter les limitations individuelles de chacun de ces approximateurs. Le nouveau approximateur FS-WNN hybride présenté dans le chapitre précédent est le résultat d'une hybridation entre les systèmes flous et les WNNs en utilisant la technique de modulation.

La technique de modulation est une technique d'hybridation qui associe à chaque approximateur un facteur qui contrôle son degré de contribution. Cette technique a été exploitée par plusieurs chercheurs dans différents domaines. Dian et al. [96] ont proposé un contrôleur backstepping adaptatif indirect basé sur un approximateur neuro-flou de type 2 pour commander un robot manipulateur flexible. L'approximateur neuro-flou de type 2 est constitué des fonctions d'appartenances floues supérieures et inférieures. La sortie du système neuro-flou de type 2 calculée en utilisant les fonctions d'appartenances inférieures est combinée avec la sortie qui est calculée en utilisant des paramètres de modulation ($\alpha, 1-\alpha$). Le rôle du facteur de modulation ici est de choisir quel type de fonctions d'appartenance aura plus d'effet sur le calcul de la sortie de l'approximateur neuro-flou de type 2. Wang et al. [99] ont développé un contrôleur hybride neuro-flou à base d'un observateur et d'un contrôleur de supervision. Ce contrôleur est le résultat d'une hybridation entre un contrôleur neuro-flou adaptatif direct/indirect. L'hybridation est assurée grâce à la technique de modulation. Les paramètres de modulation sont sélectionnés par tâtonnement entre les connaissances sur les dynamiques du système et celles du contrôleur. Le contrôleur de supervision est utilisé en cascade avec le contrôleur neuro-flou hybride, et il a pour rôle de soutenir le contrôleur FNN dans le cas où le système sous contrôle par le contrôleur neuro-flou tend vers

l'instabilité. L'intérêt de cette hybridation est de grouper les avantages de contrôleur adaptatif direct et indirect et les caractéristiques des systèmes flous dans le même schéma de commande. Néanmoins, les paramètres de modulation α et $(1-\alpha)$ ont été fixés préalablement par le concepteur, et le problème de l'ajustement en ligne des paramètres de modulation n'a pas été abordé dans leur travail.

Dans le domaine de la reconnaissance d'images, Cho [100] a proposé un système neuro-flou hybride pour la reconnaissance en ligne des caractères manuscrits. Ce système de reconnaissance est basé sur l'hybridation de plusieurs réseaux de neurones modulés en utilisant la technique de l'agrégation floue intégrale [101]. Les réseaux de neurones dans ce système ont été synthétisés en utilisant l'algorithme de back propagation selon une base de données entrée/sortie répartie sur les différents réseaux. Le degré d'importance (facteur de modulation) de chaque réseau de neurones dans la prise de décision est déterminé en fonction des performances de ce réseau sur les données de validation. Cho a constaté que les performances des sorties combinées sont de loin meilleures que les performances des réseaux de neurones séparés. Selon Cho, cette combinaison a permis de réduire significativement l'erreur quadratique car les réseaux de neurones individuels sont de nature complémentaire. Cho a donné des indications sur l'établissement des paramètres de modulation et a précisé que le choix de ces paramètres dépend fortement du problème traité. Il a de même spécifié que les performances du système de reconnaissance neuro-flou peut être vachement améliorées avec un bon choix des paramètres de modulation α , β , ...etc. Cependant, l'approche d'agrégation des systèmes neuro-flous et de choix des paramètres de modulation proposée par Cho dans le domaine de la reconnaissance optique est basée principalement sur les données entrée/sortie qui ne sont pas disponibles dans la majorité des problèmes de commande des systèmes dynamiques.

Wang et al. [102] ont proposé un contrôleur neuro-flou direct/indirect hybride à base d'observateur pour la commande des systèmes MIMO incertains. L'hybridation entre le contrôleur adaptatif direct et indirect a été réalisée à l'aide d'un paramètre de modulation α . La valeur de α a été fixée selon la valeur absolue de l'estimation de la fonction du système $g(x)$. ou α aurait une valeur positive si $|\hat{g}(x)|$ est supérieure à une constante positive ε , sinon elle sera égale à 0. Cependant, le choix des paramètres de modulation doit prendre en considération le problème de contrôle dans sa globalité. Non seulement la valeur absolue de $g(x)$. De même, un ajustement du paramètre α en ligne améliora considérablement les performances de contrôle.

Dans cette thèse on propose un nouveau système FS-WNN hybride. Comme on a vu dans le chapitre précédent, le nouveau approximateur FS-WNN hybride est le résultat de l'hybridation du système WNN de chapitre deux avec le système FS de chapitre trois. La technique de la modulation est utilisée pour combiner les approximations des deux systèmes intelligents. Par définition, la technique de modulation attribue un degré de contribution α constant à l'approximation $F(x)$ issue du système flou et elle attribue le complément de α qui est $(1 - \alpha)$ à l'approximation $F(x)$ issue du système WNN de la même fonction. La fonction $G(x)$ du système incertain est aussi approximée de la même manière en utilisant les facteurs de modulation $(\beta, 1-\beta)$. Les paramètres α et β sont dans ce cas-là fixés à priori par le concepteur.

Le choix des paramètres de modulation α et β se fait en effectuant plusieurs tests par simulation pour sélectionner la valeur des constantes α et β qui donnent les meilleures performances de l'approximateur FS-WNN hybride. La détermination de la valeur optimale des constantes α et β n'est par ailleurs pas une tâche facile. La procédure de détermination de la valeur optimale de α et β est souvent longue et compliquée. Une fois sélectionnées par tâtonnement, il n'y a aucune garantie que le choix effectué est optimal. Il est à noter de même que la valeur des paramètres α et β reste fixe et ne s'adapte pas en temps réel pour suivre l'adaptation des paramètres du système flou et de réseau de neurones dans le but de maîtriser le degré de contribution de chaque approximateur. Cela limite les performances de l'approximateur FS-WNN hybride face aux perturbations externes et aux incertitudes.

Dans ce chapitre, nous développons un mécanisme d'ajustement automatique des paramètres de modulation pour le nouveau approximateur FS-WNN hybride. Cet algorithme permet de surmonter le problème de sélection et d'ajustement des paramètres de modulation par le concepteur. Dans la synthèse du contrôleur FS-WNN hybride dans le chapitre précédent, le choix des paramètres de modulation a été fixé par le concepteur à priori dans un intervalle entre 0 et 1 par tâtonnement. Néanmoins, pour atteindre une qualité d'approximation optimale du nouveau approximateur FS-WNN hybride, nous proposons que les facteurs de modulation soient ajustés en ligne pour minimiser l'erreur de modélisation tout en assurant la stabilité du système de commande.

Les étapes de développement de l'algorithme d'ajustement en ligne des facteurs de modulation sont présentées dans ce chapitre et la validation par simulation de l'efficacité de l'algorithme est effectuée sur un robot manipulateur. On compare davantage les résultats obtenus par le nouveau approximateur FS-WNN hybride avec des facteurs de modulation ajustable en ligne aux résultats

issues de l'approximateur FS-WNN avec des paramètres de modulation fixes. On établit des conclusions sur l'intérêt d'adaptation des degrés de contribution de chaque approximateur en temps réel. L'approche développée est basée sur la théorie de Lyapunov pour ajuster automatiquement et en temps réel le degré de contribution de chaque système intelligent. Cette approche permet d'obtenir un approximateur FS-WNN hybride-variable. Ce qui améliore considérablement les performances de l'approximateur FS-WNN hybride et offre plus de robustesse et de stabilité au système de contrôle. L'Architecture de l'approximateur FS-WNN hybride-variable dans la boucle de commande d'un système non linéaire incertains est donnée par la Figure 5.1 comme suit :

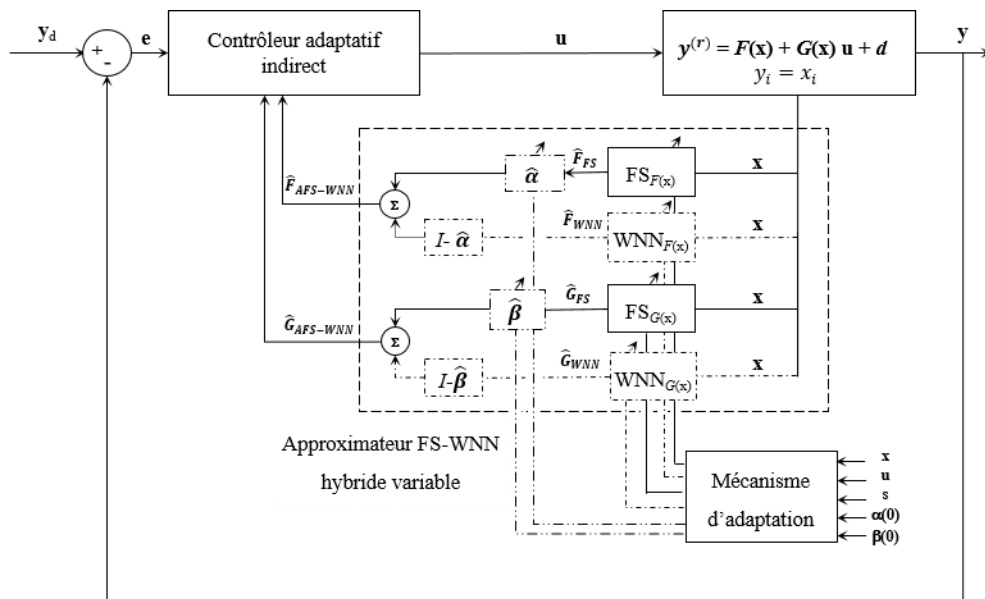


Figure 5.1 : Architecture de l'approximateur FS-WNN hybride-variable dans une boucle de commande.

L'approximation donnée par le FS-WNN hybride-variable combine les avantages du FS et du WNN, elle réunit l'intégration des connaissances de l'expert humain et les données d'entrée-sortie dans une seule structure. De plus, elle permet d'ajuster les degrés de contribution du système FS et du WNN automatiquement en temps réel pour atteindre une approximation hybride optimale des fonctions inconnues du système.

Le principe d'ajustement des degrés de contribution du FS et du WNN est le suivant : chacun des deux systèmes intelligents génère une approximation distincte des fonctions non linéaires inconnues du système à commander. Les deux approximations sont combinées à l'aide des paramètres de modulation $(\hat{\alpha}, I - \hat{\alpha})$ et $(\hat{\beta}, I - \hat{\beta})$, où $\hat{\alpha}$ et $\hat{\beta}$ sont des matrices carrées diagonales avec

des valeurs -variables en temps réel, et I est la matrice d'identité. Attribuer un coefficient de modulation adaptatif à l'approximation de chaque approximateur permet de balancer idéalement entre les deux approximateurs en ajustant automatiquement les éléments de $\hat{\alpha}$ et $\hat{\beta}$ dans l'intervalle [0 1].

De manière générale, les fonctions inconnues du système $F(x)$ et $G(x)$ sont approximées comme suit :

$$\begin{aligned} F(x) &= \alpha F_{FS} + (I - \alpha) F_{WNN} \\ G(x) &= \beta G_{FS} + (I - \beta) G_{WNN} \end{aligned} \quad (5.1)$$

En développant (5.1) et en utilisant les approximations des fonctions $F(x)$ et $G(x)$ donnés par le nouveau approximateur hybride-variable FS-WNN, on obtient :

$$\begin{aligned} F(x) &= \alpha^* \left(F_{FS}^*(x, \Theta_F^*) + \varepsilon_{F_{FS}}(x) \right) + (I - \alpha^*) \left(F_{WNN}^*(x, W_F^*, C_F^*) + \varepsilon_{F_{WNN}}(x) \right) \\ &= F_{AFS-WNN}^*(x, \alpha^*, \Theta_F^*, W_F^*, C_F^*) + \varepsilon_F(x, \alpha^*) \end{aligned} \quad (5.2)$$

$$\begin{aligned} G(x) &= \beta^* \left(G_{FS}^*(x, \Theta_G^*) + \varepsilon_{G_{FS}}(x) \right) + (I - \beta^*) \left(G_{WNN}^*(x, W_G^*, C_G^*) + \varepsilon_{G_{WNN}}(x) \right) \\ &= G_{FS-WNN}^*(x, \beta^*, \Theta_G^*, W_G^*, C_G^*) + \varepsilon_G(x, \beta^*) \end{aligned} \quad (5.3)$$

Avec :

$$\begin{aligned} \varepsilon_F(x, \alpha^*) &= \alpha^* \varepsilon_{F_{FS}}(x) + (I - \alpha^*) \varepsilon_{F_{WNN}}(x) \\ \varepsilon_G(x, \beta^*) &= \beta^* \varepsilon_{G_{FS}}(x) + (I - \beta^*) \varepsilon_{G_{WNN}}(x) \end{aligned}$$

Pratiquement, les paramètres idéaux α^*, Θ^*, W^* et C^* ne peuvent pas être déterminés, des approximations $\hat{\alpha}, \hat{\Theta}, \hat{W}$ et \hat{C} de ces paramètres idéaux peuvent être utilisées. Par conséquent, les fonctions inconnues du système $F(x)$ et $G(x)$ sont approximées comme suit:

$$\hat{F}_{AFS-WNN}(x, \hat{\alpha}, \hat{\Theta}_F, \hat{W}_F, \hat{C}_F) = \hat{\alpha} \hat{F}_{FS}(x, \hat{\Theta}_F) + (I - \hat{\alpha}) \hat{F}_{WNN}(x, \hat{W}_F, \hat{C}_F) \quad (5.4)$$

$$\hat{G}_{FS-WNN}(x, \hat{\beta}, \hat{\Theta}_G, \hat{W}_G, \hat{C}_G) = \hat{\beta} \hat{G}_{FS}(x, \hat{\Theta}_G) + (I - \hat{\beta}) \hat{G}_{WNN}(x, \hat{W}_G, \hat{C}_G) \quad (5.5)$$

Les paramètres de modulation $(\hat{\alpha}, I-\hat{\alpha})$ et $(\hat{\beta}, I-\hat{\beta})$ sont dans ce cas-là ajustés d'une façon automatique et en temps réel pour s'adapter aux évolutions des paramètres des approximateurs et pour faire face aux perturbations externes et incertitudes. En admettant que les paramètres de modulation représentent le degré de contribution de chaque approximateur, donc le degré de contribution de chaque approximateur est déterminé par un algorithme de minimisation d'une fonction coût selon la théorie du Lyapunov. Les étapes de développement d'un contrôleur adaptatif indirect basé sur l'approximateur FS-WNN hybride-variable pour le contrôle des systèmes MIMO incertains sont présentées dans ce qui suit.

2. Commande adaptative indirecte par FS-WNN hybride-variable:

Considérant le système MIMO non linéaires incertains (2.11) avec des fonctions inconnues $F(x)$ et $G(x)$ qui seront approximées par le nouveau approximateur universel FS-WNN hybride-variable de la Figure 5.1, leurs expressions sont données par les équations (5.4) et (5.5). Donc la procédure de synthèse de la nouvelle loi de commande basée sur l'approximateur FS-WNN hybride-variable est présentée comme suit :

On reprend l'équation de la dérivée de l'erreur filtrée (2.17) et on en rajoute et soustrait le terme $\hat{G}_{AFS-WNN}(x, \hat{\beta}, \hat{\Theta}_G, \hat{W}_G, \hat{C}_G)u$, on aura par la suite:

$$\begin{aligned} \dot{s} = & v - F(x) - \left(G(x) - \hat{G}_{AFS-WNN}(x, \hat{\beta}, \hat{\Theta}_G, \hat{W}_G, \hat{C}_G) \right) u - d \\ & - \hat{G}_{AFS-WNN}(x, \hat{\beta}, \hat{\Theta}_G, \hat{W}_G, \hat{C}_G) u \end{aligned} \quad (5.6)$$

La nouvelle loi de commande est synthétisée comme suit :

$$u = \hat{G}_{AFS-WNN}(x, \hat{\beta}, \hat{\Theta}_G, \hat{W}_G, \hat{C}_G)^{-1} \left(v + u_r - \hat{F}_{AFS-WNN}(x, \hat{\alpha}, \hat{\Theta}_F, \hat{W}_F, \hat{C}_F) \right) \quad (5.7)$$

En substituant (5.7) dans (5.6), on obtient :

$$\begin{aligned} \dot{s} = & - \left(F(x) - \hat{F}_{AFS-WNN}(x, \hat{\alpha}, \hat{\Theta}_F, \hat{W}_F, \hat{C}_F) \right) \\ & - \left(G(x) - \hat{G}_{AFS-WNN}(x, \hat{\beta}, \hat{\Theta}_G, \hat{W}_G, \hat{C}_G) \right) u - u_r - d \end{aligned} \quad (5.8)$$

Et en utilisant les expressions (5.2)-(5.3) et (5.4)-(5.5) dans (5.8), on obtient :

$$\begin{aligned}
 \dot{s} = & - \left[\alpha^* F_{FS}^*(x, \Theta_F^*) + (I - \alpha^*) F_{WNN}^*(x, W_F^*, C_F^*) + \varepsilon_F(x, \alpha^*) \right. \\
 & \left. - \left(\hat{\alpha} \hat{F}_{FS}(x, \hat{\Theta}_F) + (I - \hat{\alpha}) \hat{F}_{WNN}(x, \hat{W}_F, \hat{C}_F) \right) \right] \\
 & - \left[\beta^* G_{FS}^*(x, \Theta_G^*) + (I - \beta^*) G_{WNN}^*(x, W_G^*, C_G^*) + \varepsilon_G(x, \beta^*) \right. \\
 & \left. - \left(\hat{\beta} \hat{G}_{FS}(x, \hat{\Theta}_G) + (I - \hat{\beta}) \hat{G}_{WNN}(x, \hat{W}_G, \hat{C}_G) \right) \right] \mathbf{u} - \mathbf{u}_r - \mathbf{d}
 \end{aligned} \tag{5.9}$$

On peut mettre (5.9) sous la forme suivante :

$$\begin{aligned}
 \dot{s} = & - \left[\alpha^* F_{FS}^*(x, \Theta_F^*) - \hat{\alpha} \hat{F}_{FS}(x, \hat{\Theta}_F) + (I - \alpha^*) F_{WNN}^*(x, W_F^*, C_F^*) \right. \\
 & \left. - (I - \hat{\alpha}) \hat{F}_{WNN}(x, \hat{W}_F, \hat{C}_F) + \varepsilon_F(x, \alpha^*) \right] \\
 & - \left[\beta^* G_{FS}^*(x, \Theta_G^*) - \hat{\beta} \hat{G}_{FS}(x, \hat{\Theta}_G) + (I - \beta^*) G_{WNN}^*(x, W_G^*, C_G^*) \right. \\
 & \left. - (I - \hat{\beta}) \hat{G}_{WNN}(x, \hat{W}_G, \hat{C}_G) + \varepsilon_G(x, \beta^*) \right] \mathbf{u} - \mathbf{u}_r - \mathbf{d}
 \end{aligned} \tag{5.10}$$

Admettant que $\alpha^* = \hat{\alpha} + \tilde{\alpha}$ et $\beta^* = \hat{\beta} + \tilde{\beta}$, l'équation (5.10) peut être développée comme suit :

$$\begin{aligned}
 \dot{s} = & - \left[\hat{\alpha} F_{FS}^*(x, \Theta_F^*) + \tilde{\alpha} F_{FS}^*(x, \Theta_F^*) - \hat{\alpha} \hat{F}_{FS}(x, \hat{\Theta}_F) + (I - \alpha^*) F_{WNN}^*(x, W_F^*, C_F^*) \right. \\
 & \left. - (I - \hat{\alpha}) \hat{F}_{WNN}(x, \hat{W}_F, \hat{C}_F) + \varepsilon_F(x, \alpha^*) \right] \\
 & - \left[\hat{\beta} G_{FS}^*(x, \Theta_G^*) + \tilde{\beta} G_{FS}^*(x, \Theta_G^*) - \hat{\beta} \hat{G}_{FS}(x, \hat{\Theta}_G) + (I - \beta^*) G_{WNN}^*(x, W_G^*, C_G^*) \right. \\
 & \left. - (I - \hat{\beta}) \hat{G}_{WNN}(x, \hat{W}_G, \hat{C}_G) + \varepsilon_G(x, \beta^*) \right] \mathbf{u} - \mathbf{u}_r - \mathbf{d}
 \end{aligned} \tag{5.11}$$

Sachant que $F_{FS}^*(x, \Theta_F^*) = \hat{F}_{FS}(x, \hat{\Theta}_F) + \tilde{F}_{FS}(x, \tilde{\Theta}_F)$ et $\hat{\beta} G_{FS}^*(x, \Theta_G^*) = \hat{\beta} \hat{G}_{FS}(x, \hat{\Theta}_G) + \hat{\beta} \tilde{G}_{FS}(x, \tilde{\Theta}_G)$, de (5.11) on obtient :

$$\begin{aligned}
 \dot{\mathbf{s}} = & -[\hat{\alpha}\hat{\mathbf{F}}_{FS}(\mathbf{x}, \hat{\Theta}_F) + \hat{\alpha}\tilde{\mathbf{F}}_{FS}(\mathbf{x}, \tilde{\Theta}_F) + \tilde{\alpha}\mathbf{F}_{FS}^*(\mathbf{x}, \Theta_F^*) - \hat{\alpha}\hat{\mathbf{F}}_{FS}(\mathbf{x}, \hat{\Theta}_F) \\
 & + (I - \alpha^*)\mathbf{F}_{WNN}^*(\mathbf{x}, \mathbf{W}_F^*, \mathbf{C}_F^*) - (I - \hat{\alpha})\hat{\mathbf{F}}_{WNN}(\mathbf{x}, \hat{\mathbf{W}}_F, \hat{\mathbf{C}}_F) + \boldsymbol{\varepsilon}_F(\mathbf{x}, \alpha^*)] \\
 & - [\hat{\beta}\hat{\mathbf{G}}_{FS}(\mathbf{x}, \hat{\Theta}_G) + \hat{\beta}\tilde{\mathbf{G}}_{FS}(\mathbf{x}, \tilde{\Theta}_G) + \tilde{\beta}\mathbf{G}_{FS}^*(\mathbf{x}, \Theta_G^*) - \hat{\beta}\hat{\mathbf{G}}_{FS}(\mathbf{x}, \hat{\Theta}_G) \\
 & + (I - \beta^*)\mathbf{G}_{WNN}^*(\mathbf{x}, \mathbf{W}_G^*, \mathbf{C}_G^*) - (I - \hat{\beta})\hat{\mathbf{G}}_{WNN}(\mathbf{x}, \hat{\mathbf{W}}_G, \hat{\mathbf{C}}_G) + \boldsymbol{\varepsilon}_G(\mathbf{x}, \beta^*)]\mathbf{u} - \mathbf{u}_r - \mathbf{d}
 \end{aligned} \tag{5.12}$$

Après avoir éliminé le terme $\hat{\alpha}\hat{\mathbf{F}}_{FS}(\mathbf{x}, \hat{\Theta}_F)$ et $\hat{\beta}\hat{\mathbf{G}}_{FS}(\mathbf{x}, \hat{\Theta}_G)$, l'expression (5.12) devient :

$$\begin{aligned}
 \dot{\mathbf{s}} = & -[\hat{\alpha}\tilde{\mathbf{F}}_{FS}(\mathbf{x}, \tilde{\Theta}_F) + \tilde{\alpha}\mathbf{F}_{FS}^*(\mathbf{x}, \Theta_F^*) + (I - \alpha^*)\mathbf{F}_{WNN}^*(\mathbf{x}, \mathbf{W}_F^*, \mathbf{C}_F^*) \\
 & - (I - \hat{\alpha})\hat{\mathbf{F}}_{WNN}(\mathbf{x}, \hat{\mathbf{W}}_F, \hat{\mathbf{C}}_F) + \boldsymbol{\varepsilon}_F(\mathbf{x}, \alpha^*)] \\
 & - [\hat{\beta}\tilde{\mathbf{G}}_{FS}(\mathbf{x}, \tilde{\Theta}_G) + \tilde{\beta}\mathbf{G}_{FS}^*(\mathbf{x}, \Theta_G^*) + (I - \beta^*)\mathbf{G}_{WNN}^*(\mathbf{x}, \mathbf{W}_G^*, \mathbf{C}_G^*) \\
 & - (I - \hat{\beta})\hat{\mathbf{G}}_{WNN}(\mathbf{x}, \hat{\mathbf{W}}_G, \hat{\mathbf{C}}_G) + \boldsymbol{\varepsilon}_G(\mathbf{x}, \beta^*)]\mathbf{u} - \mathbf{u}_r - \mathbf{d}
 \end{aligned} \tag{5.13}$$

De la même manière que (5.12), on peut développer (5.13) sous la forme suivante :

$$\begin{aligned}
 \dot{\mathbf{s}} = & -[\hat{\alpha}\tilde{\mathbf{F}}_{FS}(\mathbf{x}, \tilde{\Theta}_F) + \tilde{\alpha}\hat{\mathbf{F}}_{FS}(\mathbf{x}, \hat{\Theta}_F) + \tilde{\alpha}\tilde{\mathbf{F}}_{FS}(\mathbf{x}, \tilde{\Theta}_F) + (I - \alpha^*)\mathbf{F}_{WNN}^*(\mathbf{x}, \mathbf{W}_F^*, \mathbf{C}_F^*) \\
 & - (I - \hat{\alpha})\hat{\mathbf{F}}_{WNN}(\mathbf{x}, \hat{\mathbf{W}}_F, \hat{\mathbf{C}}_F) + \boldsymbol{\varepsilon}_F(\mathbf{x}, \alpha^*)] \\
 & - [\hat{\beta}\tilde{\mathbf{G}}_{FS}(\mathbf{x}, \tilde{\Theta}_G) + \tilde{\beta}\hat{\mathbf{G}}_{FS}(\mathbf{x}, \hat{\Theta}_G) + \tilde{\beta}\tilde{\mathbf{G}}_{FS}(\mathbf{x}, \tilde{\Theta}_G) + (I - \beta^*)\mathbf{G}_{WNN}^*(\mathbf{x}, \mathbf{W}_G^*, \mathbf{C}_G^*) \\
 & - (I - \hat{\beta})\hat{\mathbf{G}}_{WNN}(\mathbf{x}, \hat{\mathbf{W}}_G, \hat{\mathbf{C}}_G) + \boldsymbol{\varepsilon}_G(\mathbf{x}, \beta^*)]\mathbf{u} - \mathbf{u}_r - \mathbf{d}
 \end{aligned} \tag{5.14}$$

On peut ainsi écrire (5.14) sous forme développée comme suit :

$$\begin{aligned}
 \dot{s} = & -[\hat{\alpha}\tilde{F}_{FS}(x, \tilde{\Theta}_F) + \tilde{\alpha}\hat{F}_{FS}(x, \hat{\Theta}_F) + \tilde{\alpha}\tilde{F}_{FS}(x, \tilde{\Theta}_F) + F_{WNN}^*(x, W_F^*, C_F^*) \\
 & - \alpha^* F_{WNN}^*(x, W_F^*, C_F^*) + \hat{\alpha}\hat{F}_{WNN}(x, \widehat{W}_F, \widehat{C}_F) - \hat{F}_{WNN}(x, \widehat{W}_F, \widehat{C}_F) + \varepsilon_F(x, \alpha^*)] \\
 & - [\hat{\beta}\tilde{G}_{FS}(x, \tilde{\Theta}_G) + \tilde{\beta}\hat{G}_{FS}(x, \hat{\Theta}_G) + \tilde{\beta}\tilde{G}_{FS}(x, \tilde{\Theta}_G) + G_{WNN}^*(x, W_G^*, C_G^*) \\
 & - \beta^* G_{WNN}^*(x, W_G^*, C_G^*) + \hat{\beta}\hat{G}_{WNN}(x, \widehat{W}_G, \widehat{C}_G) - \hat{G}_{WNN}(x, \widehat{W}_G, \widehat{C}_G) + \varepsilon_G(x, \beta^*)] \mathbf{u} \quad (5.15) \\
 & - \mathbf{u}_r - \mathbf{d}
 \end{aligned}$$

Selon les relations $\alpha^* = \hat{\alpha} + \tilde{\alpha}$ et $\beta^* = \hat{\beta} + \tilde{\beta}$, les expressions $\alpha^* F_{WNN}^*(x, W_F^*, C_F^*)$ et $\beta^* G_{WNN}^*(x, W_G^*, C_G^*)$ dans (5.15) peuvent être écrites comme suit :

$$\begin{aligned}
 \dot{s} = & -[\hat{\alpha}\tilde{F}_{FS}(x, \tilde{\Theta}_F) + \tilde{\alpha}\hat{F}_{FS}(x, \hat{\Theta}_F) + \tilde{\alpha}\tilde{F}_{FS}(x, \tilde{\Theta}_F) + F_{WNN}^*(x, W_F^*, C_F^*) \\
 & - \hat{\alpha}F_{WNN}^*(x, W_F^*, C_F^*) - \tilde{\alpha}F_{WNN}^*(x, W_F^*, C_F^*) + \hat{\alpha}\hat{F}_{WNN}(x, \widehat{W}_F, \widehat{C}_F) \\
 & - \hat{F}_{WNN}(x, \widehat{W}_F, \widehat{C}_F) + \varepsilon_F(x, \alpha^*)] \\
 & - [\hat{\beta}\tilde{G}_{FS}(x, \tilde{\Theta}_G) + \tilde{\beta}\hat{G}_{FS}(x, \hat{\Theta}_G) + \tilde{\beta}\tilde{G}_{FS}(x, \tilde{\Theta}_G) + G_{WNN}^*(x, W_G^*, C_G^*) \\
 & - \hat{\beta}G_{WNN}^*(x, W_G^*, C_G^*) - \tilde{\beta}G_{WNN}^*(x, W_G^*, C_G^*) + \hat{\beta}\hat{G}_{WNN}(x, \widehat{W}_G, \widehat{C}_G) \\
 & - \hat{G}_{WNN}(x, \widehat{W}_G, \widehat{C}_G) + \varepsilon_G(x, \beta^*)] \mathbf{u} - \mathbf{u}_r - \mathbf{d} \quad (5.16)
 \end{aligned}$$

Et en utilisant les deux relations $F_{WNN}^*(x, W_F^*, C_F^*) = \hat{F}_{WNN}(x, \widehat{W}_F, \widehat{C}_F) + \tilde{F}_{WNN}(x, \tilde{W}_F, \tilde{C}_F)$ et $G_{WNN}^*(x, W_G^*, C_G^*) = \hat{G}_{WNN}(x, \widehat{W}_G, \widehat{C}_G) + \tilde{G}_{WNN}(x, \tilde{W}_G, \tilde{C}_G)$, les expressions $\tilde{\alpha}F_{WNN}^*(x, W_F^*, C_F^*)$ et $\tilde{\beta}G_{WNN}^*(x, W_G^*, C_G^*)$ dans (5.16) deviennent comme suit:

$$\begin{aligned}
 \dot{\mathbf{s}} = & - \left[\hat{\alpha} \tilde{\mathbf{F}}_{FS}(\mathbf{x}, \tilde{\Theta}_F) + \tilde{\alpha} \hat{\mathbf{F}}_{FS}(\mathbf{x}, \hat{\Theta}_F) + \tilde{\alpha} \tilde{\mathbf{F}}_{FS}(\mathbf{x}, \tilde{\Theta}_F) + \mathbf{F}_{WNN}^*(\mathbf{x}, \mathbf{W}_F^*, \mathbf{C}_F^*) \right. \\
 & - \hat{\alpha} \mathbf{F}_{WNN}^*(\mathbf{x}, \mathbf{W}_F^*, \mathbf{C}_F^*) - \tilde{\alpha} \hat{\mathbf{F}}_{WNN}(\mathbf{x}, \widehat{\mathbf{W}}_F, \widehat{\mathbf{C}}_F) - \tilde{\alpha} \tilde{\mathbf{F}}_{WNN}(\mathbf{x}, \widetilde{\mathbf{W}}_F, \widetilde{\mathbf{C}}_F) \\
 & + \hat{\alpha} \hat{\mathbf{F}}_{WNN}(\mathbf{x}, \widehat{\mathbf{W}}_F, \widehat{\mathbf{C}}_F) - \hat{\mathbf{F}}_{WNN}(\mathbf{x}, \widehat{\mathbf{W}}_F, \widehat{\mathbf{C}}_F) + \boldsymbol{\varepsilon}_F(\mathbf{x}, \boldsymbol{\alpha}^*) \left. \right] \\
 & - \left[\hat{\beta} \tilde{\mathbf{G}}_{FS}(\mathbf{x}, \tilde{\Theta}_G) + \tilde{\beta} \hat{\mathbf{G}}_{FS}(\mathbf{x}, \hat{\Theta}_G) + \tilde{\beta} \tilde{\mathbf{G}}_{FS}(\mathbf{x}, \tilde{\Theta}_G) + \mathbf{G}_{WNN}^*(\mathbf{x}, \mathbf{W}_G^*, \mathbf{C}_G^*) \right. \\
 & - \hat{\beta} \mathbf{G}_{WNN}^*(\mathbf{x}, \mathbf{W}_G^*, \mathbf{C}_G^*) - \tilde{\beta} \hat{\mathbf{G}}_{WNN}(\mathbf{x}, \widehat{\mathbf{W}}_G, \widehat{\mathbf{C}}_G) - \tilde{\beta} \tilde{\mathbf{G}}_{WNN}(\mathbf{x}, \widetilde{\mathbf{W}}_G, \widetilde{\mathbf{C}}_G) \\
 & + \hat{\beta} \hat{\mathbf{G}}_{WNN}(\mathbf{x}, \widehat{\mathbf{W}}_G, \widehat{\mathbf{C}}_G) - \hat{\mathbf{G}}_{WNN}(\mathbf{x}, \widehat{\mathbf{W}}_G, \widehat{\mathbf{C}}_G) + \boldsymbol{\varepsilon}_G(\mathbf{x}, \boldsymbol{\beta}^*) \left. \right] \mathbf{u} - \mathbf{u}_r - \mathbf{d}
 \end{aligned} \tag{5.17}$$

On prend $\tilde{\alpha}$ et $\tilde{\beta}$ comme facteurs en commun pour pouvoir mettre (5.17) sous la forme suivante :

$$\begin{aligned}
 \dot{\mathbf{s}} = & - \left[\hat{\alpha} \tilde{\mathbf{F}}_{FS}(\mathbf{x}, \tilde{\Theta}_F) + \tilde{\alpha} \left(\hat{\mathbf{F}}_{FS}(\mathbf{x}, \hat{\Theta}_F) - \hat{\mathbf{F}}_{WNN}(\mathbf{x}, \widehat{\mathbf{W}}_F, \widehat{\mathbf{C}}_F) \right) + \tilde{\alpha} \tilde{\mathbf{F}}_{FS}(\mathbf{x}, \tilde{\Theta}_F) \right. \\
 & + \mathbf{F}_{WNN}^*(\mathbf{x}, \mathbf{W}_F^*, \mathbf{C}_F^*) - \hat{\alpha} \mathbf{F}_{WNN}^*(\mathbf{x}, \mathbf{W}_F^*, \mathbf{C}_F^*) - \tilde{\alpha} \tilde{\mathbf{F}}_{WNN}(\mathbf{x}, \widetilde{\mathbf{W}}_F, \widetilde{\mathbf{C}}_F) \\
 & + \hat{\alpha} \hat{\mathbf{F}}_{WNN}(\mathbf{x}, \widehat{\mathbf{W}}_F, \widehat{\mathbf{C}}_F) - \hat{\mathbf{F}}_{WNN}(\mathbf{x}, \widehat{\mathbf{W}}_F, \widehat{\mathbf{C}}_F) + \boldsymbol{\varepsilon}_F(\mathbf{x}, \boldsymbol{\alpha}^*) \left. \right] \\
 & - \left[\hat{\beta} \tilde{\mathbf{G}}_{FS}(\mathbf{x}, \tilde{\Theta}_G) + \tilde{\beta} \left(\hat{\mathbf{G}}_{FS}(\mathbf{x}, \hat{\Theta}_G) - \hat{\mathbf{G}}_{WNN}(\mathbf{x}, \widehat{\mathbf{W}}_G, \widehat{\mathbf{C}}_G) \right) + \tilde{\beta} \tilde{\mathbf{G}}_{FS}(\mathbf{x}, \tilde{\Theta}_G) \right. \\
 & + \mathbf{G}_{WNN}^*(\mathbf{x}, \mathbf{W}_G^*, \mathbf{C}_G^*) - \hat{\beta} \mathbf{G}_{WNN}^*(\mathbf{x}, \mathbf{W}_G^*, \mathbf{C}_G^*) - \tilde{\beta} \tilde{\mathbf{G}}_{WNN}(\mathbf{x}, \widetilde{\mathbf{W}}_G, \widetilde{\mathbf{C}}_G) \\
 & + \hat{\beta} \hat{\mathbf{G}}_{WNN}(\mathbf{x}, \widehat{\mathbf{W}}_G, \widehat{\mathbf{C}}_G) - \hat{\mathbf{G}}_{WNN}(\mathbf{x}, \widehat{\mathbf{W}}_G, \widehat{\mathbf{C}}_G) + \boldsymbol{\varepsilon}_G(\mathbf{x}, \boldsymbol{\beta}^*) \left. \right] \mathbf{u} - \mathbf{u}_r - \mathbf{d}
 \end{aligned} \tag{5.18}$$

On reformule (5.18) sous forme mathématique plus détaillée comme suit :

$$\begin{aligned}
 \dot{s} = & - \left[\hat{\alpha} \tilde{\Theta}_F^T \Phi_F(x) + \tilde{\alpha} \left(\hat{\Theta}_F^T \Phi_F(x) - \widehat{W}_F^T \widehat{\Psi}_F(x, \widehat{C}_F) \right) + \tilde{\alpha} \tilde{\Theta}_F^T \Phi_F(x) + W_F^{*T} \Psi_F^*(x, C_F^*) \right. \\
 & - \hat{\alpha} W_F^{*T} \Psi_F^*(x, C_F^*) - \tilde{\alpha} \widehat{W}_F^T \widehat{\Psi}_F(x, \widehat{C}_F) + \hat{\alpha} \widehat{W}_F^T \widehat{\Psi}_F(x, \widehat{C}_F) - \widehat{W}_F^T \widehat{\Psi}_F(x, \widehat{C}_F) \\
 & \left. + \varepsilon_F(x, \alpha^*) \right] \\
 & - \left[\hat{\beta} \tilde{\Theta}_G^T \Phi_G(x) + \tilde{\beta} \left(\hat{\Theta}_G^T \Phi_G(x) - \widehat{W}_G^T \widehat{\Psi}_G(x, \widehat{C}_G) \right) + \tilde{\beta} \tilde{\Theta}_G^T \Phi_G(x) + W_G^{*T} \Psi_G^*(x, C_G^*) \right. \\
 & - \hat{\beta} W_G^{*T} \Psi_G^*(x, C_G^*) - \tilde{\beta} \widehat{W}_G^T \widehat{\Psi}_G(x, \widehat{C}_G) + \hat{\beta} \widehat{W}_G^T \widehat{\Psi}_G(x, \widehat{C}_G) - \widehat{W}_G^T \widehat{\Psi}_G(x, \widehat{C}_G) \\
 & \left. + \varepsilon_G(x, \beta^*) \right] \mathbf{u} - \mathbf{u}_r - \mathbf{d}
 \end{aligned} \tag{5.19}$$

Admettant que $W_F^{*T} \Psi_F^*(x, C_F^*) = \widehat{W}_F^T \Psi_F^*(x, C_F^*) + \widetilde{W}_F^T \Psi_F^*(x, C_F^*)$ et $W_G^{*T} \Psi_G^*(x, C_G^*) = \widehat{W}_G^T \Psi_G^*(x, C_G^*) + \widetilde{W}_G^T \Psi_G^*(x, C_G^*)$, on peut mettre (5.19) sous la forme suivante :

$$\begin{aligned}
 \dot{s} = & - \left[\hat{\alpha} \tilde{\Theta}_F^T \Phi_F(x) + \tilde{\alpha} \left(\hat{\Theta}_F^T \Phi_F(x) - \widehat{W}_F^T \widehat{\Psi}_F(x, \widehat{C}_F) \right) + \tilde{\alpha} \tilde{\Theta}_F^T \Phi_F(x) + \widehat{W}_F^T \Psi_F^*(x, C_F^*) \right. \\
 & + \widetilde{W}_F^T \Psi_F^*(x, C_F^*) - \hat{\alpha} W_F^{*T} \Psi_F^*(x, C_F^*) - \tilde{\alpha} \widehat{W}_F^T \widehat{\Psi}_F(x, \widehat{C}_F) + \hat{\alpha} \widehat{W}_F^T \widehat{\Psi}_F(x, \widehat{C}_F) \\
 & \left. - \widehat{W}_F^T \widehat{\Psi}_F(x, \widehat{C}_F) + \varepsilon_F(x, \alpha^*) \right] \\
 & - \left[\hat{\beta} \tilde{\Theta}_G^T \Phi_G(x) + \tilde{\beta} \left(\hat{\Theta}_G^T \Phi_G(x) - \widehat{W}_G^T \widehat{\Psi}_G(x, \widehat{C}_G) \right) + \tilde{\beta} \tilde{\Theta}_G^T \Phi_G(x) + \widehat{W}_G^T \Psi_G^*(x, C_G^*) \right. \\
 & + \widetilde{W}_G^T \Psi_G^*(x, C_G^*) - \hat{\beta} W_G^{*T} \Psi_G^*(x, C_G^*) - \tilde{\beta} \widehat{W}_G^T \widehat{\Psi}_G(x, \widehat{C}_G) + \hat{\beta} \widehat{W}_G^T \widehat{\Psi}_G(x, \widehat{C}_G) \\
 & \left. - \widehat{W}_G^T \widehat{\Psi}_G(x, \widehat{C}_G) + \varepsilon_G(x, \beta^*) \right] \mathbf{u} - \mathbf{u}_r - \mathbf{d}
 \end{aligned} \tag{5.20}$$

Développant aussi $\widehat{W}_F^T \Psi_F^*(x, C_F^*) = \widehat{W}_F^T \widehat{\Psi}_F(x, \widehat{C}_F) + \widetilde{W}_F^T \widehat{\Psi}_F(x, \widehat{C}_F)$ et $\widehat{W}_G^T \Psi_G^*(x, C_G^*) = \widehat{W}_G^T \widehat{\Psi}_G(x, \widehat{C}_G) + \widetilde{W}_G^T \widehat{\Psi}_G(x, \widehat{C}_G)$, ce qui donne:

$$\begin{aligned}
 \dot{\mathbf{s}} = & - \left[\hat{\alpha} \tilde{\Theta}_F^T \Phi_F(\mathbf{x}) + \tilde{\alpha} \left(\hat{\Theta}_F^T \Phi_F(\mathbf{x}) - \widehat{\mathbf{W}}_F^T \hat{\Psi}_F(\mathbf{x}, \hat{\mathcal{C}}_F) \right) + \tilde{\alpha} \tilde{\Theta}_F^T \Phi_F(\mathbf{x}) + \widehat{\mathbf{W}}_F^T \hat{\Psi}_F(\mathbf{x}, \hat{\mathcal{C}}_F) \right. \\
 & + \widehat{\mathbf{W}}_F^T \tilde{\Psi}_F(\mathbf{x}, \tilde{\mathcal{C}}_F) + \widehat{\mathbf{W}}_F^T \Psi_F^*(\mathbf{x}, \mathcal{C}_F^*) - \hat{\alpha} \mathbf{W}_F^{*T} \Psi_F^*(\mathbf{x}, \mathcal{C}_F^*) - \tilde{\alpha} \widehat{\mathbf{W}}_F^T \tilde{\Psi}_F(\mathbf{x}, \tilde{\mathcal{C}}_F) \\
 & \left. + \hat{\alpha} \widehat{\mathbf{W}}_F^T \hat{\Psi}_F(\mathbf{x}, \hat{\mathcal{C}}_F) - \widehat{\mathbf{W}}_F^T \hat{\Psi}_F(\mathbf{x}, \hat{\mathcal{C}}_F) + \varepsilon_F(\mathbf{x}, \alpha^*) \right] \\
 & - \left[\hat{\beta} \tilde{\Theta}_G^T \Phi_G(\mathbf{x}) + \tilde{\beta} \left(\hat{\Theta}_G^T \Phi_G(\mathbf{x}) - \widehat{\mathbf{W}}_G^T \hat{\Psi}_G(\mathbf{x}, \hat{\mathcal{C}}_G) \right) + \tilde{\beta} \tilde{\Theta}_G^T \Phi_G(\mathbf{x}) + \widehat{\mathbf{W}}_G^T \hat{\Psi}_G(\mathbf{x}, \hat{\mathcal{C}}_G) \right. \\
 & + \widehat{\mathbf{W}}_G^T \tilde{\Psi}_G(\mathbf{x}, \tilde{\mathcal{C}}_G) + \widehat{\mathbf{W}}_G^T \Psi_G^*(\mathbf{x}, \mathcal{C}_G^*) - \hat{\beta} \mathbf{W}_G^{*T} \Psi_G^*(\mathbf{x}, \mathcal{C}_G^*) - \tilde{\beta} \widehat{\mathbf{W}}_G^T \tilde{\Psi}_G(\mathbf{x}, \tilde{\mathcal{C}}_G) \\
 & \left. + \hat{\beta} \widehat{\mathbf{W}}_G^T \hat{\Psi}_G(\mathbf{x}, \hat{\mathcal{C}}_G) - \widehat{\mathbf{W}}_G^T \hat{\Psi}_G(\mathbf{x}, \hat{\mathcal{C}}_G) + \varepsilon_G(\mathbf{x}, \beta^*) \right] \mathbf{u} - \mathbf{u}_r - \mathbf{d}
 \end{aligned} \tag{5.21}$$

Ce qui permet d'éliminer les termes $\widehat{\mathbf{W}}_F^T \hat{\Psi}_F(\mathbf{x}, \hat{\mathcal{C}}_F)$ et $\widehat{\mathbf{W}}_G^T \hat{\Psi}_G(\mathbf{x}, \hat{\mathcal{C}}_G)$:

$$\begin{aligned}
 \dot{\mathbf{s}} = & - \left[\hat{\alpha} \tilde{\Theta}_F^T \Phi_F(\mathbf{x}) + \tilde{\alpha} \left(\hat{\Theta}_F^T \Phi_F(\mathbf{x}) - \widehat{\mathbf{W}}_F^T \hat{\Psi}_F(\mathbf{x}, \hat{\mathcal{C}}_F) \right) + \tilde{\alpha} \tilde{\Theta}_F^T \Phi_F(\mathbf{x}) + \widehat{\mathbf{W}}_F^T \tilde{\Psi}_F(\mathbf{x}, \tilde{\mathcal{C}}_F) \right. \\
 & + \widehat{\mathbf{W}}_F^T \Psi_F^*(\mathbf{x}, \mathcal{C}_F^*) - \hat{\alpha} \mathbf{W}_F^{*T} \Psi_F^*(\mathbf{x}, \mathcal{C}_F^*) - \tilde{\alpha} \widehat{\mathbf{W}}_F^T \tilde{\Psi}_F(\mathbf{x}, \tilde{\mathcal{C}}_F) + \hat{\alpha} \widehat{\mathbf{W}}_F^T \hat{\Psi}_F(\mathbf{x}, \hat{\mathcal{C}}_F) \\
 & \left. + \varepsilon_F(\mathbf{x}, \alpha^*) \right] \\
 & - \left[\hat{\beta} \tilde{\Theta}_G^T \Phi_G(\mathbf{x}) + \tilde{\beta} \left(\hat{\Theta}_G^T \Phi_G(\mathbf{x}) - \widehat{\mathbf{W}}_G^T \hat{\Psi}_G(\mathbf{x}, \hat{\mathcal{C}}_G) \right) + \tilde{\beta} \tilde{\Theta}_G^T \Phi_G(\mathbf{x}) + \widehat{\mathbf{W}}_G^T \tilde{\Psi}_G(\mathbf{x}, \tilde{\mathcal{C}}_G) \right. \\
 & + \widehat{\mathbf{W}}_G^T \Psi_G^*(\mathbf{x}, \mathcal{C}_G^*) - \hat{\beta} \mathbf{W}_G^{*T} \Psi_G^*(\mathbf{x}, \mathcal{C}_G^*) - \tilde{\beta} \widehat{\mathbf{W}}_G^T \tilde{\Psi}_G(\mathbf{x}, \tilde{\mathcal{C}}_G) + \hat{\beta} \widehat{\mathbf{W}}_G^T \hat{\Psi}_G(\mathbf{x}, \hat{\mathcal{C}}_G) \\
 & \left. + \varepsilon_G(\mathbf{x}, \beta^*) \right] \mathbf{u} - \mathbf{u}_r - \mathbf{d}
 \end{aligned} \tag{5.22}$$

En développant $\hat{\alpha} \mathbf{W}_F^{*T} \Psi_F^*(\mathbf{x}, \mathcal{C}_F^*) = \hat{\alpha} \widehat{\mathbf{W}}_F^T \Psi_F^*(\mathbf{x}, \mathcal{C}_F^*) + \hat{\alpha} \tilde{\mathbf{W}}_F^T \Psi_F^*(\mathbf{x}, \mathcal{C}_F^*)$ et $\hat{\beta} \mathbf{W}_G^{*T} \Psi_G^*(\mathbf{x}, \mathcal{C}_G^*) = \hat{\beta} \widehat{\mathbf{W}}_G^T \Psi_G^*(\mathbf{x}, \mathcal{C}_G^*) + \hat{\beta} \tilde{\mathbf{W}}_G^T \Psi_G^*(\mathbf{x}, \mathcal{C}_G^*)$, on obtient :

$$\begin{aligned}
 \dot{\mathbf{s}} = & - \left[\hat{\alpha} \tilde{\Theta}_F^T \Phi_F(\mathbf{x}) + \tilde{\alpha} \left(\hat{\Theta}_F^T \Phi_F(\mathbf{x}) - \widehat{\mathbf{W}}_F^T \hat{\Psi}_F(\mathbf{x}, \hat{\mathcal{C}}_F) \right) + \tilde{\alpha} \tilde{\Theta}_F^T \Phi_F(\mathbf{x}) + \widehat{\mathbf{W}}_F^T \tilde{\Psi}_F(\mathbf{x}, \tilde{\mathcal{C}}_F) \right. \\
 & + \widehat{\mathbf{W}}_F^T \Psi_F^*(\mathbf{x}, \mathbf{C}_F^*) - \hat{\alpha} \widehat{\mathbf{W}}_F^T \Psi_F^*(\mathbf{x}, \mathbf{C}_F^*) - \hat{\alpha} \tilde{\mathbf{W}}_F^T \Psi_F^*(\mathbf{x}, \mathbf{C}_F^*) - \tilde{\alpha} \tilde{\mathbf{W}}_F^T \tilde{\Psi}_F(\mathbf{x}, \tilde{\mathcal{C}}_F) \\
 & \left. + \hat{\alpha} \widehat{\mathbf{W}}_F^T \hat{\Psi}_F(\mathbf{x}, \hat{\mathcal{C}}_F) + \varepsilon_F(\mathbf{x}, \alpha^*) \right] \\
 & - \left[\hat{\beta} \tilde{\Theta}_G^T \Phi_G(\mathbf{x}) + \tilde{\beta} \left(\hat{\Theta}_G^T \Phi_G(\mathbf{x}) - \widehat{\mathbf{W}}_G^T \hat{\Psi}_G(\mathbf{x}, \hat{\mathcal{C}}_G) \right) + \tilde{\beta} \tilde{\Theta}_G^T \Phi_G(\mathbf{x}) + \widehat{\mathbf{W}}_G^T \tilde{\Psi}_G(\mathbf{x}, \tilde{\mathcal{C}}_G) \right. \\
 & + \widehat{\mathbf{W}}_G^T \Psi_G^*(\mathbf{x}, \mathbf{C}_G^*) - \hat{\beta} \widehat{\mathbf{W}}_G^T \Psi_G^*(\mathbf{x}, \mathbf{C}_G^*) - \hat{\beta} \tilde{\mathbf{W}}_G^T \Psi_G^*(\mathbf{x}, \mathbf{C}_G^*) - \tilde{\beta} \tilde{\mathbf{W}}_G^T \tilde{\Psi}_G(\mathbf{x}, \tilde{\mathcal{C}}_G) \\
 & \left. + \hat{\beta} \widehat{\mathbf{W}}_G^T \hat{\Psi}_G(\mathbf{x}, \hat{\mathcal{C}}_G) + \varepsilon_G(\mathbf{x}, \beta^*) \right] \mathbf{u} - \mathbf{u}_r - \mathbf{d}
 \end{aligned} \tag{5.23}$$

De même, on prend $\widehat{\mathbf{W}}_F^T \Psi_F^*(\mathbf{x}, \mathbf{C}_F^*)$ et $\widehat{\mathbf{W}}_G^T \Psi_G^*(\mathbf{x}, \mathbf{C}_G^*)$ comme des facteurs en commun :

$$\begin{aligned}
 \dot{\mathbf{s}} = & - \left[\hat{\alpha} \tilde{\Theta}_F^T \Phi_F(\mathbf{x}) + \tilde{\alpha} \left(\hat{\Theta}_F^T \Phi_F(\mathbf{x}) - \widehat{\mathbf{W}}_F^T \hat{\Psi}_F(\mathbf{x}, \hat{\mathcal{C}}_F) \right) + \tilde{\alpha} \tilde{\Theta}_F^T \Phi_F(\mathbf{x}) + \widehat{\mathbf{W}}_F^T \tilde{\Psi}_F(\mathbf{x}, \tilde{\mathcal{C}}_F) \right. \\
 & + (I - \hat{\alpha}) \widehat{\mathbf{W}}_F^T \Psi_F^*(\mathbf{x}, \mathbf{C}_F^*) - \hat{\alpha} \widehat{\mathbf{W}}_F^T \Psi_F^*(\mathbf{x}, \mathbf{C}_F^*) - \tilde{\alpha} \tilde{\mathbf{W}}_F^T \tilde{\Psi}_F(\mathbf{x}, \tilde{\mathcal{C}}_F) + \hat{\alpha} \widehat{\mathbf{W}}_F^T \hat{\Psi}_F(\mathbf{x}, \hat{\mathcal{C}}_F) \\
 & \left. + \varepsilon_F(\mathbf{x}, \alpha^*) \right] \\
 & - \left[\hat{\beta} \tilde{\Theta}_G^T \Phi_G(\mathbf{x}) + \tilde{\beta} \left(\hat{\Theta}_G^T \Phi_G(\mathbf{x}) - \widehat{\mathbf{W}}_G^T \hat{\Psi}_G(\mathbf{x}, \hat{\mathcal{C}}_G) \right) + \tilde{\beta} \tilde{\Theta}_G^T \Phi_G(\mathbf{x}) + \widehat{\mathbf{W}}_G^T \tilde{\Psi}_G(\mathbf{x}, \tilde{\mathcal{C}}_G) \right. \\
 & + (I - \hat{\beta}) \widehat{\mathbf{W}}_G^T \Psi_G^*(\mathbf{x}, \mathbf{C}_G^*) - \hat{\beta} \widehat{\mathbf{W}}_G^T \Psi_G^*(\mathbf{x}, \mathbf{C}_G^*) - \tilde{\beta} \tilde{\mathbf{W}}_G^T \tilde{\Psi}_G(\mathbf{x}, \tilde{\mathcal{C}}_G) + \hat{\beta} \widehat{\mathbf{W}}_G^T \hat{\Psi}_G(\mathbf{x}, \hat{\mathcal{C}}_G) \\
 & \left. + \varepsilon_G(\mathbf{x}, \beta^*) \right] \mathbf{u} - \mathbf{u}_r - \mathbf{d}
 \end{aligned} \tag{5.24}$$

Sachant que $-\tilde{\Psi}_F(\mathbf{x}, \tilde{\mathcal{C}}_F) = -\Psi_F^*(\mathbf{x}, \mathbf{C}_F^*) + \hat{\Psi}_F(\mathbf{x}, \hat{\mathcal{C}}_F)$ et $-\tilde{\Psi}_G(\mathbf{x}, \tilde{\mathcal{C}}_G) = -\Psi_G^*(\mathbf{x}, \mathbf{C}_G^*) + \hat{\Psi}_G(\mathbf{x}, \hat{\mathcal{C}}_G)$, on peut simplifier l'expression (5.24) comme suit :

$$\begin{aligned}
 \dot{\mathbf{s}} = & - \left[\hat{\alpha} \tilde{\Theta}_F^T \Phi_F(\mathbf{x}) + \tilde{\alpha} \left(\hat{\Theta}_F^T \Phi_F(\mathbf{x}) - \widehat{\mathbf{W}}_F^T \widehat{\Psi}_F(\mathbf{x}, \widehat{\mathbf{C}}_F) \right) + \tilde{\alpha} \tilde{\Theta}_F^T \Phi_F(\mathbf{x}) + \widehat{\mathbf{W}}_F^T \tilde{\Psi}_F(\mathbf{x}, \tilde{\mathbf{C}}_F) \right. \\
 & + (I - \hat{\alpha}) \widehat{\mathbf{W}}_F^T \Psi_F^*(\mathbf{x}, \mathbf{C}_F^*) - \hat{\alpha} \widehat{\mathbf{W}}_F^T \tilde{\Psi}_F(\mathbf{x}, \tilde{\mathbf{C}}_F) - \tilde{\alpha} \widehat{\mathbf{W}}_F^T \tilde{\Psi}_F(\mathbf{x}, \tilde{\mathbf{C}}_F) + \boldsymbol{\varepsilon}_F(\mathbf{x}, \boldsymbol{\alpha}^*) \left. \right] \\
 & - \left[\hat{\beta} \tilde{\Theta}_G^T \Phi_G(\mathbf{x}) + \tilde{\beta} \left(\hat{\Theta}_G^T \Phi_G(\mathbf{x}) - \widehat{\mathbf{W}}_G^T \widehat{\Psi}_G(\mathbf{x}, \widehat{\mathbf{C}}_G) \right) + \tilde{\beta} \tilde{\Theta}_G^T \Phi_G(\mathbf{x}) + \widehat{\mathbf{W}}_G^T \tilde{\Psi}_G(\mathbf{x}, \tilde{\mathbf{C}}_G) \right. \\
 & + (I - \hat{\beta}) \widehat{\mathbf{W}}_G^T \Psi_G^*(\mathbf{x}, \mathbf{C}_G^*) - \hat{\beta} \widehat{\mathbf{W}}_G^T \tilde{\Psi}_G(\mathbf{x}, \tilde{\mathbf{C}}_G) - \tilde{\beta} \widehat{\mathbf{W}}_G^T \tilde{\Psi}_G(\mathbf{x}, \tilde{\mathbf{C}}_G) + \boldsymbol{\varepsilon}_G(\mathbf{x}, \boldsymbol{\beta}^*) \left. \right] \mathbf{u} - \mathbf{u}_r \\
 & - \mathbf{d}
 \end{aligned} \tag{5.25}$$

Et si on prend $(I - \hat{\alpha})$ et $(I - \hat{\beta})$ comme des facteurs en commun, on obtient :

$$\begin{aligned}
 \dot{\mathbf{s}} = & - \left[\hat{\alpha} \tilde{\Theta}_F^T \Phi_F(\mathbf{x}) + \tilde{\alpha} \left(\hat{\Theta}_F^T \Phi_F(\mathbf{x}) - \widehat{\mathbf{W}}_F^T \widehat{\Psi}_F(\mathbf{x}, \widehat{\mathbf{C}}_F) \right) + \tilde{\alpha} \tilde{\Theta}_F^T \Phi_F(\mathbf{x}) \right. \\
 & + (I - \hat{\alpha}) \widehat{\mathbf{W}}_F^T \tilde{\Psi}_F(\mathbf{x}, \tilde{\mathbf{C}}_F) + (I - \hat{\alpha}) \widehat{\mathbf{W}}_F^T \Psi_F^*(\mathbf{x}, \mathbf{C}_F^*) - \tilde{\alpha} \widehat{\mathbf{W}}_F^T \tilde{\Psi}_F(\mathbf{x}, \tilde{\mathbf{C}}_F) + \boldsymbol{\varepsilon}_F(\mathbf{x}, \boldsymbol{\alpha}^*) \left. \right] \\
 & - \left[\hat{\beta} \tilde{\Theta}_G^T \Phi_G(\mathbf{x}) + \tilde{\beta} \left(\hat{\Theta}_G^T \Phi_G(\mathbf{x}) - \widehat{\mathbf{W}}_G^T \widehat{\Psi}_G(\mathbf{x}, \widehat{\mathbf{C}}_G) \right) + \tilde{\beta} \tilde{\Theta}_G^T \Phi_G(\mathbf{x}) \right. \\
 & + (I - \hat{\beta}) \widehat{\mathbf{W}}_G^T \tilde{\Psi}_G(\mathbf{x}, \tilde{\mathbf{C}}_G) + (I - \hat{\beta}) \widehat{\mathbf{W}}_G^T \Psi_G^*(\mathbf{x}, \mathbf{C}_G^*) - \tilde{\beta} \widehat{\mathbf{W}}_G^T \tilde{\Psi}_G(\mathbf{x}, \tilde{\mathbf{C}}_G) + \boldsymbol{\varepsilon}_G(\mathbf{x}, \boldsymbol{\beta}^*) \left. \right] \mathbf{u} \\
 & - \mathbf{u}_r - \mathbf{d}
 \end{aligned} \tag{5.26}$$

Rappelons les relations $\Psi_F^*(\mathbf{x}, \mathbf{C}_F^*) = \widehat{\Psi}_F(\mathbf{x}, \widehat{\mathbf{C}}_F) + \tilde{\Psi}_F(\mathbf{x}, \tilde{\mathbf{C}}_F)$ et $\Psi_G^*(\mathbf{x}, \mathbf{C}_G^*) = \widehat{\Psi}_G(\mathbf{x}, \widehat{\mathbf{C}}_G) + \tilde{\Psi}_G(\mathbf{x}, \tilde{\mathbf{C}}_G)$, alors (5.26) peut se mettre sous la forme suivante :

$$\begin{aligned}
 \dot{\mathbf{s}} = & - \left[\hat{\alpha} \tilde{\Theta}_F^T \Phi_F(\mathbf{x}) + \tilde{\alpha} \left(\hat{\Theta}_F^T \Phi_F(\mathbf{x}) - \widehat{\mathbf{W}}_F^T \hat{\Psi}_F(\mathbf{x}, \hat{\mathcal{C}}_F) \right) + \tilde{\alpha} \tilde{\Theta}_F^T \Phi_F(\mathbf{x}) \right. \\
 & + (I - \hat{\alpha}) \widehat{\mathbf{W}}_F^T \tilde{\Psi}_F(\mathbf{x}, \tilde{\mathcal{C}}_F) + (I - \hat{\alpha}) \widehat{\mathbf{W}}_F^T \left(\hat{\Psi}_F(\mathbf{x}, \hat{\mathcal{C}}_F) + \tilde{\Psi}_F(\mathbf{x}, \tilde{\mathcal{C}}_F) \right) \\
 & \left. - \tilde{\alpha} \widehat{\mathbf{W}}_F^T \tilde{\Psi}_F(\mathbf{x}, \tilde{\mathcal{C}}_F) + \varepsilon_F(\mathbf{x}, \alpha^*) \right] \\
 & - \left[\hat{\beta} \tilde{\Theta}_G^T \Phi_G(\mathbf{x}) + \tilde{\beta} \left(\hat{\Theta}_G^T \Phi_G(\mathbf{x}) - \widehat{\mathbf{W}}_G^T \hat{\Psi}_G(\mathbf{x}, \hat{\mathcal{C}}_G) \right) + \tilde{\beta} \tilde{\Theta}_G^T \Phi_G(\mathbf{x}) \right. \\
 & + (I - \hat{\beta}) \widehat{\mathbf{W}}_G^T \tilde{\Psi}_G(\mathbf{x}, \tilde{\mathcal{C}}_G) + (I - \hat{\beta}) \widehat{\mathbf{W}}_G^T \left(\hat{\Psi}_G(\mathbf{x}, \hat{\mathcal{C}}_G) + \tilde{\Psi}_G(\mathbf{x}, \tilde{\mathcal{C}}_G) \right) \\
 & \left. - \tilde{\beta} \widehat{\mathbf{W}}_G^T \tilde{\Psi}_G(\mathbf{x}, \tilde{\mathcal{C}}_G) + \varepsilon_G(\mathbf{x}, \beta^*) \right] \mathbf{u} - \mathbf{u}_r - \mathbf{d}
 \end{aligned} \tag{5.27}$$

On réorganise les termes de (5.27) de la manière suivante :

$$\begin{aligned}
 \dot{\mathbf{s}} = & - \left[\hat{\alpha} \tilde{\Theta}_F^T \Phi_F(\mathbf{x}) + \tilde{\alpha} \left(\hat{\Theta}_F^T \Phi_F(\mathbf{x}) - \widehat{\mathbf{W}}_F^T \hat{\Psi}_F(\mathbf{x}, \hat{\mathcal{C}}_F) \right) + (I - \hat{\alpha}) \widehat{\mathbf{W}}_F^T \tilde{\Psi}_F(\mathbf{x}, \tilde{\mathcal{C}}_F) \right. \\
 & + (I - \hat{\alpha}) \widehat{\mathbf{W}}_F^T \left(\hat{\Psi}_F(\mathbf{x}, \hat{\mathcal{C}}_F) + \tilde{\Psi}_F(\mathbf{x}, \tilde{\mathcal{C}}_F) \right) + \tilde{\alpha} \tilde{\Theta}_F^T \Phi_F(\mathbf{x}) - \tilde{\alpha} \widehat{\mathbf{W}}_F^T \tilde{\Psi}_F(\mathbf{x}, \tilde{\mathcal{C}}_F) \\
 & \left. + \varepsilon_F(\mathbf{x}, \alpha^*) \right] \\
 & - \left[\hat{\beta} \tilde{\Theta}_G^T \Phi_G(\mathbf{x}) + \tilde{\beta} \left(\hat{\Theta}_G^T \Phi_G(\mathbf{x}) - \widehat{\mathbf{W}}_G^T \hat{\Psi}_G(\mathbf{x}, \hat{\mathcal{C}}_G) \right) + (I - \hat{\beta}) \widehat{\mathbf{W}}_G^T \tilde{\Psi}_G(\mathbf{x}, \tilde{\mathcal{C}}_G) \right. \\
 & + (I - \hat{\beta}) \widehat{\mathbf{W}}_G^T \left(\hat{\Psi}_G(\mathbf{x}, \hat{\mathcal{C}}_G) + \tilde{\Psi}_G(\mathbf{x}, \tilde{\mathcal{C}}_G) \right) + \tilde{\beta} \tilde{\Theta}_G^T \Phi_G(\mathbf{x}) - \tilde{\beta} \widehat{\mathbf{W}}_G^T \tilde{\Psi}_G(\mathbf{x}, \tilde{\mathcal{C}}_G) \\
 & \left. + \varepsilon_G(\mathbf{x}, \beta^*) \right] \mathbf{u} - \mathbf{u}_r - \mathbf{d}
 \end{aligned} \tag{5.28}$$

En utilisant la linéarisation de Taylor [77] (voir annexe A), on trouve que:

$$\tilde{\Psi}_F(\mathbf{x}, \tilde{\mathcal{C}}_F) = \mathbf{\Omega}_F^T \tilde{\mathcal{C}}_F + O_F = \mathbf{\Omega}_F^T (\mathbf{C}_F^* - \hat{\mathcal{C}}_F) + O_F \tag{5.29}$$

$$\tilde{\Psi}_G(\mathbf{x}, \tilde{\mathcal{C}}_G) = \mathbf{\Omega}_G^T \tilde{\mathcal{C}}_G + O_G = \mathbf{\Omega}_G^T (\mathbf{C}_G^* - \hat{\mathcal{C}}_G) + O_G \tag{5.30}$$

Et si on applique (5.29) et (5.30) dans (5.28), on obtient :

$$\begin{aligned}
 \dot{\mathbf{s}} = & - \left[\hat{\alpha} \tilde{\Theta}_F^T \Phi_F(\mathbf{x}) + \tilde{\alpha} \left(\hat{\Theta}_F^T \Phi_F(\mathbf{x}) - \widehat{\mathbf{W}}_F^T \widehat{\Psi}_F(\mathbf{x}, \widehat{\mathbf{C}}_F) \right) + (I - \hat{\alpha}) \widehat{\mathbf{W}}_F^T (\Omega_F^T \tilde{\mathbf{C}}_F + O_F) \right. \\
 & + (I - \hat{\alpha}) \widehat{\mathbf{W}}_F^T (\widehat{\Psi}_F(\mathbf{x}, \widehat{\mathbf{C}}_F) + \Omega_F^T (\mathbf{C}_F^* - \widehat{\mathbf{C}}_F) + O_F) + \tilde{\alpha} \tilde{\Theta}_F^T \Phi_F(\mathbf{x}) - \tilde{\alpha} \tilde{\mathbf{W}}_F^T \tilde{\Psi}_F(\mathbf{x}, \tilde{\mathbf{C}}_F) \\
 & \left. + \varepsilon_F(\mathbf{x}, \alpha^*) \right] \\
 & - \left[\hat{\beta} \tilde{\Theta}_G^T \Phi_G(\mathbf{x}) + \tilde{\beta} \left(\hat{\Theta}_G^T \Phi_G(\mathbf{x}) - \widehat{\mathbf{W}}_G^T \widehat{\Psi}_G(\mathbf{x}, \widehat{\mathbf{C}}_G) \right) + (I - \hat{\beta}) \widehat{\mathbf{W}}_G^T (\Omega_G^T \tilde{\mathbf{C}}_G + O_G) \right. \\
 & + (I - \hat{\beta}) \widehat{\mathbf{W}}_G^T (\widehat{\Psi}_G(\mathbf{x}, \widehat{\mathbf{C}}_G) + \Omega_G^T (\mathbf{C}_G^* - \widehat{\mathbf{C}}_G) + O_G) + \tilde{\beta} \tilde{\Theta}_G^T \Phi_G(\mathbf{x}) - \tilde{\beta} \tilde{\mathbf{W}}_G^T \tilde{\Psi}_G(\mathbf{x}, \tilde{\mathbf{C}}_G) \\
 & \left. + \varepsilon_G(\mathbf{x}, \beta^*) \right] \mathbf{u} - \mathbf{u}_r - \mathbf{d}
 \end{aligned} \tag{5.31}$$

Mettant (5.31) sous la forme suivante:

$$\begin{aligned}
 \dot{\mathbf{s}} = & - \left[\hat{\alpha} \tilde{\Theta}_F^T \Phi_F(\mathbf{x}) + \tilde{\alpha} \left(\hat{\Theta}_F^T \Phi_F(\mathbf{x}) - \widehat{\mathbf{W}}_F^T \widehat{\Psi}_F(\mathbf{x}, \widehat{\mathbf{C}}_F) \right) + (I - \hat{\alpha}) \widehat{\mathbf{W}}_F^T \Omega_F^T \tilde{\mathbf{C}}_F \right. \\
 & + (I - \hat{\alpha}) \widehat{\mathbf{W}}_F^T O_F + (I - \hat{\alpha}) \widehat{\mathbf{W}}_F^T (\widehat{\Psi}_F(\mathbf{x}, \widehat{\mathbf{C}}_F) - \Omega_F^T \widehat{\mathbf{C}}_F) + (I - \hat{\alpha}) \widehat{\mathbf{W}}_F^T (\Omega_F^T \mathbf{C}_F^* + O_F) \\
 & \left. + \tilde{\alpha} \tilde{\Theta}_F^T \Phi_F(\mathbf{x}) - \tilde{\alpha} \tilde{\mathbf{W}}_F^T \tilde{\Psi}_F(\mathbf{x}, \tilde{\mathbf{C}}_F) + \varepsilon_F(\mathbf{x}, \alpha^*) \right] \\
 & - \left[\hat{\beta} \tilde{\Theta}_G^T \Phi_G(\mathbf{x}) + \tilde{\beta} \left(\hat{\Theta}_G^T \Phi_G(\mathbf{x}) - \widehat{\mathbf{W}}_G^T \widehat{\Psi}_G(\mathbf{x}, \widehat{\mathbf{C}}_G) \right) + (I - \hat{\beta}) \widehat{\mathbf{W}}_G^T \Omega_G^T \tilde{\mathbf{C}}_G \right. \\
 & + (I - \hat{\beta}) \widehat{\mathbf{W}}_G^T O_G + (I - \hat{\beta}) \widehat{\mathbf{W}}_G^T (\widehat{\Psi}_G(\mathbf{x}, \widehat{\mathbf{C}}_G) - \Omega_G^T \widehat{\mathbf{C}}_G) + (I - \hat{\beta}) \widehat{\mathbf{W}}_G^T (\Omega_G^T \mathbf{C}_G^* + O_G) \\
 & \left. + \tilde{\beta} \tilde{\Theta}_G^T \Phi_G(\mathbf{x}) - \tilde{\beta} \tilde{\mathbf{W}}_G^T \tilde{\Psi}_G(\mathbf{x}, \tilde{\mathbf{C}}_G) + \varepsilon_G(\mathbf{x}, \beta^*) \right] \mathbf{u} - \mathbf{u}_r - \mathbf{d}
 \end{aligned} \tag{5.32}$$

On regroupe les termes associés à O_F et O_G de la forme suivante:

$$\begin{aligned}
 \dot{s} = & - \left[\hat{\alpha} \tilde{\Theta}_F^T \Phi_F(x) + \tilde{\alpha} \left(\hat{\Theta}_F^T \Phi_F(x) - \bar{W}_F^T \hat{\Psi}_F(x, \hat{C}_F) \right) + (I - \hat{\alpha}) \bar{W}_F^T \Omega_F^T \tilde{C}_F \right. \\
 & + (I - \hat{\alpha}) W_F^{*T} O_F + (I - \hat{\alpha}) \bar{W}_F^T (\hat{\Psi}_F(x, \hat{C}_F) - \Omega_F^T \hat{C}_F) + (I - \hat{\alpha}) \bar{W}_F^T \Omega_F^T C_F^* \\
 & \left. + \tilde{\alpha} \tilde{\Theta}_F^T \Phi_F(x) - \tilde{\alpha} \bar{W}_F^T \tilde{\Psi}_F(x, \tilde{C}_F) + \varepsilon_F(x, \alpha^*) \right] \\
 & - \left[\hat{\beta} \tilde{\Theta}_G^T \Phi_G(x) + \tilde{\beta} \left(\hat{\Theta}_G^T \Phi_G(x) - \bar{W}_G^T \hat{\Psi}_G(x, \hat{C}_G) \right) + (I - \hat{\beta}) \bar{W}_G^T \Omega_G^T \tilde{C}_G \right. \\
 & + (I - \hat{\beta}) W_G^{*T} O_G + (I - \hat{\beta}) \bar{W}_G^T (\hat{\Psi}_G(x, \hat{C}_G) - \Omega_G^T \hat{C}_G) + (I - \hat{\beta}) \bar{W}_G^T \Omega_G^T C_G^* \\
 & \left. + \tilde{\beta} \tilde{\Theta}_G^T \Phi_G(x) - \tilde{\beta} \bar{W}_G^T \tilde{\Psi}_G(x, \tilde{C}_G) + \varepsilon_G(x, \beta^*) \right] \mathbf{u} - \mathbf{u}_r - \mathbf{d}
 \end{aligned} \tag{5.33}$$

Et en mettant $\Lambda_F = (I - \hat{\alpha}) W_F^{*T} O_F + (I - \hat{\alpha}) \bar{W}_F^T \Omega_F^T C_F^* + \tilde{\alpha} \tilde{\Theta}_F^T \Phi_F(x) - \tilde{\alpha} \bar{W}_F^T \tilde{\Psi}_F(x, \tilde{C}_F)$ et $\Lambda_G = (I - \hat{\beta}) W_G^{*T} O_G + (I - \hat{\beta}) \bar{W}_G^T \Omega_G^T C_G^* + \tilde{\beta} \tilde{\Theta}_G^T \Phi_G(x) - \tilde{\beta} \bar{W}_G^T \tilde{\Psi}_G(x, \tilde{C}_G)$. L'expression (5.33) devient :

$$\begin{aligned}
 \dot{s} = & - \left[\hat{\alpha} \tilde{\Theta}_F^T \Phi_F(x) + \tilde{\alpha} \left(\hat{\Theta}_F^T \Phi_F(x) - \bar{W}_F^T \hat{\Psi}_F(x, \hat{C}_F) \right) + (I - \hat{\alpha}) \bar{W}_F^T \Omega_F^T \tilde{C}_F \right. \\
 & \left. + (I - \hat{\alpha}) \bar{W}_F^T (\hat{\Psi}_F(x, \hat{C}_F) - \Omega_F^T \hat{C}_F) + \Lambda_F + \varepsilon_F(x, \alpha^*) \right] \\
 & - \left[\hat{\beta} \tilde{\Theta}_G^T \Phi_G(x) + \tilde{\beta} \left(\hat{\Theta}_G^T \Phi_G(x) - \bar{W}_G^T \hat{\Psi}_G(x, \hat{C}_G) \right) + (I - \hat{\beta}) \bar{W}_G^T \Omega_G^T \tilde{C}_G \right. \\
 & \left. + (I - \hat{\beta}) \bar{W}_G^T (\hat{\Psi}_G(x, \hat{C}_G) - \Omega_G^T \hat{C}_G) + \Lambda_G + \varepsilon_G(x, \beta^*) \right] \mathbf{u} - \mathbf{u}_r - \mathbf{d}
 \end{aligned} \tag{5.34}$$

On regroupe les incertitudes et les erreurs d'approximation dans un seul terme Δ comme suit:

$$\begin{aligned}
 \dot{s} = & - \left[\hat{\alpha} \tilde{\Theta}_F^T \Phi_F(x) + \tilde{\alpha} \left(\hat{\Theta}_F^T \Phi_F(x) - \widehat{W}_F^T \widehat{\Psi}_F(x, \widehat{C}_F) \right) + (I - \hat{\alpha}) \widehat{W}_F^T \Omega_F^T \tilde{C}_F \right. \\
 & \left. + (I - \hat{\alpha}) \widehat{W}_F^T \left(\widehat{\Psi}_F(x, \widehat{C}_F) - \Omega_F^T \widehat{C}_F \right) \right] \\
 & - \left[\hat{\beta} \tilde{\Theta}_G^T \Phi_G(x) + \tilde{\beta} \left(\hat{\Theta}_G^T \Phi_G(x) - \widehat{W}_G^T \widehat{\Psi}_G(x, \widehat{C}_G) \right) + (I - \hat{\beta}) \widehat{W}_G^T \Omega_G^T \tilde{C}_G \right. \\
 & \left. + (I - \hat{\beta}) \widehat{W}_G^T \left(\widehat{\Psi}_G(x, \widehat{C}_G) - \Omega_G^T \widehat{C}_G \right) \right] \mathbf{u} - \mathbf{u}_r - \mathbf{d} + \Delta_{VFS-WNN}
 \end{aligned} \tag{5.35}$$

Avec $\Delta_{VFS-WNN} = -\Lambda_F - \boldsymbol{\varepsilon}_F(x, \boldsymbol{\alpha}^*) - (\Lambda_G + \boldsymbol{\varepsilon}_G(x, \boldsymbol{\beta}^*))$.

Hypothèse 9:

L'erreur d'approximation est supposée être bornée par $\|\Delta_{VFS-WNN}\| \leq \Gamma_{VFS-WNN}$ [78].

Pour atteindre les objectifs de la commande, il est nécessaire d'ajuster les paramètres de l'approximateur FS-WNN hybride-variable afin d'avoir $\mathbf{e}(t) \rightarrow 0$ quand $t \rightarrow \infty$. Ainsi, les lois d'adaptation des paramètres du FS-WNN hybride-variable sont les suivantes:

$$\dot{\hat{\alpha}}_{FVS-WNN} = -\sigma_{\Theta_F} \hat{\Theta}_{FVS} \hat{\alpha} \tag{5.36}$$

$$\dot{\hat{\beta}}_{GVFS-WNN} = -\sigma_{\Theta_G} \hat{\Theta}_{GFS} \hat{\beta} - 2\tau_{\Theta_G} \hat{\Theta}_{GVFS-WNN} \tag{5.37}$$

$$\dot{\widehat{W}}_{FVFS-WNN} = -\sigma_{W_F} \widehat{W}_{FWNN} (I - \hat{\alpha}) - 2\tau_{W_F} \widehat{W}_{FVFS-WNN} \tag{5.38}$$

$$\dot{\widehat{W}}_{GVFS-WNN} = -\sigma_{W_G} \widehat{W}_{GWNN} (I - \hat{\beta}) - 2\tau_{W_G} \widehat{W}_{GVFS-WNN} \tag{5.39}$$

$$\dot{\widehat{C}}_{FVFS-WNN} = -\sigma_{C_F} \Omega_F \widehat{W}_F (I - \hat{\alpha})^T \mathbf{s} - 2\tau_{C_F} \widehat{C}_{FVFS-WNN} \tag{5.40}$$

$$\dot{\widehat{C}}_{GVFS-WNN} = -\sigma_{C_G} \Omega_G \widehat{W}_G (I - \hat{\beta}) - 2\tau_{C_G} \widehat{C}_{GVFS-WNN} \tag{5.41}$$

$$\dot{\hat{\alpha}} = -\sigma_{\alpha} \left(\widehat{F}_{FS}(x, \widehat{\Theta}_F) - \widehat{F}_{WNN}(x, \widehat{W}_F, \widehat{C}_F) \right) \mathbf{s}^T - 2\tau_{\alpha} \hat{\alpha} \tag{5.42}$$

$$\dot{\hat{\beta}} = -\sigma_{\beta} \left(\widehat{G}_{FS}(x, \widehat{\Theta}_G) - \widehat{G}_{WNN}(x, \widehat{W}_G, \widehat{C}_G) \right) \mathbf{u} - 2\tau_{\beta} \hat{\beta} \tag{5.43}$$

où $\tau_{\Theta_F}, \tau_{\Theta_G}, \tau_{W_F}, \tau_{W_G}, \tau_{C_F}, \tau_{C_G}, \tau_{\alpha}, \tau_{\beta}$ sont des constantes positives.

Théorème 5:

Considérant la boucle fermée de commande de la Figure 5.1, avec le système MIMO non linéaire incertain (2.11). La loi de commande hybride est choisie comme (5.7) et les lois d'adaptation des paramètres sont donné par (5.36)-(5.43). Le contrôleur à base des approximateurs (FS-WNN)s hybrides -variables garantit que le système de commande en boucle fermée est stable, et que tous les signaux sont bornés.

Preuve du théorème 5 :

Considérant une fonction candidate de Lyapunov comme suit :

$$V = \frac{1}{2} \mathbf{s}^T \mathbf{s} + V_{FS} + V_{WNN} + \frac{1}{2\sigma_\alpha} \text{tr}(\tilde{\boldsymbol{\alpha}}^T \tilde{\boldsymbol{\alpha}}) + \frac{1}{2\sigma_\beta} \text{tr}(\tilde{\boldsymbol{\beta}}^T \tilde{\boldsymbol{\beta}}) \quad (5.44)$$

Avec :

$$V_{FS} = V_{F_{FS}} + V_{G_{FS}} \quad (5.45)$$

$$V_{WNN} = V_{F_{WNN}} + V_{G_{WNN}} \quad (5.46)$$

La dérivée de (5.44) par rapport au temps est donnée par :

$$\dot{V} = \mathbf{s}^T \dot{\mathbf{s}} + \dot{V}_{FS} + \dot{V}_{WNN} + \frac{1}{\sigma_\alpha} \text{tr}(\tilde{\boldsymbol{\alpha}}^T \dot{\tilde{\boldsymbol{\alpha}}}) + \frac{1}{\sigma_\beta} \text{tr}(\tilde{\boldsymbol{\beta}}^T \dot{\tilde{\boldsymbol{\beta}}}) \quad (5.47)$$

En substituant (5.35) dans (5.47), on obtient :

$$\begin{aligned} \dot{V} = & -\mathbf{s}^T \left(\left[\hat{\boldsymbol{\alpha}} \tilde{\boldsymbol{\Theta}}_F^T \Phi_F(\mathbf{x}) + \tilde{\boldsymbol{\alpha}} \left(\hat{\boldsymbol{\Theta}}_F^T \Phi_F(\mathbf{x}) - \widehat{\mathbf{W}}_F^T \hat{\Psi}_F(\mathbf{x}, \hat{\mathbf{C}}_F) \right) + (I - \hat{\boldsymbol{\alpha}}) \widehat{\mathbf{W}}_F^T \Omega_F^T \tilde{\mathbf{C}}_F \right. \right. \\ & \left. \left. + (I - \hat{\boldsymbol{\alpha}}) \widehat{\mathbf{W}}_F^T (\hat{\Psi}_F(\mathbf{x}, \hat{\mathbf{C}}_F) - \Omega_F^T \hat{\mathbf{C}}_F) \right] \right. \\ & \left. - \left[\hat{\boldsymbol{\beta}} \tilde{\boldsymbol{\Theta}}_G^T \Phi_G(\mathbf{x}) + \tilde{\boldsymbol{\beta}} \left(\hat{\boldsymbol{\Theta}}_G^T \Phi_G(\mathbf{x}) - \widehat{\mathbf{W}}_G^T \hat{\Psi}_G(\mathbf{x}, \hat{\mathbf{C}}_G) \right) + (I - \hat{\boldsymbol{\beta}}) \widehat{\mathbf{W}}_G^T \Omega_G^T \tilde{\mathbf{C}}_G \right. \right. \\ & \left. \left. + (I - \hat{\boldsymbol{\beta}}) \widehat{\mathbf{W}}_G^T (\hat{\Psi}_G(\mathbf{x}, \hat{\mathbf{C}}_G) - \Omega_G^T \hat{\mathbf{C}}_G) \right] \mathbf{u} - \mathbf{u}_r - \mathbf{d} + \Delta_{V_{FS-WNN}} \right) + \dot{V}_{FS} + \dot{V}_{WNN} \\ & + \frac{1}{\sigma_\alpha} \text{tr}(\tilde{\boldsymbol{\alpha}}^T \dot{\tilde{\boldsymbol{\alpha}}}) + \frac{1}{\sigma_\beta} \text{tr}(\tilde{\boldsymbol{\beta}}^T \dot{\tilde{\boldsymbol{\beta}}}) \end{aligned} \quad (5.48)$$

Les dérivées des fonctions \dot{V}_{FS} et \dot{V}_{WNN} sont données par (2.51) et (3.22) comme suit:

$$\dot{V}_{FS} = \dot{V}_{F_{FS}} + \dot{V}_{G_{FS}}$$

$$\dot{V}_{WNN} = \dot{V}_{F_{WNN}} + \dot{V}_{G_{WNN}}$$

Donc, l'expression (5.48) devient:

$$\begin{aligned}
 \dot{V} = & - \left[\mathbf{s}^T \left(\hat{\alpha} \tilde{\Theta}_F^T \Phi_F(\mathbf{x}) + \tilde{\alpha} \left(\hat{\Theta}_F^T \Phi_F(\mathbf{x}) - \widehat{\mathbf{W}}_F^T \hat{\Psi}_F(\mathbf{x}, \hat{\mathbf{C}}_F) \right) + (I - \hat{\alpha}) \widehat{\mathbf{W}}_F^T \Omega_F^T \tilde{\mathbf{C}}_F \right. \right. \\
 & \left. \left. + (I - \hat{\alpha}) \widehat{\mathbf{W}}_F^T \left(\hat{\Psi}_F(\mathbf{x}, \hat{\mathbf{C}}_F) - \Omega_F^T \hat{\mathbf{C}}_F \right) \right) + \dot{V}_{F_{FS}} + \dot{V}_{F_{WNN}} \right] \\
 & - \left[\mathbf{s}^T \left(\hat{\beta} \tilde{\Theta}_G^T \Phi_G(\mathbf{x}) + \tilde{\beta} \left(\hat{\Theta}_G^T \Phi_G(\mathbf{x}) - \widehat{\mathbf{W}}_G^T \hat{\Psi}_G(\mathbf{x}, \hat{\mathbf{C}}_G) \right) + (I - \hat{\beta}) \widehat{\mathbf{W}}_G^T \Omega_G^T \tilde{\mathbf{C}}_G \right. \right. \\
 & \left. \left. + (I - \hat{\beta}) \widehat{\mathbf{W}}_G^T \left(\hat{\Psi}_G(\mathbf{x}, \hat{\mathbf{C}}_G) - \Omega_G^T \hat{\mathbf{C}}_G \right) \right) \mathbf{u} + \dot{V}_{G_{FS}} + \dot{V}_{G_{WNN}} \right] + \mathbf{s}^T (\mathbf{u}_r - \mathbf{d} + \Delta_{V_{FS-WNN}}) \\
 & + \frac{1}{\sigma_\alpha} \text{tr}(\tilde{\alpha}^T \dot{\tilde{\alpha}}) + \frac{1}{\sigma_\beta} \text{tr}(\tilde{\beta}^T \dot{\tilde{\beta}})
 \end{aligned} \tag{5.49}$$

De ce fait, on peut écrire l'expression (5.49) comme suit :

$$\begin{aligned}
 \dot{V} = & - \left[\left(\mathbf{s}^T \hat{\alpha} \tilde{\Theta}_F^T \Phi_F(\mathbf{x}) + \dot{V}_{F_{FS}} \right) \right. \\
 & \left. + \left(\mathbf{s}^T (I - \hat{\alpha}) \widehat{\mathbf{W}}_F^T \Omega_F^T \tilde{\mathbf{C}}_F + \mathbf{s}^T (I - \hat{\alpha}) \widehat{\mathbf{W}}_F^T \left(\hat{\Psi}_F(\mathbf{x}, \hat{\mathbf{C}}_F) - \Omega_F^T \hat{\mathbf{C}}_F \right) + \dot{V}_{F_{WNN}} \right) \right. \\
 & \left. + \mathbf{s}^T \tilde{\alpha} \left(\hat{\Theta}_F^T \Phi_F(\mathbf{x}) - \widehat{\mathbf{W}}_F^T \hat{\Psi}_F(\mathbf{x}, \hat{\mathbf{C}}_F) \right) + \frac{1}{\sigma_\alpha} \text{tr}(\tilde{\alpha}^T \dot{\tilde{\alpha}}) \right] \\
 & - \left[\left(\mathbf{s}^T \hat{\beta} \tilde{\Theta}_G^T \Phi_G(\mathbf{x}) \mathbf{u} + \dot{V}_{G_{FS}} \right) \right. \\
 & \left. + \left(\mathbf{s}^T (I - \hat{\beta}) \widehat{\mathbf{W}}_G^T \Omega_G^T \tilde{\mathbf{C}}_G \mathbf{u} + \mathbf{s}^T (I - \hat{\beta}) \widehat{\mathbf{W}}_G^T \left(\hat{\Psi}_G(\mathbf{x}, \hat{\mathbf{C}}_G) - \Omega_G^T \hat{\mathbf{C}}_G \right) \mathbf{u} + \dot{V}_{G_{WNN}} \right) \right. \\
 & \left. + \mathbf{s}^T \tilde{\beta} \left(\hat{\Theta}_G^T \Phi_G(\mathbf{x}) - \widehat{\mathbf{W}}_G^T \hat{\Psi}_G(\mathbf{x}, \hat{\mathbf{C}}_G) \right) \mathbf{u} + \frac{1}{\sigma_\beta} \text{tr}(\tilde{\beta}^T \dot{\tilde{\beta}}) \right] + \mathbf{s}^T (\mathbf{u}_r - \mathbf{d} + \Delta_{V_{FS-WNN}})
 \end{aligned} \tag{5.50}$$

Et on rappelle les dérivées par rapport au temps des fonctions candidates de Lyapunov qu'on a établi précédemment dans (2.52) et (3.23) :

$$\begin{aligned}
 \dot{V}_{F_{FS}} &= \frac{1}{\sigma_{\Theta_F}} \text{tr} \left(\tilde{\Theta}_F^T \dot{\tilde{\Theta}}_F \right) \\
 \dot{V}_{G_{FS}} &= \frac{1}{\sigma_{\Theta_G}} \text{tr} \left(\tilde{\Theta}_G^T \dot{\tilde{\Theta}}_G \right) \\
 \dot{V}_{F_{WNN}} &= \frac{1}{\sigma_{W_F}} \text{tr} \left(\widehat{\mathbf{W}}_F^T \dot{\widehat{\mathbf{W}}}_F \right) + \frac{1}{\sigma_{C_F}} \text{tr} \left(\tilde{\mathbf{C}}_F^T \dot{\tilde{\mathbf{C}}}_F \right) \\
 \dot{V}_{G_{WNN}} &= \frac{1}{\sigma_{W_G}} \text{tr} \left(\widehat{\mathbf{W}}_G^T \dot{\widehat{\mathbf{W}}}_G \right) + \frac{1}{\sigma_{C_G}} \text{tr} \left(\tilde{\mathbf{C}}_G^T \dot{\tilde{\mathbf{C}}}_G \right)
 \end{aligned}$$

En substituant ces expressions dans (5.51), on obtient:

$$\begin{aligned}
 \dot{V} = & - \left[\left(\mathbf{s}^T \hat{\boldsymbol{\alpha}} \tilde{\Theta}_F^T \Phi_F(\mathbf{x}) + \frac{1}{\sigma_{\Theta_F}} \text{tr} \left(\tilde{\Theta}_F^T \dot{\tilde{\Theta}}_F \right) \right) \right. \\
 & + \left(\mathbf{s}^T (I - \hat{\boldsymbol{\alpha}}) \tilde{\mathbf{W}}_F^T (\hat{\Psi}_F(\mathbf{x}, \hat{\mathbf{C}}_F) - \boldsymbol{\Omega}_F^T \hat{\mathbf{C}}_F) + \frac{1}{\sigma_{W_F}} \text{tr} \left(\tilde{\mathbf{W}}_F^T \dot{\tilde{\mathbf{W}}}_F \right) \right) \\
 & + \left(\mathbf{s}^T (I - \hat{\boldsymbol{\alpha}}) \tilde{\mathbf{W}}_F^T \boldsymbol{\Omega}_F^T \tilde{\mathbf{C}}_F + \frac{1}{\sigma_{C_F}} \text{tr} \left(\tilde{\mathbf{C}}_F^T \dot{\tilde{\mathbf{C}}}_F \right) \right) \\
 & \left. + \left(\mathbf{s}^T \tilde{\boldsymbol{\alpha}} \left(\hat{\Theta}_F^T \Phi_F(\mathbf{x}) - \tilde{\mathbf{W}}_F^T \hat{\Psi}_F(\mathbf{x}, \hat{\mathbf{C}}_F) \right) + \frac{1}{\sigma_{\alpha}} \text{tr} \left(\tilde{\boldsymbol{\alpha}}^T \dot{\tilde{\boldsymbol{\alpha}}} \right) \right) \right] \\
 & - \left[\left(\mathbf{s}^T \tilde{\boldsymbol{\beta}} \tilde{\Theta}_G^T \Phi_G(\mathbf{x}) \mathbf{u} + \frac{1}{\sigma_{\Theta_G}} \text{tr} \left(\tilde{\Theta}_G^T \dot{\tilde{\Theta}}_G \right) \right) \right. \\
 & + \left(\mathbf{s}^T (I - \tilde{\boldsymbol{\beta}}) \tilde{\mathbf{W}}_G^T (\hat{\Psi}_G(\mathbf{x}, \hat{\mathbf{C}}_G) - \boldsymbol{\Omega}_G^T \hat{\mathbf{C}}_G) \mathbf{u} + \frac{1}{\sigma_{W_G}} \text{tr} \left(\tilde{\mathbf{W}}_G^T \dot{\tilde{\mathbf{W}}}_G \right) \right) \\
 & + \left(\mathbf{s}^T (I - \tilde{\boldsymbol{\beta}}) \tilde{\mathbf{W}}_G^T \boldsymbol{\Omega}_G^T \tilde{\mathbf{C}}_G \mathbf{u} + \frac{1}{\sigma_{C_G}} \text{tr} \left(\tilde{\mathbf{C}}_G^T \dot{\tilde{\mathbf{C}}}_G \right) \right) \\
 & \left. + \left(\mathbf{s}^T \tilde{\boldsymbol{\beta}} \left(\hat{\Theta}_G^T \Phi_G(\mathbf{x}) - \tilde{\mathbf{W}}_G^T \hat{\Psi}_G(\mathbf{x}, \hat{\mathbf{C}}_G) \right) \mathbf{u} + \frac{1}{\sigma_{\beta}} \text{tr} \left(\tilde{\boldsymbol{\beta}}^T \dot{\tilde{\boldsymbol{\beta}}} \right) \right) \right] \\
 & + \mathbf{s}^T (\mathbf{u}_r - \mathbf{d} + \Delta_{VFS-WNN})
 \end{aligned} \tag{5.51}$$

Ce qui donne :

$$\begin{aligned}
 \dot{V} = & \left[\text{tr} \left(\tilde{\Theta}_F^T \left(\dot{\tilde{\Theta}}_F - \sigma_{\Theta_F} \Phi_F(\mathbf{x}) \mathbf{s}^T \hat{\boldsymbol{\alpha}} \right) \right) \right. \\
 & + \text{tr} \left(\tilde{\mathbf{W}}_F^T \left(\dot{\tilde{\mathbf{W}}}_F - \sigma_{W_F} (\hat{\Psi}_F(\mathbf{x}, \hat{\mathbf{C}}_F) - \boldsymbol{\Omega}_F^T \hat{\mathbf{C}}_F) \mathbf{s}^T (I - \hat{\boldsymbol{\alpha}}) \right) \right) \\
 & + \text{tr} \left(\tilde{\mathbf{C}}_F^T \left(\dot{\tilde{\mathbf{C}}}_F - \sigma_{C_F} \boldsymbol{\Omega}_F \tilde{\mathbf{W}}_F (I - \hat{\boldsymbol{\alpha}})^T \mathbf{s} \right) \right) \\
 & \left. + \text{tr} \left(\tilde{\boldsymbol{\alpha}}^T \left(\dot{\tilde{\boldsymbol{\alpha}}} - \sigma_{\alpha} \left(\hat{\Theta}_F^T \Phi_F(\mathbf{x}) - \tilde{\mathbf{W}}_F^T \hat{\Psi}_F(\mathbf{x}, \hat{\mathbf{C}}_F) \right) \mathbf{s} \right) \right) \right] \\
 & + \left[\text{tr} \left(\tilde{\Theta}_G^T \left(\dot{\tilde{\Theta}}_G - \sigma_{\Theta_G} \Phi_G(\mathbf{x}) \mathbf{u} \mathbf{s}^T \tilde{\boldsymbol{\beta}} \right) \right) \right. \\
 & + \text{tr} \left(\tilde{\mathbf{W}}_G^T \left(\dot{\tilde{\mathbf{W}}}_G - \sigma_{W_G} (\hat{\Psi}_G(\mathbf{x}, \hat{\mathbf{C}}_G) - \boldsymbol{\Omega}_G^T \hat{\mathbf{C}}_G) \mathbf{u} \mathbf{s}^T (I - \tilde{\boldsymbol{\beta}}) \right) \right) \\
 & + \text{tr} \left(\tilde{\mathbf{C}}_G^T \left(\dot{\tilde{\mathbf{C}}}_G - \sigma_{C_G} \boldsymbol{\Omega}_G \tilde{\mathbf{W}}_G (I - \tilde{\boldsymbol{\beta}})^T \mathbf{u} \mathbf{s}^T \right) \right) \\
 & \left. + \text{tr} \left(\tilde{\boldsymbol{\beta}}^T \left(\dot{\tilde{\boldsymbol{\beta}}} - \sigma_{\beta} \left(\hat{\Theta}_G^T \Phi_G(\mathbf{x}) - \tilde{\mathbf{W}}_G^T \hat{\Psi}_G(\mathbf{x}, \hat{\mathbf{C}}_G) \right) \mathbf{u} \mathbf{s}^T \right) \right) \right] \\
 & + \mathbf{s}^T (\mathbf{u}_r - \mathbf{d} + \Delta_{VFS-WNN})
 \end{aligned} \tag{5.52}$$

Avec $\dot{\tilde{\Theta}}_F = -\dot{\hat{\Theta}}_F$, $\dot{\tilde{\Theta}}_G = -\dot{\hat{\Theta}}_G$, $\tilde{\mathbf{C}}_F^T = -\dot{\hat{\mathbf{C}}}_F$, $\tilde{\mathbf{C}}_G^T = -\dot{\hat{\mathbf{C}}}_G$, $\dot{\tilde{\mathbf{W}}}_F = -\dot{\hat{\mathbf{W}}}_F$, $\dot{\tilde{\mathbf{W}}}_G = -\dot{\hat{\mathbf{W}}}_G$.

On peut voir (5.52) de la façon suivante :

$$\begin{aligned} \dot{V}_{VFS-WNN} = & \dot{V}_{FS-WNN} \Big|_{\alpha=\hat{\alpha}} + tr \left(\tilde{\alpha}^T \left(\dot{\tilde{\alpha}} - \sigma_{\alpha} \left(\hat{F}_{FS}(x, \hat{\Theta}_F) - \hat{F}_{WNN}(x, \hat{W}_F, \hat{C}_F) \right) s \right) \right) + \\ & tr \left(\tilde{\beta}^T \left(\dot{\tilde{\beta}} - \sigma_{\beta} \left(\hat{G}_{FS}(x, \hat{\Theta}_G) - \hat{G}_{WNN}(x, \hat{W}_G, \hat{C}_G) \right) u s^T \right) \right) + s^T (u_r - d + \Delta_{VFS-WNN}) \end{aligned}$$

Par conséquence, en utilisant (5.36)-(5.43), on obtient:

$$\dot{V} = s^T (u_r - d + \Delta_{VFS-WNN}) \quad (5.53)$$

En utilisant l'hypothèse 09 :

$$\dot{V} \leq s^T (-K_{VFS-WNN} sgn(s) + D + \Gamma_{VFS-WNN}) \quad (5.54)$$

$$\dot{V} \leq |s^T| (-K_{VFS-WNN} + \xi_{VFS-WNN}) \quad (5.55)$$

Étant donné que $K_{VFS-WNN}$ et $\xi_{VFS-WNN} = D + \Gamma_{VFS-WNN}$ sont sélectionnés par le concepteur, $K_{VFS-WNN}$ doit être choisi de telle sorte que $\dot{V} \leq 0$. Par conséquent, le système en boucle fermée est stable. Ainsi, tous les signaux sont bornés ce qui implique que l'erreur de poursuite e et tous ses dérivés sont bornées, également s est bornée, c'est-à-dire, $s \in L_{\infty}$.

L'intégration des deux côtés de (5.55) donne:

$$\int_0^t |s| d\tau \leq \frac{1}{K_{VFS-WNN} + \xi_{VFS-WNN}} (|V(0)| - |V(t)|) \quad (5.56)$$

Ce qui signifie que $s \in L_1$, en se basant sur le Lemme de Barbalat [95], on trouve que:

$\lim_{t \rightarrow \infty} |s(t)| = 0$, donc $\lim_{t \rightarrow \infty} |e(t)| = 0$. Ainsi, l'erreur de poursuite convergera vers zéro quand $t \rightarrow \infty$.

4. Résultats de simulation

On utilise le modèle du robot manipulateur à deux degrés de liberté de la Figure 2.3. Les conditions initiales du robot manipulateur sont choisies comme $(x_1, x_2, x_3, x_4) = (0,0,0,0)$. Dans cette partie, les deux FS-WNNs hybrides qui ont été synthétisés dans le chapitre précédent pour

approximer les fonctions inconnues $\mathbf{F}(\mathbf{x})$ et $\mathbf{G}(\mathbf{x})$ seront exploités de la même manière pour approximer les fonctions inconnues du système, mais cette fois ci, les paramètres de modulation $\boldsymbol{\alpha}$ et $\boldsymbol{\beta}$ sont ajustés en ligne en utilisant les lois d'adaptation (5.42)-(5.43). Les approximations obtenues sont utilisées dans la synthèse de la loi de commande du contrôleur adaptatif indirect donnée par (5.7). Les paramètres de la surface de glissement sont donnés par $\lambda = 20I_2$.

On a vu dans le chapitre précédent que chaque fonction du système est approximée par le FS-WNN hybride en combinant l'approximation du système flou développé dans le chapitre 03 avec l'approximation du WNN développé dans le chapitre 02 à travers la technique de modulation. La technique de modulation utilisée dans le chapitre 04 fournit des paramètres fixes et préalablement choisis par le concepteur. L'optimisation en ligne des paramètres de modulation a pour objectif d'atteindre les performances optimales de l'approximation FS-WNN hybride. Dans ce chapitre, l'approximation de vecteur des fonctions $\mathbf{F}(\mathbf{x})$ générée par le FS est multipliée par un paramètre de modulation adaptatif $\hat{\boldsymbol{\alpha}}$, et l'approximation de la fonction $\mathbf{F}(\mathbf{x})$ générée par le WNN est multipliée par le complément adaptatif de $\hat{\boldsymbol{\alpha}}$. L'adjonction de ces approximations constitue l'approximation hybride-variable de la fonction $\mathbf{F}(\mathbf{x})$ inconnue du système. La même procédure sera utilisée pour approximer la matrice des fonctions $\mathbf{G}(\mathbf{x})$, où les paramètres adaptatifs de modulation ($\hat{\boldsymbol{\beta}}, I-\hat{\boldsymbol{\beta}}$) seront utilisés pour associer les différentes approximations de $\mathbf{G}(\mathbf{x})$. Les valeurs de $\hat{\boldsymbol{\alpha}}$ et $\hat{\boldsymbol{\beta}}$ sont initialisées arbitrairement comme $\hat{\boldsymbol{\alpha}} = 0.5I_2$ et $\hat{\boldsymbol{\beta}} = 0.5I_2$, ce choix donne initialement le même degré de contribution à chaque approximateur. Après, l'algorithme d'adaptation se charge d'ajuster les degrés de contribution selon un critère de minimisation d'une fonction coût au cours du temps jusqu'à ce qu'il trouve les valeurs optimales de $\hat{\boldsymbol{\alpha}}$ et $\hat{\boldsymbol{\beta}}$ qui offrent les meilleures performances de l'approximateur FS-WNN hybride-variable. Les trajectoires désirées sont sélectionnées comme $\mathbf{y}_d^T = [0.1\sin(t) \quad 0.1\sin(t)]$.

Afin de montrer l'intérêt de l'adaptation en ligne des paramètres de modulation de l'approximateur FS-WNN hybride, on reprend l'approximateur FS-WNN hybride développé dans le chapitre 04. On garde la même architecture et on établit les instructions d'adaptation des paramètres de modulation $\hat{\boldsymbol{\alpha}}$ et $\hat{\boldsymbol{\beta}}$. On considère de même les nouvelles lois d'adaptations des autres paramètres de l'approximateur FS-WNN hybride qui sont données par les équations (5.36)-(5.41). Cela permet à l'approximateur FS-WNN hybride d'être intégralement autonome dans la sélection et l'adaptation des paramètres. Ce qui lui donne plus de flexibilité et de robustesse face

aux incertitudes et perturbations. Pour mettre en évidence ces avantages, une analyse des performances du nouveau approximateur FS-WNN hybride-variable est présentée par simulation. Le nouveau approximateur FS-WNN hybride-variable est utilisé pour contrôler le robot manipulateur à deux degrés de liberté (2.63).

Les performances de poursuite dans des conditions nominales pour la première articulation y_1 et la deuxième articulation y_2 du robot manipulateur sont données dans la Figure 5.2 et la Figure 5.3, respectivement. Bien que les Figure 5.4 et 5.5 montrent les signaux de commande pour les deux articulations, et les erreurs de poursuite sont illustrées dans la Figure 5.6 pour la première articulation et la Figure 5.7 pour la deuxième articulation. Enfin, les trajectoires des éléments diagonaux de $\hat{\alpha}$ et $\hat{\beta}$ en fonction du temps sont donnés dans la Figure 5.8 et 5.9, respectivement.

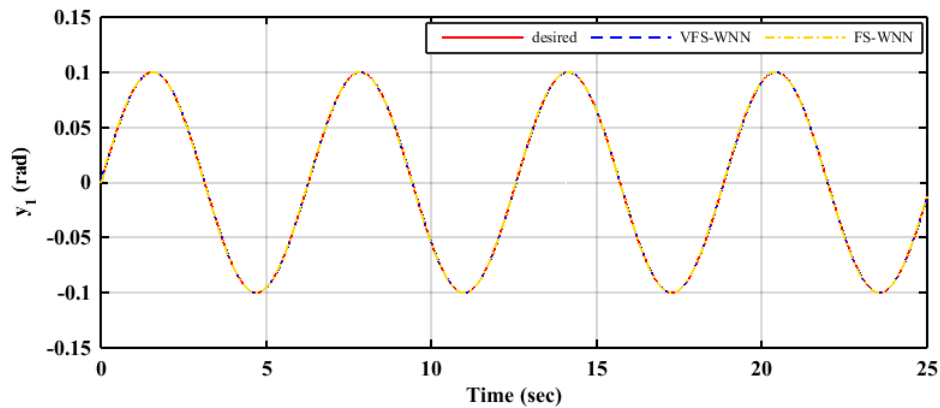


Figure 5.2 : Performance de poursuite de y_1 d'une trajectoire désirée y_{d1} en absence des perturbations et des variations paramétriques.

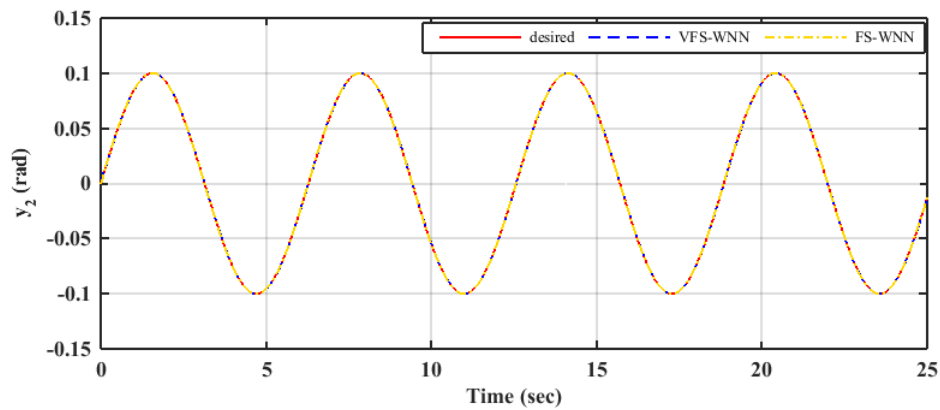


Figure 5.3 : Performance de poursuite de y_2 d'une trajectoire désirée y_{d2} en absence des perturbations et des variations paramétriques.

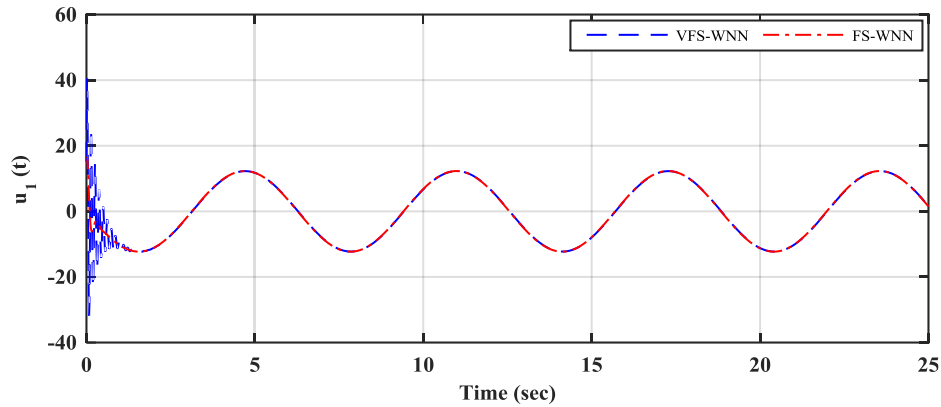


Figure 5.4 : Signal de commande u_1 en absence des perturbations et des variations paramétriques.

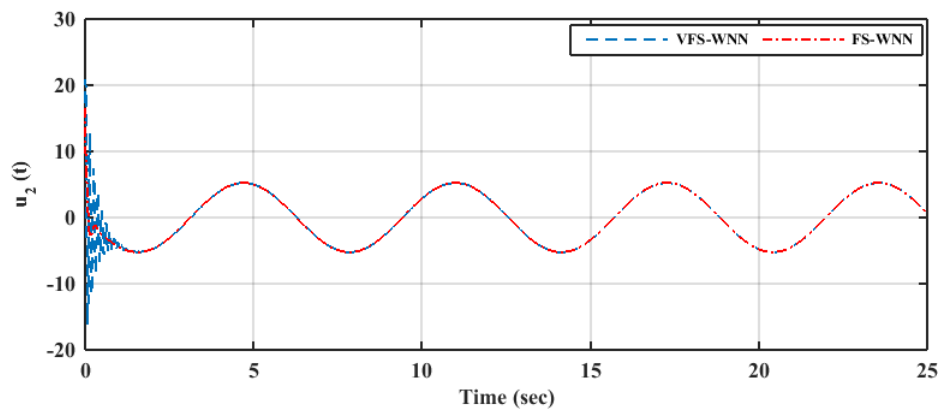


Figure 5.5 : Signal de commande u_2 en absence des perturbations et des variations paramétriques.

La Figure 5.2 et la Figure 5.3 montrent les performances de poursuite des sorties du système y_1 et y_2 pour le contrôleur FS-WNN hybride-variable proposé dans ce chapitre avec des paramètres de modulation adaptatifs. Le contrôleur FS-WNN hybride-variable pourrait assurer des bonnes performances, où les sorties du système y_1 et y_2 suivaient avec précision la trajectoire désirée y_d .

Les signaux de commande du contrôleur FS-WNN hybride-variable en absence des perturbations et des variations paramétriques sont donnés dans la Figure 5.4 pour la première articulation et dans la Figure 5.5 pour la deuxième articulation. Le contrôleur synthétisé avec des paramètres de modulation adaptatifs montre un signal de commande avec de petites oscillations dans la phase transitoire qui disparaissent rapidement dans la phase transitoire, cela est due à l'effet de l'adaptation dans la recherche et l'ajustement des différents paramètres du système FS-WNN hybride-variable.

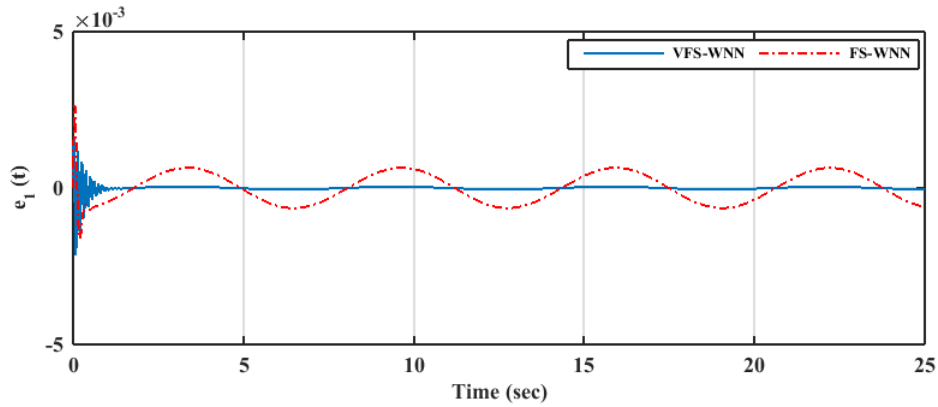


Figure 5.6 : Erreur de poursuite e_1 en absence des perturbations et des variations paramétriques.

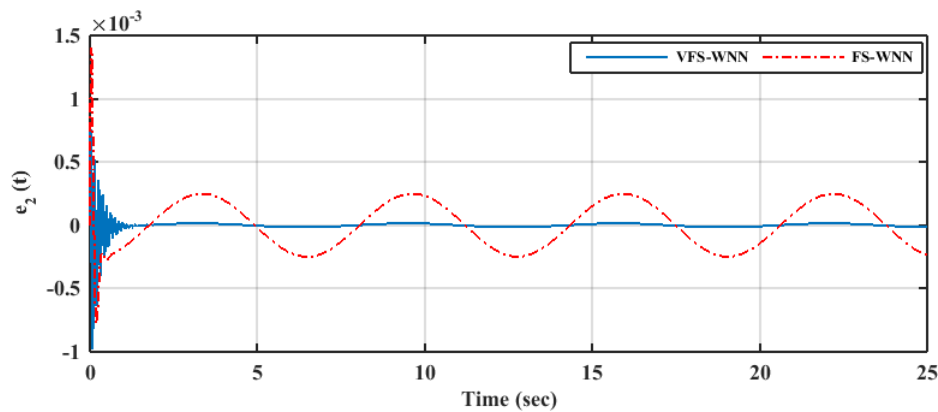


Figure 5.7 : Erreur de poursuite e_2 en absence des perturbations et des variations paramétriques.

La Figure 5.6 et la Figure 5.7 montrent les erreurs de poursuite du contrôleur FS-WNN hybride-variable en l'absence des perturbations externes et des variations paramétriques. D'après la Figure 5.6 et la Figure 5.7, on peut voir que l'erreur de poursuite est très petite et converge rapidement vers 0. Cela est grâce à l'association de deux approximateurs adaptatifs à travers des paramètres de modulation adaptatifs aussi. Dans la Figure 5.8 et 5.9 on voit la trajectoire des courbes des paramètres de modulation en fonction du temps.

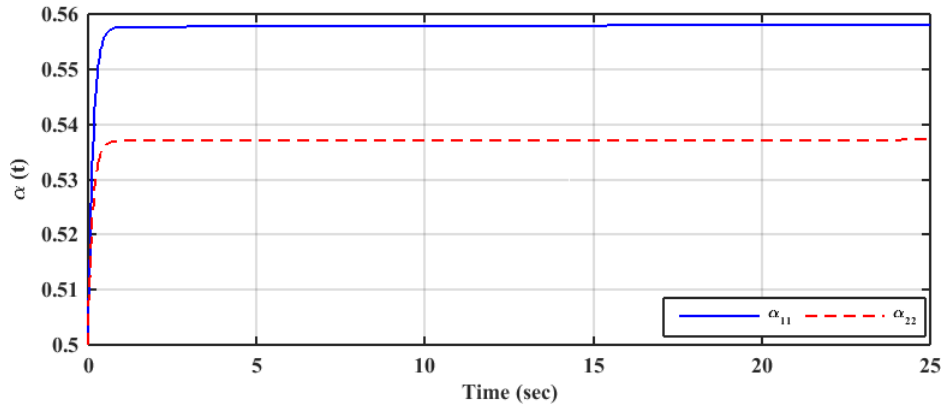


Figure 5.8 : Trajectoire des éléments diagonaux de facteur de modulation $\hat{\alpha}$ en fonction du temps en absence des perturbations et des variations paramétriques.

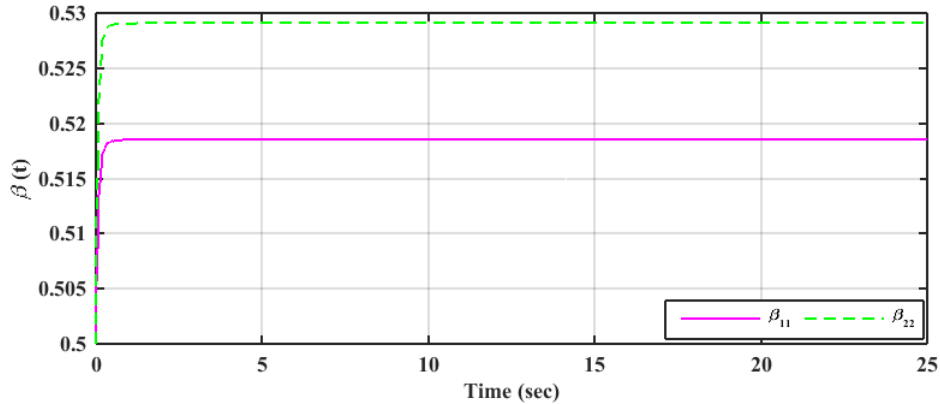


Figure 5.9 : Trajectoire des éléments diagonaux de facteur de modulation $\hat{\beta}$ en fonction du temps en absence des perturbations et des variations paramétriques.

A partir de la Figure 5.8 et de la Figure 5.9 on peut observer l'allure des courbes des paramètres de modulation en fonction du temps. On remarque que l'algorithme d'ajustement des paramètres de modulation a ajusté les paramètres du facteur de modulation $\hat{\alpha}$ qui étaient initialisés comme $\hat{\alpha} = \text{diag}(0.5, 0.5)$ pour atteindre les paramètres optimaux du facteur de modulation $\hat{\alpha}$ dans les alentours de $\hat{\alpha} = \text{diag}(0.54, 0.56)$. De même pour facteur de modulation $\hat{\beta}$ qui était initialisé aussi comme $\hat{\beta} = \text{diag}(0.5, 0.5)$ et qui a été ajusté vers les paramètres optimaux dans les alentours de $\hat{\beta} = \text{diag}(0.52, 0.53)$. On remarque que les valeurs finales de $\hat{\alpha}$ et $\hat{\beta}$ sont proches des valeurs d'initialisation ce qui est parfaitement justifiable par le bon choix des paramètres de $\hat{\alpha}$ et $\hat{\beta}$ qui a été fait par tâtonnement (trial and error) dans le chapitre précédent.

Tableau 5.1 : Comparaison des performances de contrôleur FS-WNN hybride-variable avec le contrôleur FS-WNN dans le cas d'absence des perturbations externes et des variations paramétriques.

	Link 1		Link 2	
	MSE(y_1)	$E(u_1)$	MSE(y_2)	$E(u_2)$
FS-WNN	2.2844×10^{-7}	1.9152×10^5	3.7398×10^{-8}	3.5192×10^4
VFS-WNN	1.6935×10^{-8}	2.0209×10^6	3.3481×10^{-9}	3.7925×10^6

Le Tableau 5.1 compare les performances des performances de contrôleur FS-WNN hybride-variable avec le contrôleur FS-WNN pour la première et la deuxième articulation. Le MSE de l'erreur de poursuite et l'énergie du signal de commande sont fixés comme des critères d'évaluation. L'expression de l'énergie E d'un signal est donnée par [49]:

$$E = \sum_1^N u^2$$

En observant le MSE pour les deux articulations, nous remarquons que les deux contrôleurs FS-WNN hybride et le FS-WNN hybride-variable ont présenté une bonne poursuite avec une erreur de poursuite de l'ordre de 10^{-8} pour la première articulation et de l'ordre 10^{-9} pour la deuxième articulation. Cependant, le contrôleur basé sur le FS-WNN -variable a donné un meilleur MSE grâce à l'ajustement en ligne des paramètres de modulation $\hat{\alpha}$ et $\hat{\beta}$. Les paramètres ajustés n'étaient pas loin de ceux qu'on a choisi par tâtonnement pour le FS-WNN ce qui explique la différence légère dans la précision d'approximation entre les deux approximateurs FS-WNN et le FS-WNN -variable.

Pour évaluer la performance du FS-WNN hybride-variable dans le cas de présence des perturbations et des variations paramétriques par en comparaison au FS-WNN hybride, les deux contrôleurs subissent un test des effets de frottement ajoutés à $t=15$ sec, en plus d'un changement des paramètres du système ajouté à $t=35$ sec. La position de la première articulation y_1 et de la deuxième articulation y_2 sont données aux Figure 5.10 et Figure 5.11, respectivement. Les Figures 5.12 et 5.13 montrent les signaux de commande pour les deux articulations.

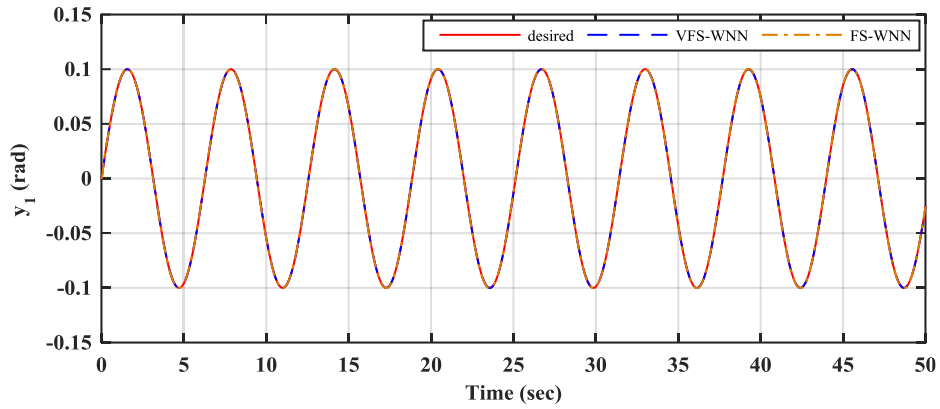


Figure 5.10 : Performance de poursuite de y_1 d'une trajectoire désirée y_{d1} en présence des perturbations et des variations paramétriques

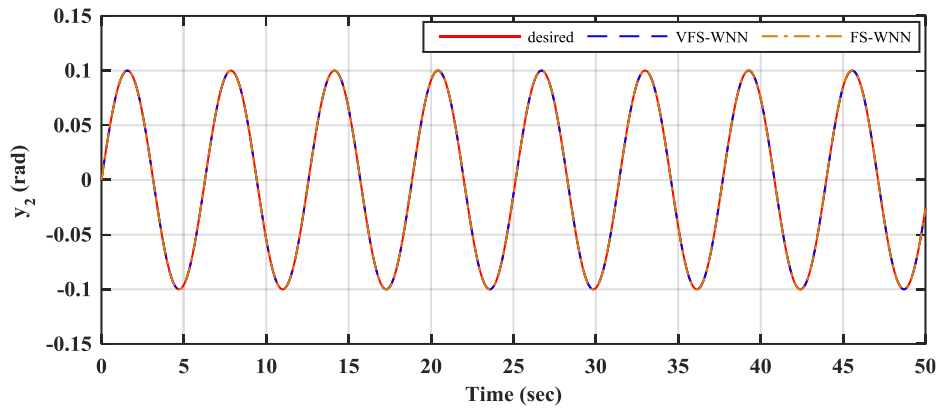


Figure 5.11 : Performance de poursuite de y_2 d'une trajectoire désirée y_{d2} en absence des perturbations et des variations paramétriques

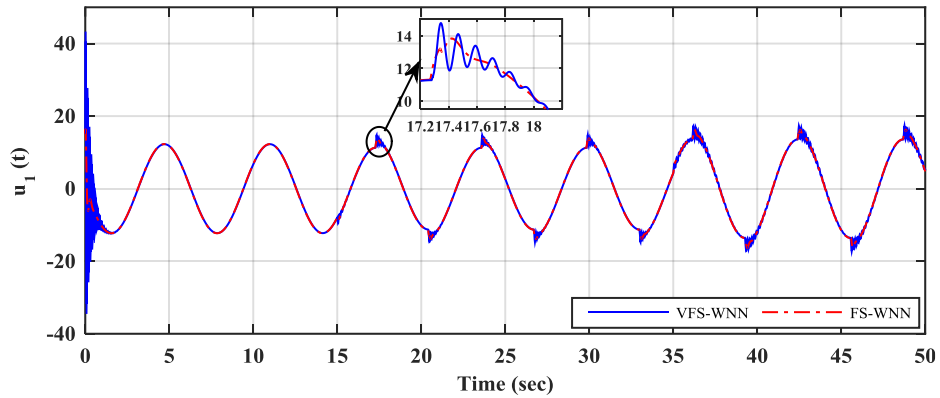


Figure 5.12 : Signal de commande u_1 en présence des perturbations et des variations paramétriques

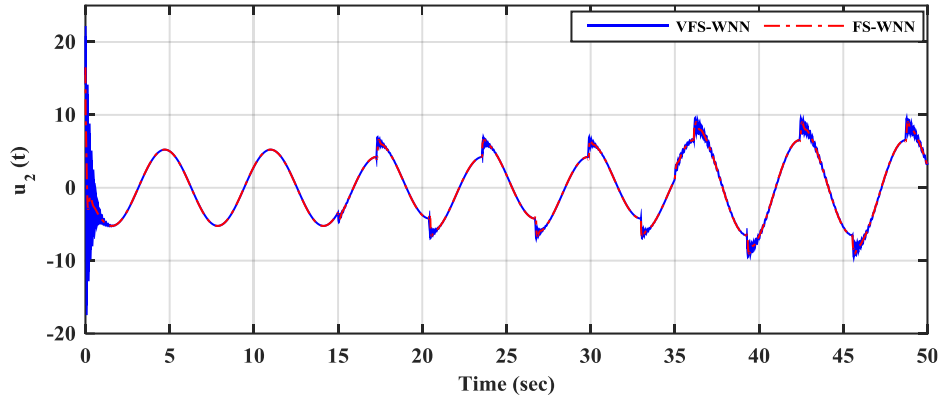


Figure 5.13 : Signal de commande u_2 en présence des perturbations et des variations paramétriques

En observant la Figure 5.10 et la Figure 5.11, on constate que les deux contrôleurs FS-WNN et VFS-WNN ont assuré une excellente poursuite de la trajectoire désirée en présence de perturbations et des variations paramétriques. Toutefois, la Figure 4.12 et la Figure 4.13 montrent des petites oscillations dans les signaux de commande générés par les deux contrôleurs au moment d'application de la perturbation à l'instant $t=15$ sec. Ces oscillations augmentent légèrement après l'application de la variation paramétrique à l'instant $t=35$ sec. Cependant, ces perturbations n'ont pas affecté la poursuite de la trajectoire désirée. D'où on peut conclure la robustesse de l'approche d'hybridation proposée dans le cadre de cette thèse face aux perturbations et variations des paramètres du système.

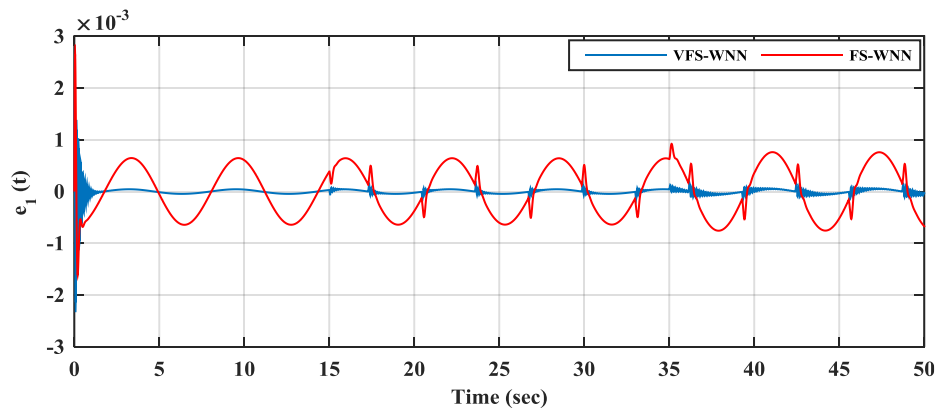


Figure 5.14 : Erreur de poursuite e_1 en présence des perturbations et des variations paramétriques

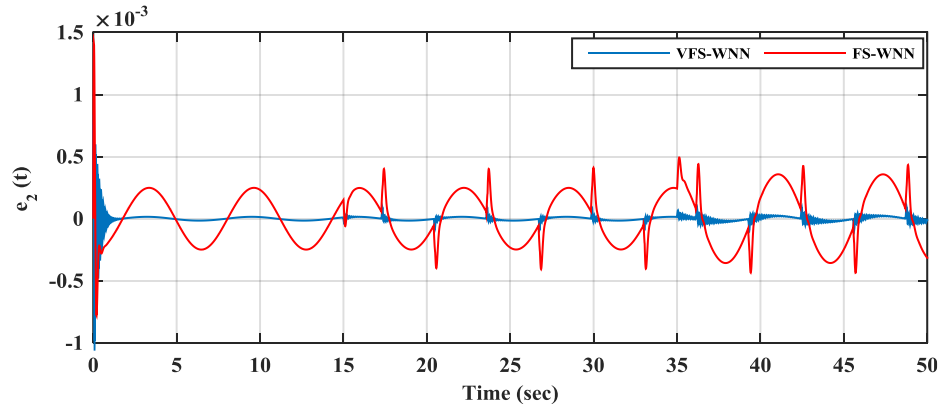


Figure 5.15 : Erreur de poursuite e_2 en présence des perturbations et des variations paramétriques

La Figure 5.14 et la Figure 5.15 montrent les erreurs de poursuites de la trajectoire désirée pour les deux articulations du robot manipulateur. La Figure 5.14 et la Figure 5.15 montrent une petite erreur de poursuite pour les deux articulations avec une légère amplification de l'erreur dans le cas du contrôleur FS-WNN. Par contre, le contrôleur VFS-WNN a assuré une erreur de poursuite infinitésimale sans aucune augmentation considérable après l'application de la perturbation et le changement d'un paramètre dans le modèle du système. A partir de ces observations, on remarque clairement l'avantage de l'hybridation de plusieurs systèmes intelligents et l'efficacité de l'ajustement des paramètres de modulation sur les performances du système de contrôle.

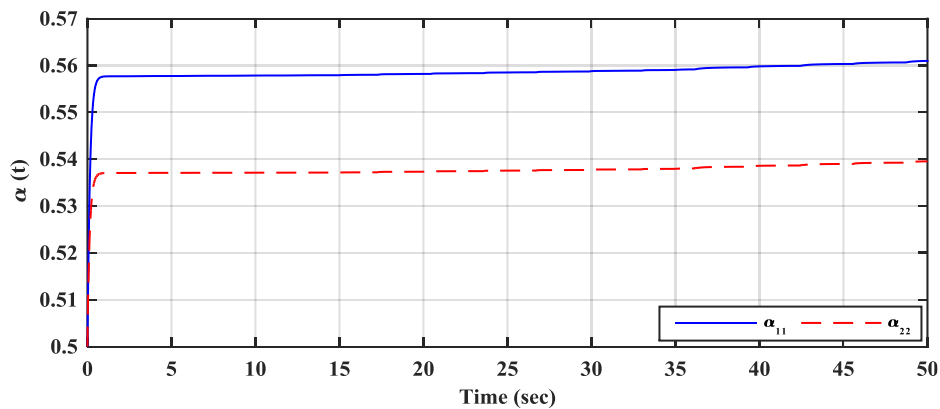


Figure 5.16 : Trajectoire des éléments diagonaux de facteur de modulation $\hat{\alpha}$ en fonction du temps en présence des perturbations et des variations paramétriques

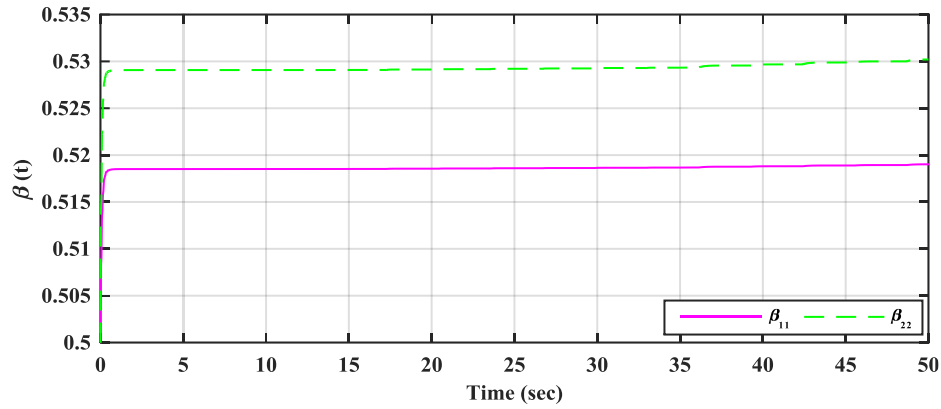


Figure 5.17 : Trajectoire des éléments diagonaux de facteur de modulation $\hat{\beta}$ en fonction du temps en présence des perturbations et des variations paramétriques

La Figure 5.15 et la Figure 5.16 présentent l'allure des éléments diagonaux des paramètres modulation $\hat{\alpha}$ et $\hat{\beta}$. On remarque une légère augmentation dans la valeur des paramètres $\hat{\alpha}_{11}$, $\hat{\alpha}_{22}$ et $\hat{\beta}_{11}$ après application des perturbations et d'une variation paramétrique. Cette augmentation a permis la compensation de l'effet des perturbations qui dégradent les performances du système de contrôle. on remarque de même que avec l'ajustement de ces paramètres, le contrôleur VFS-WNN a assuré une excellente poursuite, une erreur infinitésimale, et un signal de commande avec minimum d'effort.

Afin d'analyser avec précision les performances des contrôleurs FS-WNN et VFS-WNN dans le cas de variations paramétriques et de perturbations externes, leurs performances sont comparées en termes de leurs précisions de poursuite à l'aide de la fonction MSE et en fonction de l'énergie du signal de commande fournie par chaque approximateur. Les résultats obtenus sont donnés dans le Tableau 5.2 pour les deux articulations.

Tableau 5.2 : Comparaison des performances de contrôleur FS-WNN hybride-variable avec le contrôleur FS-WNN dans le cas de présence des perturbations externes et des variations paramétriques.

	Link 1		Link 2	
	MSE(y ₁)	E(u ₁)	MSE(y ₂)	E(u ₂)
FS-WNN	2.4599x10 ⁻⁷	4.3337x10 ⁵	4.8820x10 ⁻⁸	9.6053x10 ⁴
VFS-WNN	9.5746x10 ⁻⁹	4.4377x10 ⁶	1.9305x10 ⁻⁹	9.9363x10 ⁵

Le Tableau 5.2 compare les performances de contrôleur VFS-WNN hybride-variable avec le contrôleur FS-WNN hybride pour la première et la deuxième articulation. Le MSE de l'erreur de

poursuite et l'énergie du signal de commande ont été fixés comme des critères d'évaluation. En observant le MSE pour les deux articulations, nous remarquons que les deux contrôleurs FS-WNN hybride et le FS-WNN hybride-variable ont présenté une excellente poursuite avec une erreur de poursuite de l'ordre de 10^{-8} pour le contrôleur FS-WNN et de l'ordre 10^{-9} pour de contrôleur VFS-WNN. Le contrôleur basé sur l'approximateur VFS-WNN a donné un meilleure MSE grâce à l'ajustement en ligne des paramètres de modulation $\hat{\alpha}$ et $\hat{\beta}$. Les paramètres ajustés ont subi une petite augmentation après l'application de la perturbation et la variation paramétrique. Cette augmentation a permis de corriger l'effet de ces dernières et d'assurer la stabilité et la robustesse de système de contrôle.

5. Conclusion

Dans ce chapitre, on a développé un nouvel approximateur universel FS-WNN hybride-variable. Le FS-WNN est basé sur la combinaison des systèmes flous et les réseaux de neurones à base d'ondelettes en utilisant une nouvelle approche adaptative de la technique de modulation. L'intérêt de cette nouvelle approche est l'automatisation du choix des paramètres de modulation et l'ajustement en ligne de ces paramètres selon les besoins de contrôle. Cela permet un gain de temps et d'effort car il évite au constructeur de passer de temps à chercher les bons paramètres de modulation et de les réajuster si les conditions de fonctionnement changent. L'algorithme d'ajustement proposé dans cette partie de thèse se charge de chercher les paramètres optimaux de modulation en ligne en minimisant un critère d'erreur selon l'approche de stabilité de Lyapunov. Ce qui garantit une meilleure précision d'approximation et de contrôle tout en assurant la stabilité du système de commande. Cela offre au nouveau approximateur FS-WNN hybride-variable une grande capacité de généralisation grâce à l'association de différents approximateurs intelligents et encore plus de flexibilité grâce à l'adaptation entière de toutes ses paramètres en ligne face aux problèmes d'approximation et de contrôle des systèmes complexes incertains.

Dans le chapitre suivant, on envisage appliquer le nouveau approximateur FS-WNN hybride-variable pour le contrôle des quadrotors qui sont naturellement instables et très difficiles à commander. Ce qui rend l'établissement et l'application des lois de commandes pour ces systèmes un challenge dans le domaine de contrôle actuellement.

Chapitre 6

Commande adaptative directe par VFS-WNN hybride-variable de l'attitude de quadrotor

1. Introduction

Un quadrotor ou quadcopter appartient à la famille des véhicules aériens sans pilote (UAV). Le quadrotor est capable d'effectuer des décollages et atterrissages verticaux (VTOL), similaires aux hélicoptères typiques. Cependant, le système de contrôle diffère considérablement entre les hélicoptères et les quadrotors selon la dynamique du vol [103]. l'étude de marché de l'année 2020 effectué par Grand View Research a montré que l'utilisation des quadrotors dans l'industrie a considérablement augmenté au cours des dernières années et continuera de croître [104]. La raison revient à leur facilité d'utilisation parallèlement à leur large domaine d'application. En utilisant des quadrotors, il est possible d'effectuer des opérations qui sont nocives pour les humains ou qui nécessiteraient plus d'investissement de ressources si elles étaient effectuées par des humains. Par exemple, les quadrotors sont utilisés pour vérifier l'état des machines, des structures ou des infrastructures situées dans des zones éloignées ou à une altitude considérablement élevée par rapport au sol ou pour surveiller certaines zones. Les quadrotors sont de même utilisés dans le transport, la livraisons et même dans la collecte des données [105]. Une autre utilisation des quadrotors est la reconnaissance d'images et la cartographie 3D mobile, où le quadrotor prend des photos ou scanne certaines zones afin obtenir une version numérique de l'environnement réel étudié [106].

Le quadrotor contient quatre hélices dans une structure croisée. Les hélices sont installées sur des moteurs à courant continu et montées sur les bords des axes perpendiculaires. Chaque ensemble de deux hélices identiques montées sur le même axe tourne dans le même sens. Un

ensemble tourne dans le sens des aiguilles d'une montre, tandis que l'autre ensemble tourne dans le sens inverse des aiguilles d'une montre. La configuration du quadrotor est illustrée dans la Figure 6.1.

Le quadrotor est un système fortement non linéaire, couplé, instable et sous-actionné [107]–[109]. Le système de contrôle doit être suffisamment performant pour pouvoir contrôler les dynamiques complexes du quadrotor [110]. Jusqu'à présent, les stratégies de contrôle appliquées au quadrotor sont principalement divisées en contrôle dépendant de modèle du quadrotor et contrôle indépendant de modèle du quadrotor [111]. Le contrôle dépendant du modèle quadrotor nécessite principalement de connaître le modèle mathématique du quadrotor. La commande par mode glissant (SMC) est une méthode typique basée sur le modèle du système. Mu et al. [112] ont proposé une stratégie de commande par mode glissant intégral pour contrôler le suivi des points de navigations d'un quadrotor en présence d'incertitudes de modèle et de perturbations externes. En premier, le modèle linéaire du quadrotor a été dérivé et un contrôleur à la base de mode glissant intégral est synthétisé pour contrôler les angles de roulis et de tangage dans la boucle interne. Un contrôleur LQR a été utilisé pour assurer la poursuite des positions désirées dans la boucle externe. Dans Islam et al. [113], une étude comparative entre un contrôleur LQR et des contrôleurs LQG d'ordre un et deux a été proposée sur la base du modèle dynamique linéaire du quadrotor, le contrôleur LQG d'ordre deux a montré plus de robustesse face aux perturbations externes que les deux autres contrôleurs. Wang et Yang [114] ont proposé un schéma d'estimation de l'attitude d'un quadrotor à base de filtre de Kalman. Les estimations obtenues sont utilisées dans la synthèse d'un contrôleur proportionnel dérivé à double gains conçu pour réguler l'attitude du quadrotor. Une stratégie de commande par backstepping basée sur le modèle linéaire de quadrotor a été proposée par Mo et al. [115] pour le contrôle de l'altitude et l'attitude du quadrotor. Les résultats de simulation sur un package de pilotage automatique PX4 Pixhawk dans l'environnement de Matlab ont montré l'efficacité de la méthode proposée. Des tests expérimentaux ont été réalisés et la viabilité pratique de l'approche développée a été ainsi prouvée.

Avec les méthodes de contrôle linéaire mentionnées ci-dessus, tous les résultats expérimentaux et par simulation ont montré de bonnes performances de suivi et une bonne stabilité. Le problème est qu'en ingénierie réelle, la précision mathématique de modèle du système ne peut pas être obtenue en raison des incertitudes paramétriques et des dynamiques non linéaire non modélisée, qui surviennent de la variation de la masse et des moments d'inertie de la masse, les turbulences du vent, les défauts de l'actionneur qui s'ajoutent à ces problèmes et exigent également une

compensation appropriée. Par conséquent, il est difficile pour les contrôleurs qui reposent uniquement sur la dynamique précise du système d'assurer des bonnes performances en présence des perturbations inconnues et des paramètres incertains [116], [117]. Ces facteurs nécessitent des approches de contrôle non linéaires adaptatives et intelligentes pour compenser leurs effets sur le système de contrôle du quadrotor [118]. Pour faire face à de tels problèmes, les réseaux de neurones artificiels ont leur immense utilité dans l'approximation des dynamiques inconnues et non modélisées. Par conséquent, dans plusieurs approches de contrôle des quadrotors, les réseaux de neurones ont été intégrés aux systèmes de contrôle pour modéliser la dynamique non linéaire du quadrotor via des méthodes en ligne et hors ligne. Ce qui améliore la robustesse des contrôleurs de vol de quadrotor.

De nombreux chercheurs ont utilisé les réseaux de neurones dans la conception de contrôleurs pour les quadrotors et ils ont exploité leur propriété d'approximation pour apprendre la dynamique complexe non linéaire du quadrotor. Qiao et al. [119] ont proposé un contrôleur adaptatif neuronal par backstepping pour le contrôle d'un avion qui se dit « morphing ». Le mot morphing se réfère au changement de la forme de l'aile pendant le vol. dans cette approche, le réseau de neurones estime les dynamiques de l'avion et l'estimation obtenue est utilisée dans la synthèse du contrôleur par backstepping. Madani et Benallegue [120] ont présenté une vaste étude sur le contrôle par backstepping neuronal. L'un des principaux avantages de ce schéma de contrôle proposé est la non exigence d'un modèle exact du système et la connaissance des paramètres physiques n'est pas nécessaire. Le contrôleur proposé par Madani et Benallegue peut donc être implémenté sur n'importe quel quadrotor de masse et d'inertie différentes dans la même classe. De plus, le réglage des poids des réseaux de neurones a été effectué en ligne, ce qui ne nécessite aucune phase d'apprentissage préalable. Une étude de Dierks et Jagannathan [121] traite l'apprentissage en ligne de la dynamique non linéaire complète du modèle de quadrotor à travers des réseaux de neurones. L'apprentissage en ligne est préféré à l'apprentissage hors ligne car le quadrotor peut voler dans un environnement dynamique où différentes perturbations du pire cas se produiront et de tels scénarios de fonctionnement sont difficiles à générer hors ligne. Dans [122]–[125], des schémas de contrôle ont été développés pour que plusieurs quadrotors suivent un leader. Les réseaux de neurones sont utilisés pour approximer les dynamiques non modélisées pendant le vol.

En parallèle, une étude de Gao et al. [126] présente le contrôleur proportionnel dérivateur adaptatif flou. Indépendamment des bonnes performances des contrôleurs proportionnel dérivateur, ceux-ci doivent toujours intégrer l'aspect adaptatif pour des conditions de fonctionnement

incertaines. Pour atteindre cet objectif, le contrôleur proportionnel dérivateur a été étendu au proportionnel dérivateur adaptatif flou. De ce fait, les gains proportionnel et dérivé sont automatiquement ajustés par rapport au taux de signal d'erreur. Les résultats ont montré que le contrôleur proportionnel dérivateur adaptatif flou est plus rapide pour obtenir la réponse souhaitée par rapport au proportionnel dérivateur conventionnel. Cette étude montre l'intérêt de l'intégration des systèmes intelligents et l'adaptation en ligne aux approches de contrôle conventionnelles. L'étude de Niroumand et al. [127], s'est concentrée sur l'idée d'intégrer le backstepping intégral avec la logique floue pour améliorer les performances du contrôleur de vol. Les résultats de simulation ont montré une mise en œuvre supérieure de l'approche de backstepping intégral flou par rapport au backstepping intégral conventionnel. Le contrôle par mode glissant à base des systèmes flous pour les quadrotors a récemment été rapporté dans le travail de Pazooki et Mazinan [128] où les paramètres du système flou sont optimisés à l'aide d'un algorithme génétique. Le système flou était introduit pour compenser le dépassement « Overshoot » issu de l'application du contrôleur glissant seul.

Les quadrotors représentent un vrai challenge au niveau de la modélisation et de contrôle due à leur nature complexe et sous-actionnée. Les nombreuses approches de contrôle intelligent des quadrotors citées ci-dessus ont utilisé soit les systèmes flous ou les réseaux de neurones pour résoudre les problèmes liés à la variation des paramètres et des perturbations qui peuvent affecter le système de contrôle. Dans ce chapitre, on établit une présentation détaillée des dynamiques du quadrotor et on applique le nouveau approximateur FS-WNN hybride qui combine les réseaux de neurones et les systèmes flous dans une même structure pour commander l'attitude de ce système complexe. Pour valider l'efficacité du schéma de commande proposée dans ce chapitre, on implémente le schéma de commande développé dans un Toolbox Matlab PX4 Pixhawk. On utilise le Toolbox pour commander l'attitude d'un quadrotor équipé avec un système de pilotage PX4 Pixhawk par simulation et on établit des conclusions sur une éventuelle implémentation de ce schéma de commande sur un quadrotor réel.

2. Dynamiques du quadrotor

Le quadrotor a une configuration croisée et il possède quatre rotors à pas fixe à chaque extrémité. En se référant à la Figure 6.1, les rotors 1 et 3 constituent une paire et les rotors 2 et 4 constituent la deuxième paire. Les deux paires de rotors tournent dans le sens contraire pour équilibrer le moment total de rotation du corps. Les états de sortie à contrôler sont plus que les

entrées disponibles dans un quadrotor, il constitue donc un système sous-actionné (moins d'actionneurs ayant plus de DOF à contrôler) et englobe une instabilité inhérente. Six états de sortie qui doivent être contrôlés sont constitués de trois états de position cartésiens (x , y et z) et trois angles d'Euler (roulis (φ), tangage (θ) et lacet (ψ)). Les entrées sont des termes de poussée et de couple générés par quatre combinaisons différentes des vitesses d'hélices. La configuration du quadrotor est indiquée dans la Figure. 6.1 [129].

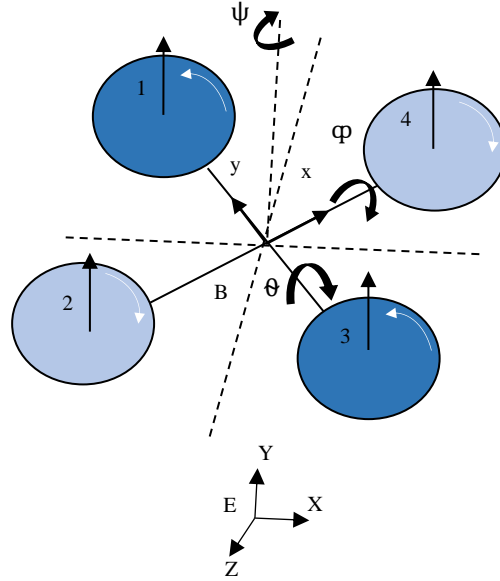


Figure 6.1 : Configuration du quadrotor

Deux repères sont définis pour décrire le mouvement du quadrotor. Le repère inertiel $E(O, \vec{e}_x, \vec{e}_y, \vec{e}_z)$ et le repère lié au corps du quadrotor $B(O_b, \vec{e}_x, \vec{e}_y, \vec{e}_z)$. Le vecteur ξ de la position linéaire et le vecteur η des angles d'Euler du quadrotor sont donnés dans le repère inertiel comme suit :

$$\xi = (x, y, z)^T \quad (6.1)$$

$$\eta = (\varphi, \theta, \psi)^T \quad (6.2)$$

Soit $V = (u, v, w)^T$ la vitesse linéaire et $\Omega = (p, q, r)^T$ la vitesse angulaire exprimée dans le repère B lié au corps du quadrotor, respectivement. Donc:

$$V = R^T(\eta)\dot{\xi} \quad (6.3)$$

$$\Omega = N(\eta)\dot{\eta} \quad (6.4)$$

Avec $R(\eta)$ et $N(\eta)$ sont les matrices de rotation et de transformation, respectivement :

$$R = \begin{bmatrix} c\psi c\theta & -s\psi c\varphi + c\psi s\theta s\varphi & s\psi s\varphi + c\psi s\theta c\varphi \\ s\psi c\theta & c\psi c\varphi + s\psi s\theta s\varphi & -c\psi s\varphi + s\psi s\theta c\varphi \\ -s\theta & c\theta s\varphi & c\theta c\varphi \end{bmatrix} \quad (6.5)$$

$$N = \begin{bmatrix} 1 & 0 & -s\theta \\ 0 & c\varphi & c\theta s\varphi \\ 0 & -s\varphi & c\varphi c\theta \end{bmatrix} \quad (6.6)$$

Les hypothèses suivantes sont prises en considération dans la modélisation du quadrotor :

1. Le quadrotor est un corps rigide symétrique.
2. Les hélices sont rigides.
3. Les forces de poussée et de traînée sont proportionnelles au carré de la vitesse de rotation des hélices.

En utilisant le formalisme de Newton-Euler. Les dynamiques du quadrotor sont données comme suit:

$$\begin{bmatrix} mL_3 & 0 \\ 0 & J \end{bmatrix} \begin{bmatrix} \dot{V} \\ \dot{\Omega} \end{bmatrix} + \begin{bmatrix} \Omega x m V \\ \Omega x J \Omega \end{bmatrix} = \begin{bmatrix} \Sigma F_{ext} \\ \Sigma T_{ext} \end{bmatrix} \quad (6.7)$$

Où m est la masse du corps rigide et $J = \text{diag}(I_x, I_y, I_z)$ est la matrice d'inertie. Ainsi, les dynamiques du vol de quadrotor sous les forces aérodynamiques et les moments sont décrites par les équations suivantes :

$$\Sigma F_{ext} = F_{prop} + F_{aero} + F_{grav} \quad (6.8)$$

$$\Sigma T_{ext} = T_{prop} + T_{aero} + T_{gyro} \quad (6.9)$$

F_{prop} est la résultante des forces F_i produites par les quatre rotors :

$$F_{prop} = \sum_{i=1}^4 F_i e_3 \quad (6.10)$$

$$F_i = k_b \omega_i^2 \quad (6.11)$$

Avec $e_3 = (0,0,1)^T$. Et k_b est une constante positive de portance, ω_i est la vitesse de rotor.

Le quadrotor génère un moment qui est décrit dans le cadre du repère lié à son corps comme suit:

$$T_{prop} = \begin{bmatrix} r(F_3 - F_1) \\ r(F_4 - F_2) \\ c_d \sum_{i=1}^4 ((-1)^{i+1} F_i) \end{bmatrix} \quad (6.12)$$

Où r est la distance entre le centre de gravité du quadrotor et l'axe de rotation de l'hélice. $c_d = k_d/k_b$ est le coefficient de traînée.

Les couples de frottements aérodynamiques sont donnés par :

$$\begin{cases} F_{aero} = -A_F(U) \\ T_{aero} = -A_T(U) \end{cases} \quad (6.13)$$

Avec $A_F = [A_u \ A_v \ A_w]$, $A_T = [A_p \ A_q \ A_r]$ et $A_i = \frac{1}{2} \rho_{air} C_i U^2$. telque ρ_{air} est la densité d'air, C_i est un coefficient aérodynamique et U est la vitesse du quadrotor relativement à l'air.

L'effet de la gravité est donné par:

$$F_{grav} = -mgR^T e_3 \quad (6.14)$$

Les effets gyroscopiques sont donnés par l'équation suivante :

$$T_{gyro} = - \sum_{i=1}^4 (\Omega \times e_3) ((-1)^i J_r \omega_i) \quad (6.15)$$

Avec J_r est l'inertie motrice.

Le modèle dynamique complet qui décrit le quadrotor est donné comme suit :

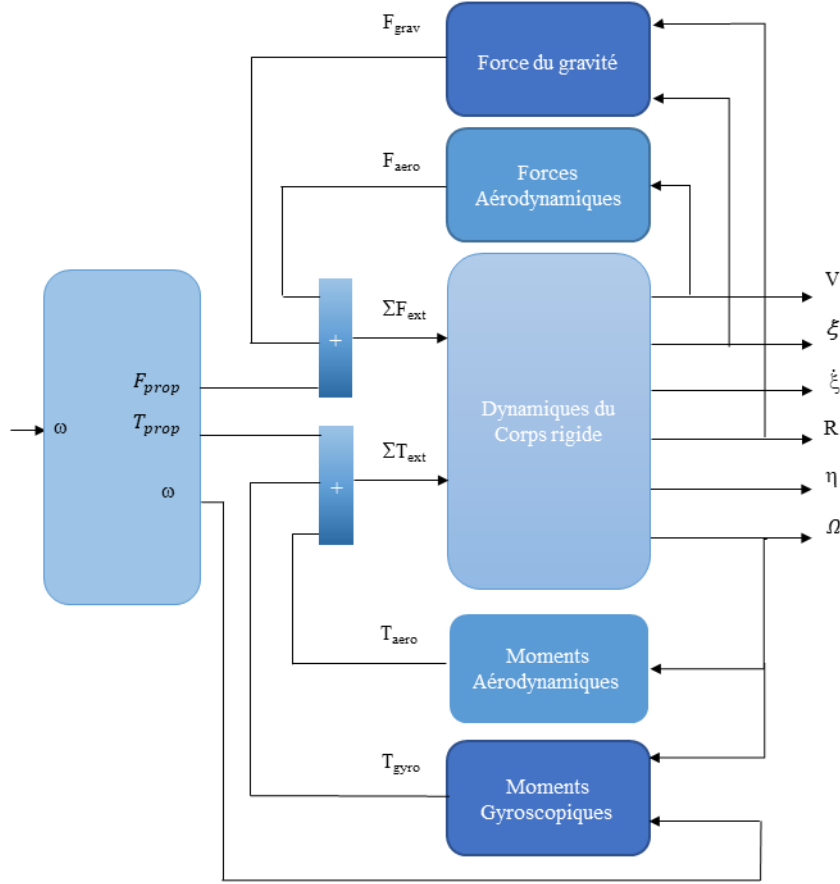


Figure 6.2 : Architecture du modèle dynamique du quadrotor

Les dynamiques du quadrotor peuvent être exprimées sous une forme compacte comme suit :

$$\dot{x}_i = f_i(\mathbf{x}) + g_i(\mathbf{x})u_i \quad (6.16)$$

Avec $\mathbf{x} = [\varphi \dot{\varphi} \theta \dot{\theta} \psi \dot{\psi} X \dot{X} Y \dot{Y} Z \dot{Z}]$ est le vecteur d'état, qui est supposé disponible à la mesure (observable). $u = [u_1 \ u_2 \ u_3]^T$ est le signal de commande. $f_i(\mathbf{x})$ et $g_i(\mathbf{x})$ sont des fonctions non linéaires inconnues et $g_i(\mathbf{x})$ est supposée lentement variable dans le temps. On s'intéresse par le contrôle de l'attitude du quadrotor donc, le vecteur d'état est sélectionné comme $\mathbf{x} = [\varphi \ \dot{\varphi} \ \theta \ \dot{\theta} \ \psi \ \dot{\psi}]$ et le vecteur de sortie est donné par $\mathbf{y} = [y_1 \ y_2 \ y_3]^T = [\varphi \ \theta \ \psi]$.

3. Contrôle par VFS-WNN hybride-variable de l'attitude du quadrotor

Dans ce chapitre, nous proposons un contrôleur FS-WNN hybride-variable pour la stabilisation du quadrotor. Contrairement aux contrôleurs classiques qui dépend des paramètres identifiés du

système pour la synthèse du contrôleur, nous utilisons un contrôleur FS-WNN hybride-variable adaptatif pour contrôler en ligne l'attitude du quadrotor. Ainsi, le contrôleur proposé est indépendant du modèle du système et ne nécessite pas une connaissance préalable de la dynamique du quadrotor pour déterminer ses paramètres inconnus. De plus, les paramètres estimés sont ajustés en ligne pour assurer une performance optimale du système de contrôle. Les lois d'adaptations des paramètres sont dérivées à l'aide de la méthode directe de Lyapunov pour garantir la stabilité du système en boucle fermée. Par conséquent, le contrôleur proposé a une architecture simple et stable. Les résultats de la simulation sont présentés pour mettre en évidence les performances du contrôleur développé.

L'objectif de contrôle est de concevoir une loi de commande de telle sorte que la sortie du système y_i suit une trajectoire désirée y_{di} .

Définissant l'erreur de poursuite comme suit :

$$e_i(t) = y_{d_i}(t) - y_i(t) \quad i=1, \dots, 3 \quad (6.17)$$

La dérivée par rapport au temps de l'erreur de poursuite est donnée par :

$$\dot{e}_i(t) = \dot{y}_{d_i}(t) - \dot{y}_i(t) \quad i=1, \dots, 3 \quad (6.18)$$

On définit l'erreur filtrée s_i telle que:

$$s_i(t) = \dot{e}_i(t) + \lambda_i e_i(t) \quad (6.19)$$

La dérivée par rapport au temps de l'erreur filtrée s_i est donnée comme suit :

$$\dot{s}_i(t) = \ddot{e}_i(t) + \lambda_i \dot{e}_i(t) \quad (6.20)$$

On remplace (6.16) dans (6.20), on obtient :

$$\dot{s}_i(t) = \ddot{y}_{d_i}(t) - f_i(x) - g_i(x)u_i + \lambda_i \dot{e}_i(t) \quad (6.21)$$

Dans le cas où les fonctions du systèmes $f_i(x)$ et $g_i(x)$ sont connues avec précision, la loi de commande idéale $u_i^*(t)$ est synthétisée comme suit:

$$u_i^*(t) = g_i(x)^{-1}(-f_i(x) + \ddot{y}_{d_i}(t) + \lambda_i \dot{e}_i(t)) \quad (6.22)$$

Cependant, les non-linéarités du système $f_i(x)$ et $g_i(x)$ sont inconnues. Ainsi, la loi idéale de contrôle $u_i^*(t)$ ne peut pas être mise en œuvre. Pour surmonter ce problème, la loi idéale de contrôle u_i^* est approximée par une loi de commande intelligente adaptative hybride comme suit:

$$u_i^* = Y_1^{*i}(x(t), K^i) + \varepsilon^i(x(t), \alpha_i^*) \quad (6.23)$$

Avec k_i , k_i et k_i sont les gains proportionnel, intégral et dérivé du PID, respectivement. On peut développer (6.21) comme suit :

$$u_i^* = [\alpha_i^* K_{FS}^{i*} \Phi^i(x(t)) + (1 - \alpha_i^*) K_{WNN}^{i*} \psi^{*i}(x(t), c^{*i}) + \varepsilon^i(x(t), \alpha_i^*)] \quad (6.24)$$

où c_k^{*i} est le paramètre idéale de translation de l'ondelette ψ_k^* par rapport à l'entrée x_k dans le sous-système i . K_{WNN}^{i*} le vecteur idéale des paramètres de sortie du réseau WNN et K_{FS}^{i*} le vecteur idéale des paramètres de sortie du réseau FS. Avec

$$\varepsilon^i(x(t), \alpha_i^*) = [\alpha_i^* \varepsilon_{FS}^i(x) + (1 - \alpha_i^*) \varepsilon_{WNN}^i(x)] \quad (6.25)$$

Où ε^i représente l'erreur d'approximation de la loi de commande et elle est supposée bornée par $\bar{\varepsilon}^i > 0$. nous utilisons le FS-WNN hybride-variable pour estimer en ligne la loi de commande idéale u_i^* . L'architecture du contrôleur proposé est présentée dans la Figure. 6.3 où $x_i = [x_i(t), \dots, x_n(t)]^T$ est le vecteur d'entrées du réseau WNN et du réseau FS. u_{WNNi} est la sortie du réseau WNN et u_{FSi} est la sortie du réseau FS. Le vecteur K_{WNN}^i est le vecteur des paramètres adaptatifs de sortie du réseau WNN et le vecteur K_{FS}^i est le vecteur des paramètres adaptatifs de sortie du réseau FS. m est le nombre des nœuds cachés et L est le nombre des règles floues. les fonctions ψ_k^i et Φ_k^i sont choisie comme fonction ondelettes de type Chapeau Mexicain comme dans les chapitres précédents.

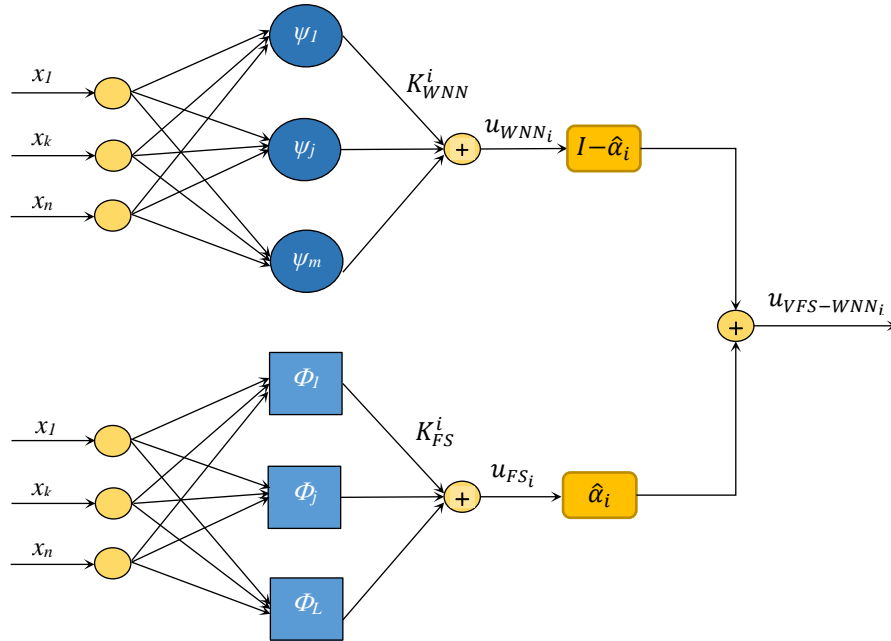


Figure 6.3 : Architecture du contrôleur adaptatif direct à base du VFS-WNN hybride-variable.

Les sorties du réseau WNN et du réseau FS sont calculées comme suit :

$$\begin{aligned} u_{FSi}(x) &= \sum_{k=1}^L k_{J_{FS}}^i \Phi_k^i(x) = K_{FS}^{iT} \Phi^i(x) \\ u_{WNNi}(x) &= \sum_{k=1}^m k_{J_{WNN}}^i \psi_k^i(x, c_j^i) = K_{WNN}^{iT} \Psi^i(x, C^i) \end{aligned} \quad (6.26)$$

Et la loi de commande globale du contrôleur VFS-WNN est calculée comme suit :

$$u_{VFS-WNNi}(x) = \alpha_i K_{FS}^{iT} \Phi^i(x) + (1 - \alpha_i) K_{WNN}^{iT} \Psi^i(x, C^i) \quad (6.27)$$

Définissant les erreurs d'approximation des paramètres comme étant $\tilde{K}^i = K^{*i} - \hat{K}^i$. On rajoute un terme robuste à la loi de commande (6.29) pour assurer la robustesse du contrôleur :

$$u_{VFS-WNNi}(x) = \alpha_i K_{FS}^{iT} \Phi^i(x) + (1 - \alpha_i) K_{WNN}^{iT} \Psi^i(x, C^i) + \hat{h}_{1i} s_i + \hat{h}_{2i} \text{sgn}(s_i) \quad (6.28)$$

Avec $\lambda_i, \hat{h}_i \geq 0$, On rajoute et on enlève le terme $g_i(x)u_i^*$ à (6.20), on obtient :

$$\dot{s}_i(t) = \ddot{y}_{d_i}(t) - f_i(x) - g_i(x)u_i + \lambda_i \dot{e}_i(t) + g_i(x)u_i^* - g_i(x)u_i^* \quad (6.29)$$

En remplaçant par (6.22) dans (6.31), on trouve :

$$\dot{s}_i(t) = g_i(x)(u_i^* - u_i) \quad (6.30)$$

A partir de (6.24) et (6.30) :

$$\begin{aligned} \dot{s}_i(t) &= g_i(x) \left(\alpha_i^* K_{FS}^{*iT} \Phi^i(x) + (1 - \alpha_i^*) K_{WNN}^{*iT} \Psi^i(x, C^{*i}) + \varepsilon^i(x, \alpha_i^*) - \right. \\ &\left. \alpha_i K_{FS}^{iT} \Phi^i(x) - (1 - \alpha_i) K_{WNN}^{iT} \Psi^i(x^i, C^i) \right) - g_i(x) (\hat{h}_{1i} s_i + \hat{h}_{2i} \text{sgn}(s_i)) \end{aligned} \quad (6.31)$$

En suivant la même analogie utilisée dans le chapitre cinq, l'expression (6.32) devient comme suit :

$$\begin{aligned} \dot{s}_i(t) &= g_i(x) \left(\hat{\alpha}_i \tilde{K}_{FS}^{iT} \Phi^i(x) + \tilde{\alpha}_i \left(\tilde{K}_{FS}^{iT} \Phi^i(x) - \tilde{K}_{WNN}^{iT} \tilde{\Psi}^i(x, \tilde{C}^i) \right) + (1 - \right. \\ &\left. \hat{\alpha}_i) \tilde{K}_{WNN}^{iT} \tilde{\Psi}^i(x, \tilde{C}^i) + (1 - \hat{\alpha}_i) \tilde{K}_{WNN}^{iT} \left(\tilde{\Psi}^i(x, \hat{C}^i) + \tilde{\Psi}^i(x, \tilde{C}^i) \right) + \tilde{\alpha}_i \tilde{K}_{FS}^{iT} \Phi^i(x) - \right. \\ &\left. \tilde{\alpha}_i \tilde{K}_{WNN}^{iT} \tilde{\Psi}^i(x, \tilde{C}^i) + \varepsilon^i(x, \alpha_i^*) \right) - g_i(x) (\hat{h}_{1i} s_i + \hat{h}_{2i} \text{sgn}(s_i)) \end{aligned} \quad (6.32)$$

En utilisant la linéarisation de Taylor [77] (voir annexe A), on trouve que :

$$\tilde{\Psi}^i(x, \tilde{C}^i) = \Omega^{iT} \tilde{C}^i + O^i = \Omega^{iT} (C^{*i} - \hat{C}^i) + O^i \quad (6.33)$$

Et si on applique (6.34) dans (6.33), on obtient :

$$\begin{aligned} \dot{s}_i(t) &= g_i(x) \left(\hat{\alpha}_i \tilde{K}_{FS}^{iT} \Phi^i(x) + \tilde{\alpha}_i \left(\tilde{K}_{FS}^{iT} \Phi^i(x) - \tilde{K}_{WNN}^{iT} \tilde{\Psi}^i(x, \hat{C}^i) \right) + (1 - \right. \\ &\left. \hat{\alpha}_i) \tilde{K}_{WNN}^{iT} \left(\Omega^{iT} \tilde{C}^i + O^i \right) + (1 - \hat{\alpha}_i) \tilde{K}_{WNN}^{iT} \left(\tilde{\Psi}^i(x, \hat{C}^i) + \Omega^{iT} (C^{*i} - \hat{C}^i) + O^i \right) + \right. \\ &\left. \tilde{\alpha}_i \tilde{K}_{FS}^{iT} \Phi^i(x) - \tilde{\alpha}_i \tilde{K}_{WNN}^{iT} \tilde{\Psi}^i(x, \tilde{C}^i) + \varepsilon^i(x, \alpha_i^*) \right) - g_i(x) (\hat{h}_{1i} s_i + \hat{h}_{2i} \text{sgn}(s_i)) \end{aligned} \quad (6.34)$$

On peut de même écrire (6.35) sous cette forme :

$$\begin{aligned} \dot{s}_i(t) = & g_i(x) \left(\hat{\alpha}_i \tilde{K}_{FS}^{iT} \Phi^i(x) + \tilde{\alpha}_i \left(\hat{K}_{FS}^{iT} \Phi^i(x) - \hat{K}_{WNN}^{iT} \hat{\Psi}^i(x, \hat{C}^i) \right) + (1 - \right. \\ & \hat{\alpha}_i) \hat{K}_{WNN}^{iT} \Omega^{iT} \tilde{C}^i + (1 - \hat{\alpha}_i) K_{WNN}^{*iT} O^i + (1 - \hat{\alpha}_i) \tilde{K}_{WNN}^{iT} \left(\hat{\Psi}^i(x, \hat{C}^i) - \Omega^{iT} \hat{C}^i \right) + \\ & \left. (1 - \hat{\alpha}_i) \tilde{K}_{WNN}^{iT} \Omega^{iT} C^{*i} + \tilde{\alpha}_i \tilde{K}_{FS}^{iT} \Phi^i(x) - \tilde{\alpha}_i \tilde{K}_{WNN}^{iT} \hat{\Psi}^i(x, \tilde{C}^i) + \varepsilon^i(x, \alpha_i^*) \right) - \\ & g_i(x) (\hat{h}_{1i} s_i + \hat{h}_{2i} \text{sgn}(s_i)) \end{aligned} \quad (6.35)$$

Admettant que $\mathcal{L}^i = (1 - \hat{\alpha}_i) K_{WNN}^{*iT} O^i + (1 - \hat{\alpha}_i) \tilde{K}_{WNN}^{iT} \Omega^{iT} C^{*i} + \tilde{\alpha}_i \tilde{K}_{FS}^{iT} \Phi^i(x) - \tilde{\alpha}_i \tilde{K}_{WNN}^{iT} \hat{\Psi}^i(x, \tilde{C}^i) + \varepsilon^i(x, \alpha_i^*)$. L'expression (6.35) devient :

$$\begin{aligned} \dot{s}_i(t) = & g_i(x) \left[\hat{\alpha}_i \tilde{K}_{FS}^{iT} \Phi^i(x) + \tilde{\alpha}_i \left(\hat{K}_{FS}^{iT} \Phi^i(x) - \hat{K}_{WNN}^{iT} \hat{\Psi}^i(x, \hat{C}^i) \right) + (1 - \right. \\ & \hat{\alpha}_i) \hat{K}_{WNN}^{iT} \Omega^{iT} \tilde{C}^i + (1 - \hat{\alpha}_i) \tilde{K}_{WNN}^{iT} \left(\hat{\Psi}^i(x, \hat{C}^i) - \Omega^{iT} \hat{C}^i \right) \left. \right] + g_i(x) \mathcal{L}^i_{VFS-WNN} - \\ & g_i(x) (\hat{h}_{1i} s_i + \hat{h}_{2i} \text{sgn}(s_i)) \end{aligned} \quad (6.36)$$

Hypothèse 10:

L'erreur d'approximation est supposée être bornée par $\|\mathcal{L}^i_{VFS-WNN}\| \leq \bar{\mathcal{L}}^i_{VFS-WNN}$ [78].

Pour atteindre les objectifs de contrôle, on doit d'ajuster les paramètres de l'approximateur FS-WNN hybride-variable pour que $e(t) \rightarrow 0$ quand $t \rightarrow \infty$. Ainsi, les lois d'ajustement des paramètres du réseau FS-WNN hybride-variable sont données comme suit [96]:

$$\dot{\hat{K}}_{FS}^i = \Upsilon_{K_{FS}^i} \hat{\alpha}_i \Phi^i(x) s_i - 2\tau_{K_{FS}^i} |s_i| \hat{K}_{FS}^i \quad (6.37)$$

$$\dot{\hat{K}}_{WNN}^i = \Upsilon_{K_{WNN}^i} (1 - \hat{\alpha}_i) \left(\hat{\Psi}^i(x, \hat{C}^i) - \Omega^{iT} \hat{C}^i \right) s_i - 2\tau_{K_{WNN}^i} |s_i| \hat{K}_{WNN}^i \quad (6.38)$$

$$\dot{\hat{C}}^i = \Upsilon_{C^i} (1 - \hat{\alpha}_i) \Omega^i \hat{K}_{WNN}^i s_i - 2\tau_{C^i} |s_i| \hat{C}^i \quad (6.39)$$

$$\dot{\hat{\alpha}}_i = \Upsilon_{\alpha_i} \left(\hat{K}_{FS}^{iT} \Phi^i(x) - \hat{K}_{WNN}^{iT} \hat{\Psi}^i(x, \hat{C}^i) \right) s_i - 2\tau_{\alpha_i} |s_i| \hat{\alpha}_i \quad (6.40)$$

À ce niveau, l'analyse de stabilité du système de commande de la Figure 6.5 qui comprend le modèle complet du quadrotor de la Figure. 6.2, et le contrôleur indiqué dans la Figure. 6.3 est détaillée.

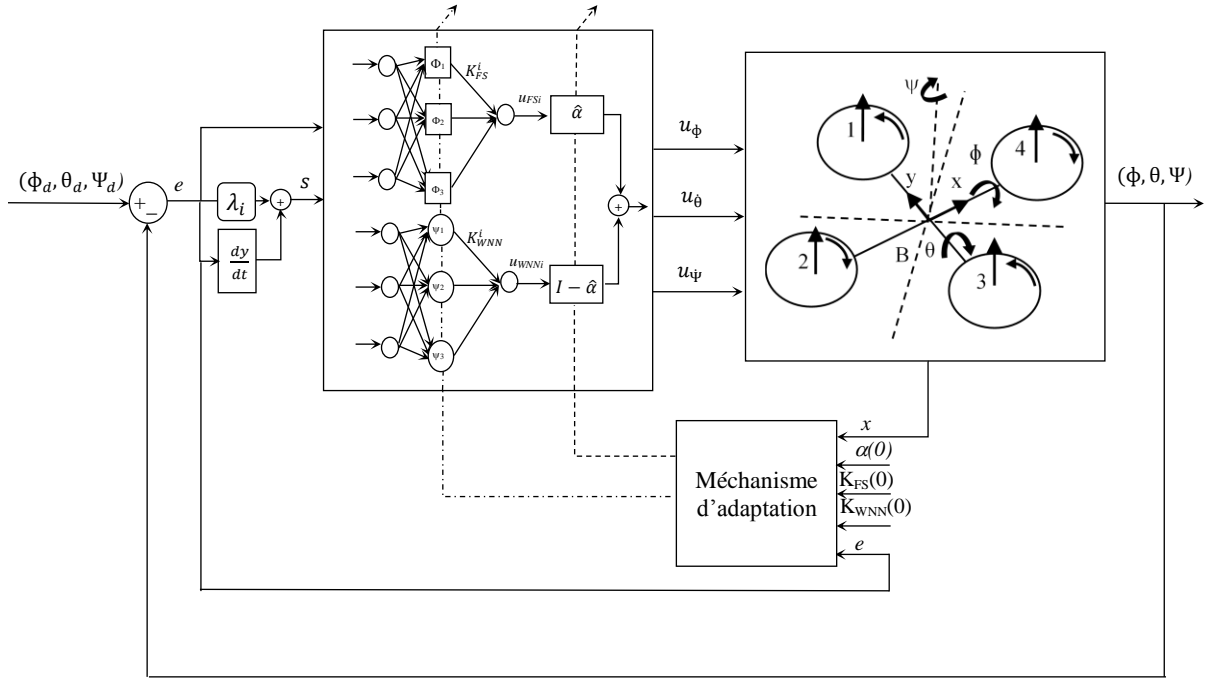


Figure 6.4 : Architecture d'une boucle de contrôle d'un quadrotor par FS-WNN hybride-variable

Théorème 6 :

Considérant le système MIMO non linéaire incertain (6.7). La loi de commande hybride est choisie comme (6.28) et les lois d'adaptation des paramètres sont données par (6.38)-(6.41). Le contrôleur FS-WNN hybride-variable adaptatif direct garantit que le système de commande en boucle fermée est stable, et que tous les signaux sont bornés.

Preuve du théorème 6 :

On choisit une fonction candidate de Lyapunov comme suit:

$$V_i = \frac{g_i^{-1}}{2} s_i^2 + \frac{1}{2\Upsilon_{K_{FS}^i}} \tilde{K}_{FS}^{iT} \tilde{K}_{FS}^i + \frac{1}{2\Upsilon_{K_{WNN}^i}} \tilde{K}_{WNN}^{iT} \tilde{K}_{WNN}^i + \frac{1}{2\Upsilon_{C^i}} tr(\tilde{C}^{iT} \tilde{C}^i) + \frac{1}{2\Upsilon_{\alpha_i}} \tilde{\alpha}_i^2 \quad (6.41)$$

La dérivée par rapport au temps de fonction candidate de Lyapunov (6.41) est donnée par:

$$\dot{V}_i = g_i^{-1} s_i \dot{s}_i + \frac{1}{\Upsilon_{K_{FS}^i}} \tilde{K}_{FS}^{iT} \dot{\tilde{K}}_{FS}^i + \frac{1}{\Upsilon_{K_{WNN}^i}} \tilde{K}_{WNN}^{iT} \dot{\tilde{K}}_{WNN}^i + \frac{1}{\Upsilon_{C^i}} tr(\tilde{C}^{iT} \dot{\tilde{C}}^i) + \frac{1}{\Upsilon_{\alpha_i}} \tilde{\alpha}_i \dot{\tilde{\alpha}}_i \quad (6.42)$$

Si on remplace par l'expression (6.36) dans (6.41), on obtient :

$$\begin{aligned}
 \dot{V}_i = & g_i^{-1} s_i \left(g_i(x) \left(\hat{\alpha}_i \tilde{K}_{FS}^{iT} \Phi^i(x) + \tilde{\alpha}_i \left(\tilde{K}_{FS}^{iT} \Phi^i(x) - \tilde{K}_{WNN}^{iT} \hat{\Psi}^i(x, \hat{C}^i) \right) \right. \right. \\
 & + (1 - \hat{\alpha}_i) \tilde{K}_{WNN}^{iT} \Omega^{iT} \tilde{C}^i + (1 - \hat{\alpha}_i) \tilde{K}_{WNN}^{iT} \left(\hat{\Psi}^i(x, \hat{C}^i) - \Omega^{iT} \hat{C}^i \right) + \mathcal{J}^i_{VFS-WNN} \\
 & \left. \left. - (\hat{h}_{1i} s_i + \hat{h}_{2i} \text{sgn}(s_i)) \right) \right) + \frac{1}{\Upsilon_{K_{FS}^i}} \tilde{K}_{FS}^{iT} \dot{\tilde{K}}_{FS}^i + \frac{1}{\Upsilon_{K_{WNN}^i}} \tilde{K}_{WNN}^{iT} \dot{\tilde{K}}_{WNN}^i \\
 & + \frac{1}{\Upsilon_{C^i}} \text{tr} \left(\tilde{C}^{iT} \dot{\tilde{C}}^i \right) + \frac{1}{\Upsilon_{\alpha_i}} \tilde{\alpha}_i \dot{\tilde{\alpha}}_i
 \end{aligned} \quad (6.43)$$

On développe (6.43) comme suit :

$$\begin{aligned}
 \dot{V}_i = & \hat{\alpha}_i \tilde{K}_{FS}^{iT} \Phi^i(x) s_i + \frac{1}{\Upsilon_{K_{FS}^i}} \tilde{K}_{FS}^{iT} \dot{\tilde{K}}_{FS}^i + \tilde{\alpha}_i \left(\tilde{K}_{FS}^{iT} \Phi^i(x) - \tilde{K}_{WNN}^{iT} \hat{\Psi}^i(x, \hat{C}^i) \right) s_i + \frac{1}{\Upsilon_{\alpha_i}} \tilde{\alpha}_i \dot{\tilde{\alpha}}_i \\
 & + (1 - \hat{\alpha}_i) \tilde{K}_{WNN}^{iT} \Omega^{iT} \tilde{C}^i s_i + \frac{1}{\Upsilon_{C^i}} \text{tr} \left(\tilde{C}^{iT} \dot{\tilde{C}}^i \right) + (1 - \hat{\alpha}_i) \tilde{K}_{WNN}^{iT} \left(\hat{\Psi}^i(x, \hat{C}^i) - \Omega^{iT} \hat{C}^i \right) s_i \\
 & + \frac{1}{\Upsilon_{K_{WNN}^i}} \tilde{K}_{WNN}^{iT} \dot{\tilde{K}}_{WNN}^i + \left(\mathcal{J}^i_{VFS-WNN} - (\hat{h}_{1i} s_i + \hat{h}_{2i} \text{sgn}(s_i)) \right) s_i
 \end{aligned} \quad (6.44)$$

On regroupe les termes correspondants comme suit:

$$\begin{aligned}
 \dot{V}_i = & \tilde{K}_{FS}^{iT} \left(\dot{\tilde{K}}_{FS}^i + \Upsilon_{K_{FS}^i} \hat{\alpha}_i \Phi^i(x) s_i \right) + \tilde{\alpha}_i \left(\dot{\tilde{\alpha}}_i + \Upsilon_{\alpha_i} \left(\tilde{K}_{FS}^{iT} \Phi^i(x) - \tilde{K}_{WNN}^{iT} \hat{\Psi}^i(x, \hat{C}^i) \right) s_i \right) + \\
 & \tilde{C}^{iT} \text{tr} \left(\dot{\tilde{C}}^i + \Upsilon_{C^i} (1 - \hat{\alpha}_i) \Omega^i \tilde{K}_{WNN}^i s_i \right) + \tilde{K}_{WNN}^{iT} \left(\dot{\tilde{K}}_{WNN}^i + \Upsilon_{K_{WNN}^i} (1 - \hat{\alpha}_i) \left(\hat{\Psi}^i(x, \hat{C}^i) - \right. \right. \\
 & \left. \left. \Omega^{iT} \hat{C}^i \right) s_i \right) + \left(\mathcal{J}^i_{VFS-WNN} - (\hat{h}_{1i} s_i + \hat{h}_{2i} \text{sgn}(s_i)) \right) s_i
 \end{aligned} \quad (6.45)$$

En utilisant (6.38) à (6.41) :

$$\begin{aligned}
 \dot{V}_i = & \tilde{K}_{FS}^{iT} \left(2\tau_{K_{FS}^i} |s_i| \tilde{K}_{FS}^i \right) + \tilde{\alpha}_i \left(2\tau_{\alpha_i} |s_i| \tilde{\alpha}_i \right) + \tilde{C}^{iT} \left(2\tau_{C^i} |s_i| \tilde{C}^i \right) + \\
 & \tilde{K}_{WNN}^{iT} \left(2\tau_{K_{WNN}^i} |s_i| \tilde{K}_{WNN}^i \right) + \left(\mathcal{J}^i_{VFS-WNN} - (\hat{h}_{1i} s_i + \hat{h}_{2i} \text{sgn}(s_i)) \right) s_i
 \end{aligned} \quad (6.46)$$

Admettant que :

$$\begin{aligned}
 \tilde{K}_{FS}^{iT} \tilde{K}_{FS}^i & \leq -\frac{1}{2} \|\tilde{K}_{FS}^i\|^2 + \frac{1}{2} \|K_{FS}^{i*}\|^2 \\
 \tilde{\alpha}^{iT} \tilde{\alpha}^i & \leq -\frac{1}{2} \|\tilde{\alpha}_i\|^2 + \frac{1}{2} \|\alpha_i^*\|^2 \\
 \tilde{C}^{iT} \tilde{C}^i & \leq \tilde{K}_{FS}^{iT} - \frac{1}{2} \|\tilde{C}^i\|^2 + \frac{1}{2} \|C^{i*}\|^2 \\
 \tilde{K}_{WNN}^{iT} \tilde{K}_{WNN}^i & \leq -\frac{1}{2} \|\tilde{K}_{WNN}^i\|^2 + \frac{1}{2} \|K_{WNN}^{i*}\|^2
 \end{aligned} \quad (6.47)$$

Si on prend en considération les inégalités de (6.47), l'équation (6.46) peut se mettre sous la forme suivante :

$$\begin{aligned}
 \dot{V}_i \leq & \left(\tau_{K_{FS}^i} |s_i| \|K_{FS}^{i*}\|^2 - \tau_{K_{FS}^i} |s_i| \|\tilde{K}_{FS}^i\|^2 \right) + \left(\tau_{\alpha_i} |s_i| \|\alpha_i^*\|^2 - \tau_{\alpha_i} |s_i| \|\tilde{\alpha}_i\|^2 \right) \\
 & + \left(\tau_{C^i} |s_i| \|C^{i*}\|^2 - \tau_{C^i} |s_i| \|\tilde{C}^i\|^2 \right) + \left(\tau_{K_{WNN}^i} |s_i| \|K_{WNN}^{i*}\|^2 - \tau_{K_{WNN}^i} |s_i| \|\tilde{K}_{WNN}^i\|^2 \right) \quad (6.48) \\
 & + \left(\mathcal{I}_{VFS-WNN}^i - (\mathfrak{h}_{1i} s_i + \mathfrak{h}_{2i} \text{sgn}(s_i)) \right) s_i
 \end{aligned}$$

Prenant $|s_i|$ comme facteur en commun, on obtient :

$$\begin{aligned}
 \dot{V}_i \leq & \left(\tau_{K_{FS}^i} \|K_{FS}^{i*}\|^2 + \tau_{\alpha_i} \|\alpha_i^*\|^2 + \tau_{C^i} \|C^{i*}\|^2 + \tau_{K_{WNN}^i} \|K_{WNN}^{i*}\|^2 \right) |s_i| \\
 & - \left(\tau_{K_{FS}^i} \|\tilde{K}_{FS}^i\|^2 + \tau_{\alpha_i} \|\tilde{\alpha}_i\|^2 + \tau_{K_{WNN}^i} \|\tilde{K}_{WNN}^i\|^2 + \tau_{C^i} \|\tilde{C}^i\|^2 \right) |s_i| \quad (6.49) \\
 & + \left(\mathcal{I}_{VFS-WNN}^i - (\mathfrak{h}_{1i} s_i + \mathfrak{h}_{2i} \text{sgn}(s_i)) \right) s_i
 \end{aligned}$$

En utilisant l'hypothèse 10, l'expression (6.49) devient :

$$\begin{aligned}
 \dot{V}_i \leq & -\mathfrak{h}_{1i} s_i^2 - \left(\mathfrak{h}_{2i} - \left(\tau_{K_{FS}^i} \|K_{FS}^{i*}\|^2 + \tau_{\alpha_i} \|\alpha_i^*\|^2 + \tau_{C^i} \|C^{i*}\|^2 + \right. \right. \\
 & \left. \left. \tau_{K_{WNN}^i} \|K_{WNN}^{i*}\|^2 + \bar{\mathcal{I}}_{VFS-WNN}^i \right) \right) |s_i| - \left(\tau_{K_{FS}^i} \|\tilde{K}_{FS}^i\|^2 + \tau_{\alpha_i} \|\tilde{\alpha}_i\|^2 + \right. \quad (6.50) \\
 & \left. \tau_{K_{WNN}^i} \|\tilde{K}_{WNN}^i\|^2 + \tau_{C^i} \|\tilde{C}^i\|^2 \right) |s_i|
 \end{aligned}$$

Sachant que les paramètres \mathfrak{h}_{1i} et \mathfrak{h}_{2i} sont choisis par le concepteur, \mathfrak{h}_{2i} doit être choisi de telle sorte que $\dot{V} \leq 0$. C'est-à-dire : $\mathfrak{h}_{2i} \geq \left(\tau_{K_{FS}^i} \|K_{FS}^{i*}\|^2 + \tau_{\alpha_i} \|\alpha_i^*\|^2 + \tau_{C^i} \|C^{i*}\|^2 + \tau_{K_{WNN}^i} \|K_{WNN}^{i*}\|^2 + \bar{\mathcal{I}}_{VFS-WNN}^i \right)$. Par conséquent, le système en boucle fermée de la Figure 6.4 est stable. Ainsi, tous les signaux sont bornés. Cela signifie que l'erreur de poursuite e_i et tous ses dérivés jusqu'à l'ordre $n - 1$ sont bornées, également s_i est bornée, c'est-à-dire, $s_i \in L_\infty$. Dans ce cas-là, la fonction \dot{V}_i de l'équation (6.50) est bornée comme suit :

$$\dot{V}_i \leq -\mathfrak{h}_i s_i^2 \quad (6.51)$$

Où \mathfrak{h}_i est une constante positive. L'équation (6.51) implique que V_i et les vecteurs d'erreurs des paramètres \tilde{K}_{FS}^i , \tilde{K}_{WNN}^i , \tilde{C}^i , $\tilde{\alpha}_i$ sont bornés.

4. Résultats de simulation

Dans cette étude, le contrôleur FS-WNN hybride-variable est construit selon (6.28), et il est appliqué pour contrôler l'attitude du quadrotor. On commence en premier lieu par une étude comparative entre le FS-WNN hybride avec des paramètres de modulation fixes $\alpha_i=0.5$. Ensuite, on étudie le cas des paramètres de modulation adaptatifs avec deux initialisations différentes. Le

premier cas α est initialisée comme $\alpha=0.9$ et le deuxième cas pour $\alpha=0.1$. Pour analyser les performances du contrôleur VFS-WNN hybride-variable, ses performances sont comparées au PID [109]. Les paramètres du quadrotor sont donnés dans le Tableau 6.1.

Tableau 6.1 : Paramètres du quadrotor

Paramètre	Valeur
Masse de quadrotor : m	0.6 kg
Gravité : g	9.81 N m ²
Diamètre : r	0.24 m
Inertie du quadrotor : J	$[8.5 * 10^{-3}, 8.5 * 10^{-3}, 2.3 * 10^{-2}]^T$ kg m ²
Densité de l'air : ρ_{air}	1.225
Coefficient de portance : k_b	$3.13 * 10^{-5}$ N s ²
Coefficient aéro dynamique ($u \ v \ w$) : C_F	$[5.567 * 10^{-4}, 5.5 * 10^{-4}, 6.354 * 10^{-4}]^T$
Coefficient aérodynamique (pqr) : C_T	$[7.567 * 10^{-4}, 7.5 * 10^{-4}, 6.354 * 10^{-4}]^T$
Coefficient de trainé : c_d	$7.5 * 10^{-7}$ N m s ²
Inertie du moteur : J_r	$6 * 10^{-5}$ kg m ²

a) Cas du FS-WNN hybride avec un paramètre de modulation $\alpha=0.9$ fixe

Les objectifs de contrôle sont d'amener le roulis, le tangage et le lacet à suivre les trajectoires désirées. Les performances de poursuite des roulis et tangage sont illustrées dans la Figure 6.5 et la Figure 6.6, respectivement, dans le cas du contrôleur PID et du contrôleur FS-WNN hybride-variable. La Figure 6.7 montre les performances de poursuite du lacet dans les deux cas aussi.

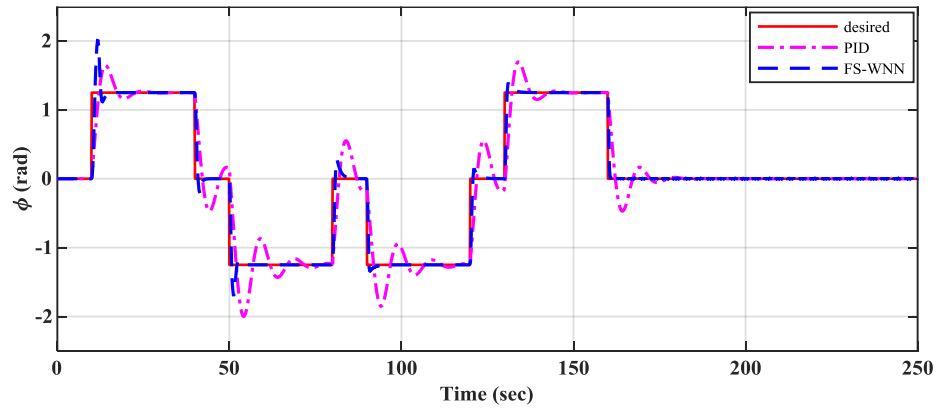


Figure 6.5 : Performances de poursuite de roulis par le contrôleur FS-WNN hybride et le PID

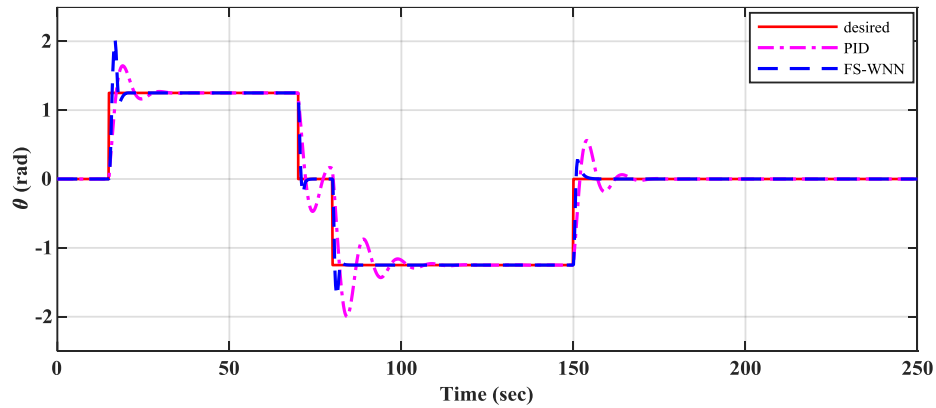


Figure 6.6 : Performances de poursuite de tangage par le contrôleur FS-WNN hybride et le PID

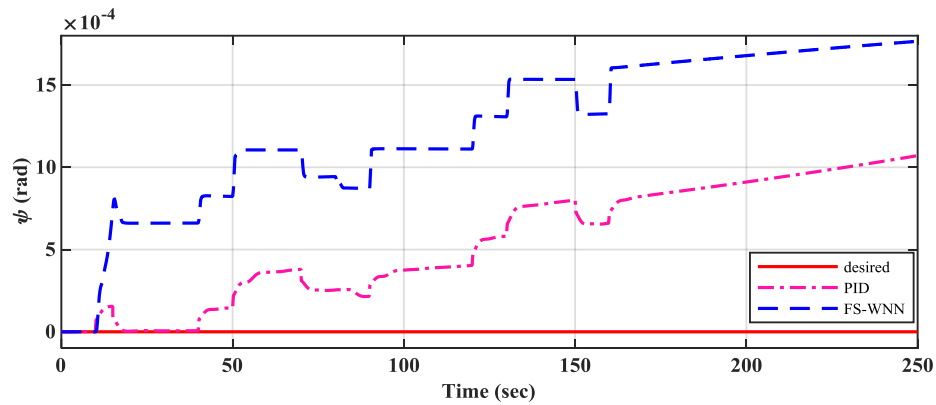


Figure 6.7 : Performances de poursuite de lacet par le contrôleur FS-WNN hybride et le PID

La Figure 6.5, la Figure 6.6 et la Figure 6.7 montrent les performances de poursuite des angles de roulis, tangage et lacet, respectivement, du quadrotor dans le cas du contrôleur PID et du contrôleur FS-WNN hybride avec un paramètre de modulation α fixé à 0.5. Sur la base de ces Figures, on peut voir que le roulis, le tangage et le lacet suivent avec précision les trajectoires désirées dans le

cas du contrôleur FS-WNN hybride. Cependant, dans le cas du contrôleur PID, des fortes oscillations sont présentes dans le régime transitoire et le temps d'établissement est long, ce qui dégrade la précision de poursuite. On remarque cette influence en observant les erreurs de poursuite dans la Figure 6.8 pour le roulis, la Figure 6.9 pour le tangage et la Figure 6.10 pour le lacet.

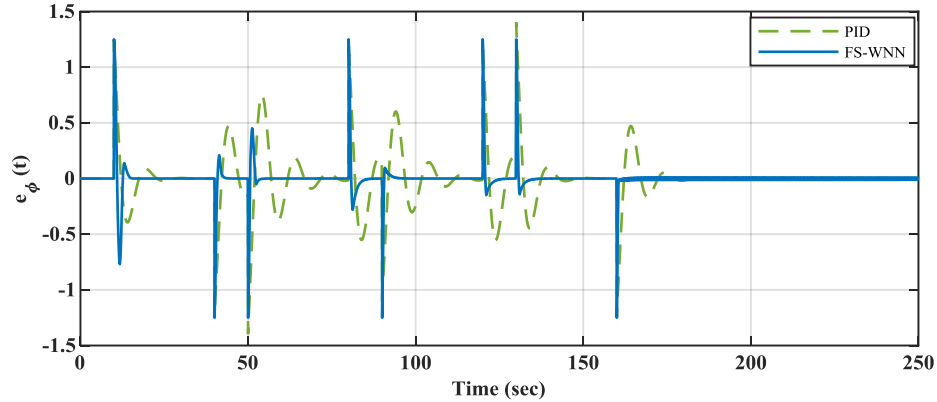


Figure 6.8 : Erreur de poursuite de roulis par le contrôleur FS-WNN hybride et le PID

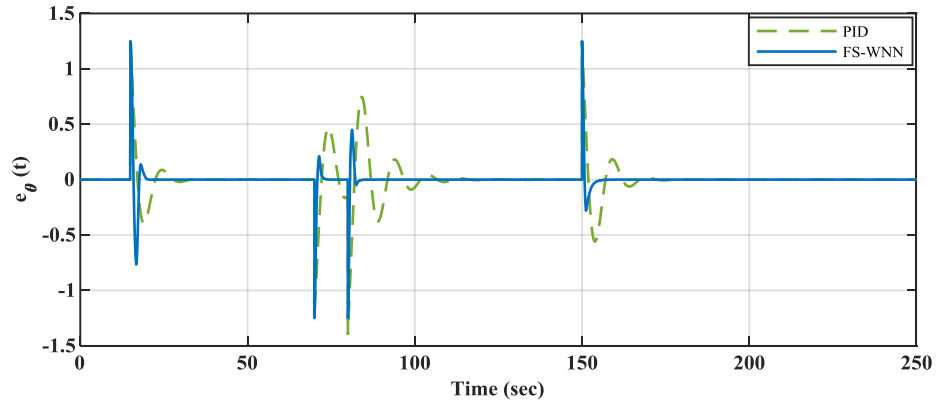


Figure 6.9 : Erreur de poursuite de tangage par le contrôleur FS-WNN hybride et le PID

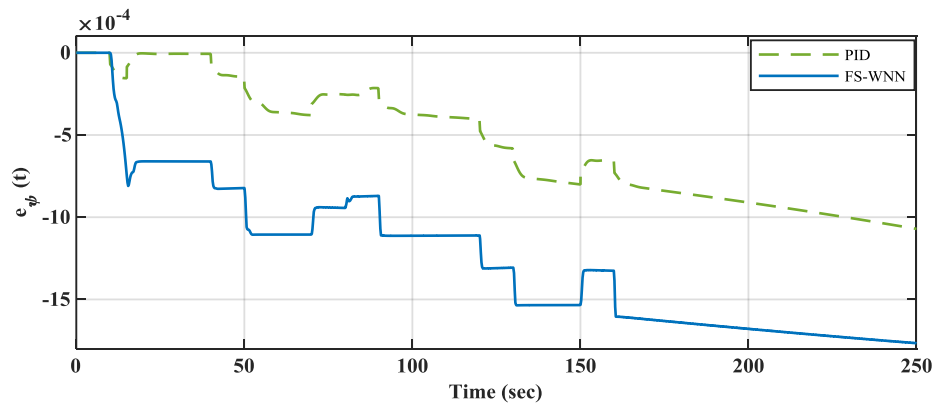


Figure 6.10 : Erreur de poursuite de lacet par le contrôleur FS-WNN hybride et le PID

On observe à partir de la Figure 6.8, la Figure 6.9 et la Figure 6.10, les erreurs de poursuite du contrôleur FS-WNN hybride-variable et du contrôleur PID. Dans le cas du contrôleur FS-WNN hybride-variable, pendant le changement de l'attitude, les erreurs de poursuite sont rapidement corrigées par le mécanisme d'adaptation. Le contrôleur PID avec des paramètres proportionnel, intégral et dérivé fixes n'a pas bien suivi les trajectoires désirées et la réponse du quadrotor était oscillatoire avec un temps d'établissement long. Malgré que le contrôleur FS-WNN hybride-variable a une architecture simple, il a montré des meilleures performances pendant le changement de l'attitude en terme de dépassement et de temps d'établissement par rapport au PID. De ce fait, l'hybridation adaptative de deux systèmes intelligents a prouvé son efficacité pour le contrôle de l'attitude du quadrotor. L'effort de contrôle de roulis est illustré dans la Figure 6.11 pour le roulis, dans la Figure 6.12 pour le tangage et dans la Figure 6.13 pour le lacet dans le cas du contrôleur FS-WNN hybride-variable avec un paramètre de modulation α fixe d'une valeur de 0.9 et dans le cas du contrôleur PID.

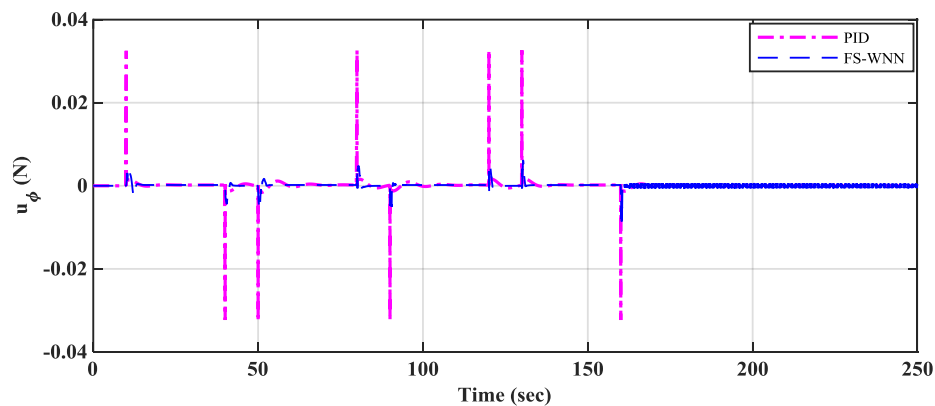


Figure 6.11: Signal de commande de roulis

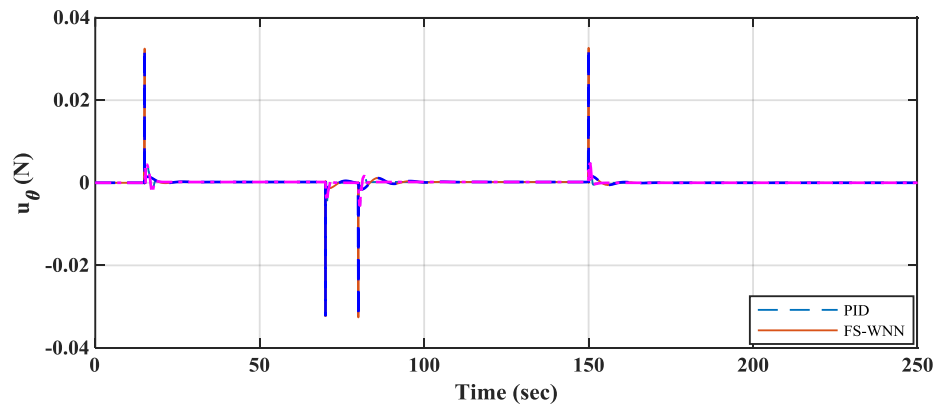


Figure 6.12: Signal de commande de tangage

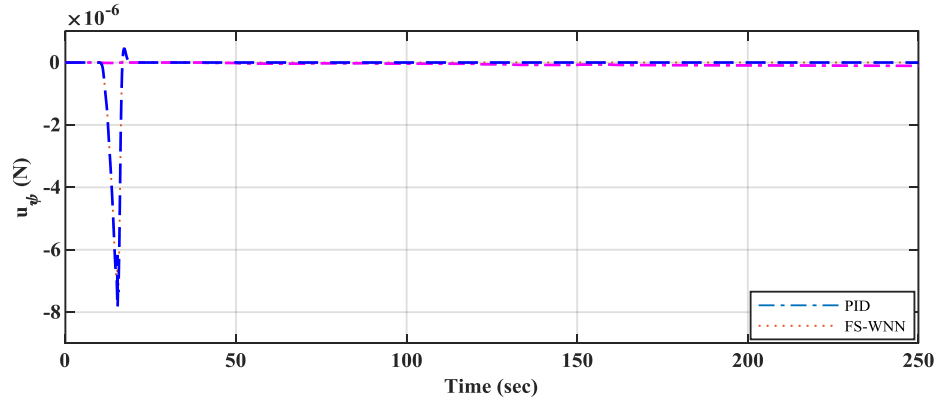


Figure 6.13 : Signal de commande de lacet

En observant le signal de commande de roulis de la Figure 6.11 pour le contrôleur FS-WNN hybride, on remarque que le signal de commande n'est pas lisse et il présente des oscillations permanentes. Pour investiguer la source de ces oscillations, on a illustré les forces de commande dans la Figure 6.14 pour le contrôleur FS-WNN hybride et le contrôleur PID.

La Figure 6.14 montre que les oscillations dans le signal de commande du contrôleur FS-WNN hybride sont originaires des forces F_2 et F_4 . Ces deux forces contrôlent le mouvement de tangage principalement et elles influencent le mouvement de roulis produit par les forces F_1 et F_3 .

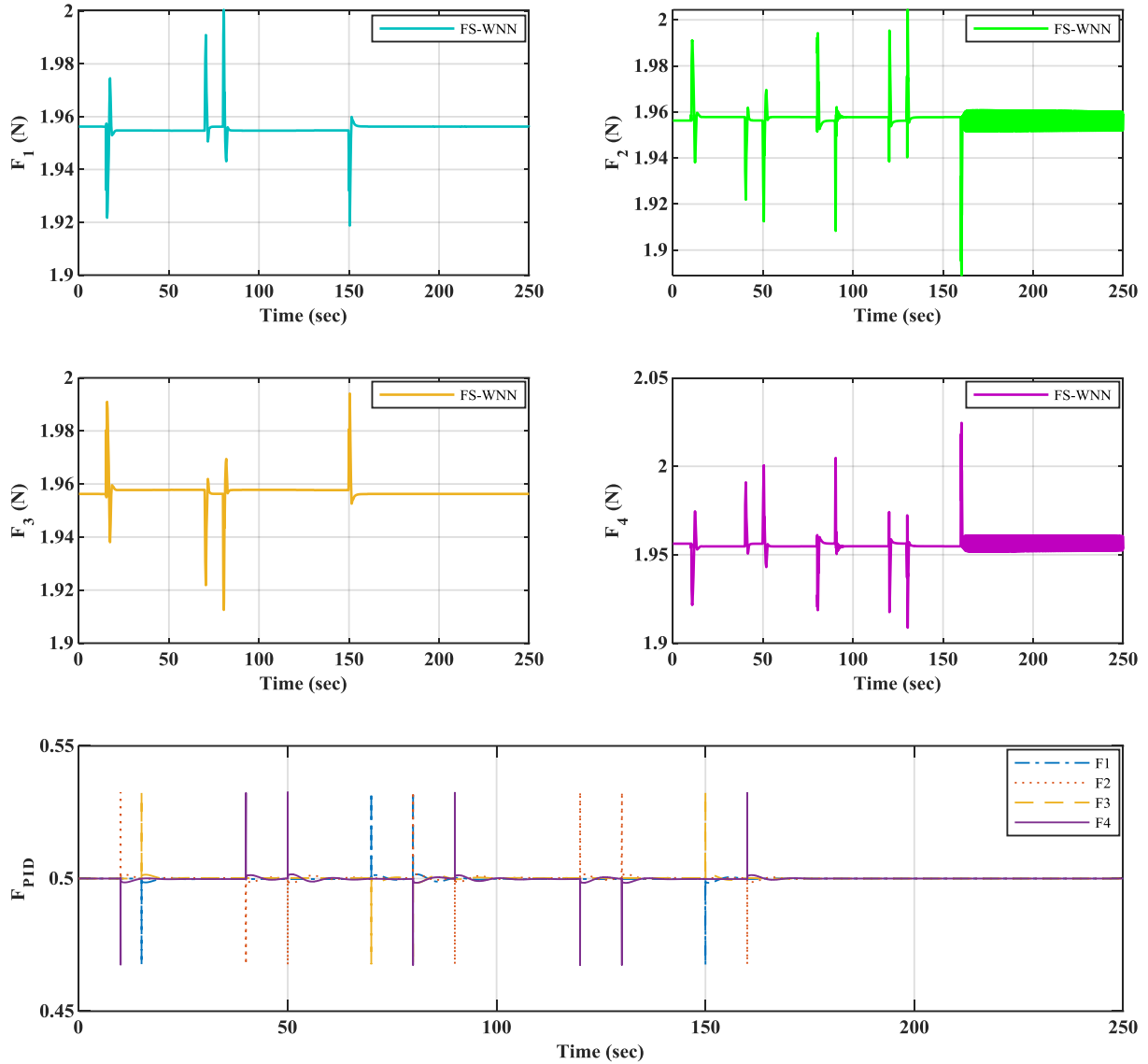


Figure 6.14: Forces de commande du contrôleur FS-WNN hybride et le contrôleur PID

Pour améliorer les performances du contrôleur FS-WNN hybride et afin de corriger le problème des oscillations observé dans le mouvement de roulis, on remplace le paramètre de modulation α fixe par un paramètre α adaptatif. On initialise la valeur de α par 0.9 en premier lieu et on observe son influence sur les performances du contrôleur FS-WNN hybride-variable.

b) Cas du VFS-WNN hybride-variable avec un paramètre de modulation $\hat{\alpha}$ adaptatif initialisé à 0.9

Les performances de poursuite des roulis et tangage sont illustrées dans la Figure 6.15 et la Figure 6.16, respectivement, dans le cas du contrôleur PID et du contrôleur FS-WNN hybride-variable. La Figure 6.15 montre les performances de poursuite du lacet dans les deux contrôleurs.

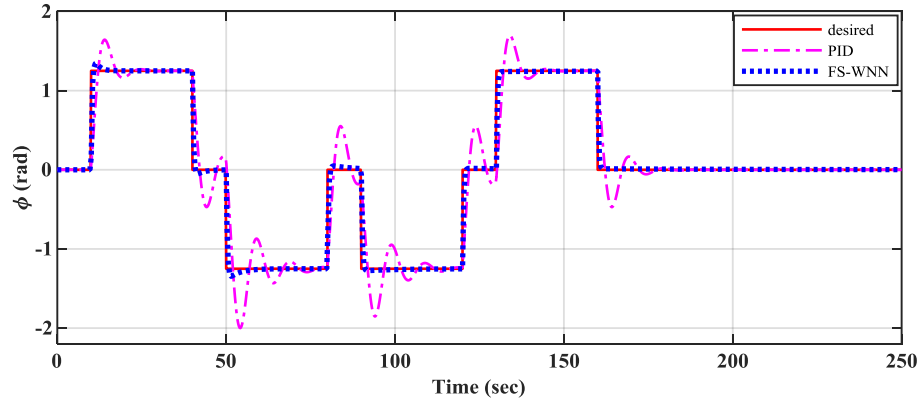


Figure 6.16 : Performances de poursuite de roulis par le contrôleur FS-WNN hybride et le contrôleur PID

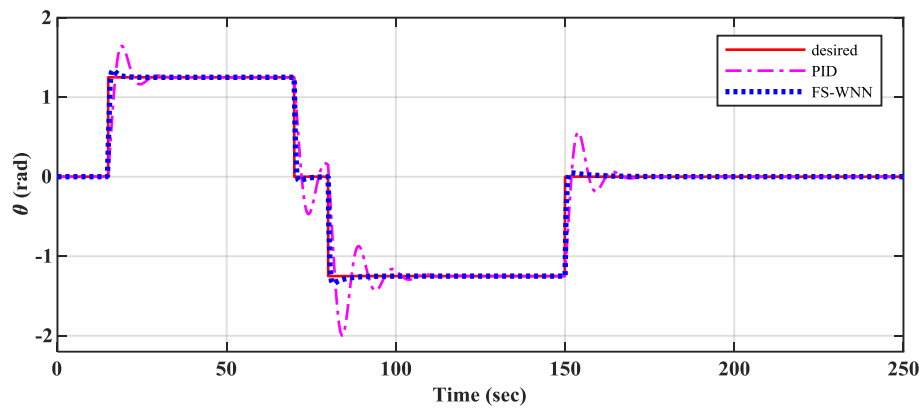


Figure 6.17 : Performances de poursuite de tangage par le contrôleur FS-WNN hybride et le contrôleur PID

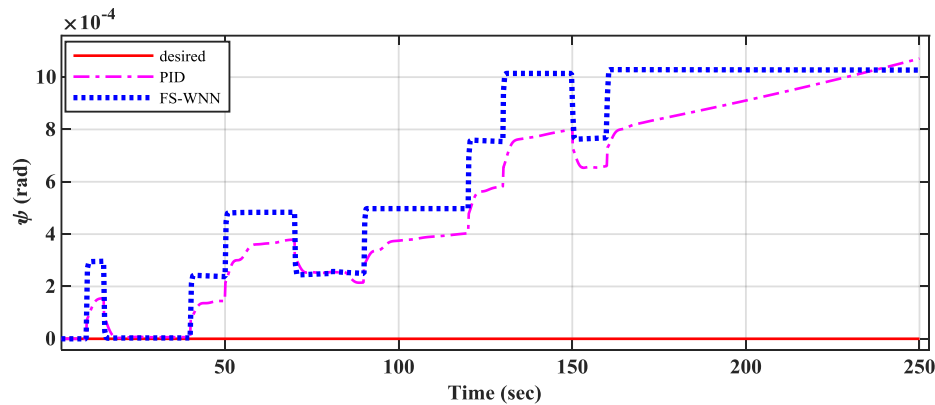


Figure 6.18: Performances de poursuite de roulis par le contrôleur FS-WNN hybride et le contrôleur PID

Les forces de contrôle qui gèrent les différents mouvements de quadrotor sont données dans la Figure 6.18. la supériorité du contrôleur FS-WNN hybride-variable est démontrée. Les sorties du système sous contrôle par FS-WNN hybride-variable suivent avec précision les trajectoires désirées. On remarque que grâce à l'adaptation du paramètre de modulation, la poursuite de roulis est largement améliorée en régime transitoire et les oscillations sont complètement supprimées en

régime permanent. Contrairement au contrôleur PID qui maintient une réponse oscillatoire et un dépassement important (overshoot/undershoot) car ses paramètres fixes ne permettent pas une adaptation en temps réel pour corriger la réponse du système.

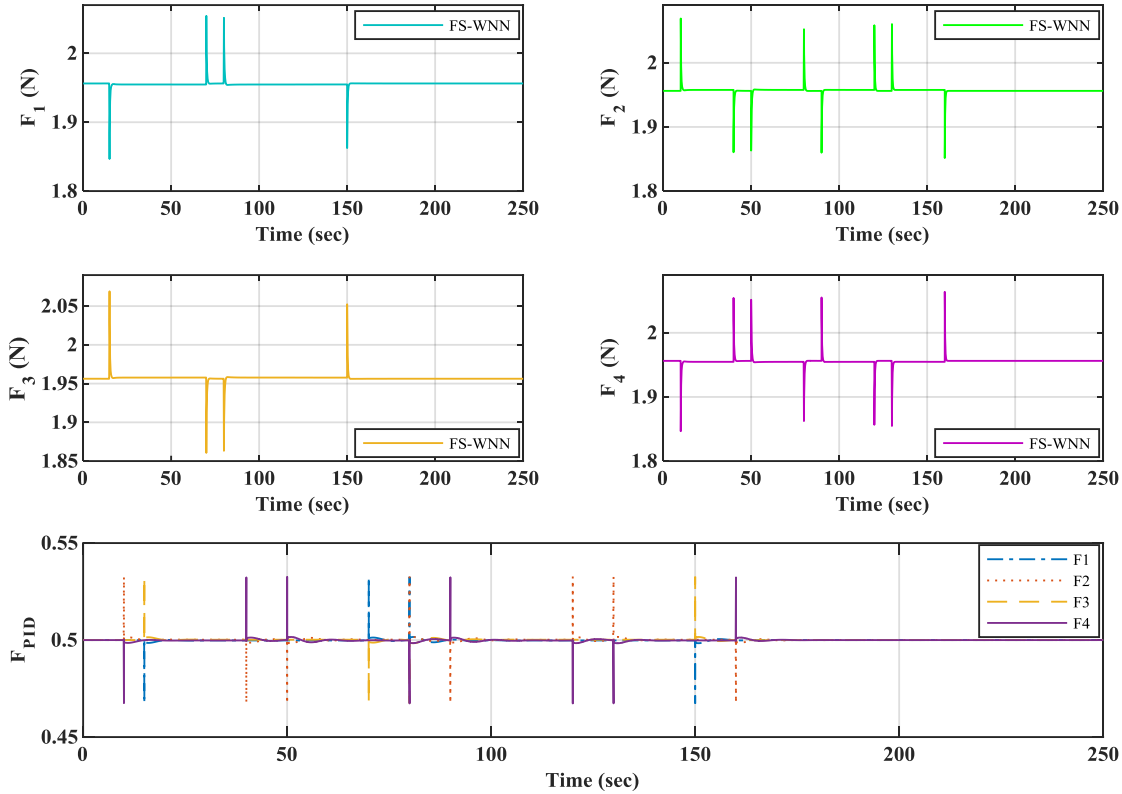


Figure 6.19: Forces de commande du contrôleur FS-WNN hybride et le contrôleur PID

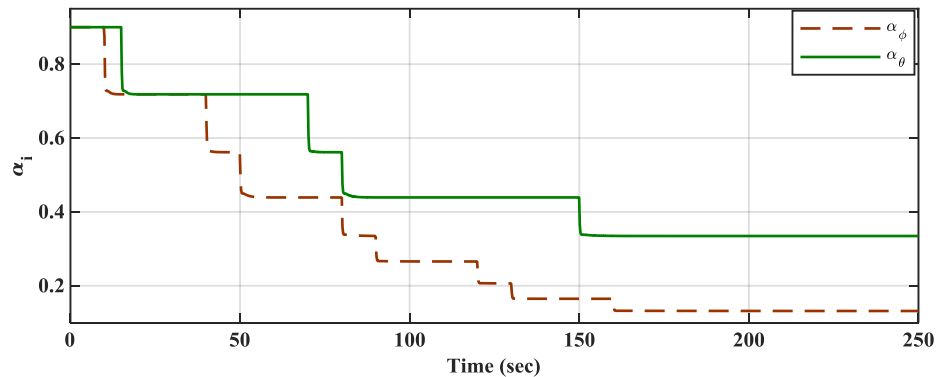


Figure 6.20: Paramètres de modulation du contrôleur FS-WNN hybride pour le mouvement de roulis et tangage pour $\hat{\alpha}(0) = 0.1$

La Figure 6.20 montre les paramètres de modulation du contrôleur FS-WNN hybride pour le mouvement de roulis et de tangage dans le cas où $\alpha=0.9$. On remarque que la valeur de α a

diminué jusqu'à 0.1 pour le roulis et 0.35 pour le tangage. Ce qui signifie que plus d'importance a été accordée au contrôleur WNN par le mécanisme d'adaptation selon les besoins de contrôle.

c) Cas du contrôleur VFS-WNN hybride-variable avec un paramètre de modulation $\hat{\alpha}$ adaptatif initialisé à 0.1

Dans cette partie, on utilise des paramètres de modulation adaptatifs et on les initialise par $\hat{\alpha} = 0.1$. Les performances de poursuite de roulis et tangage sont illustrées dans la Figure 6.21 et la Figure 6.22, respectivement, dans le cas du contrôleur PID et du contrôleur FS-WNN hybride-variable. La Figure 6.23 montre les performances de poursuite du lacet pour chaque contrôleur.

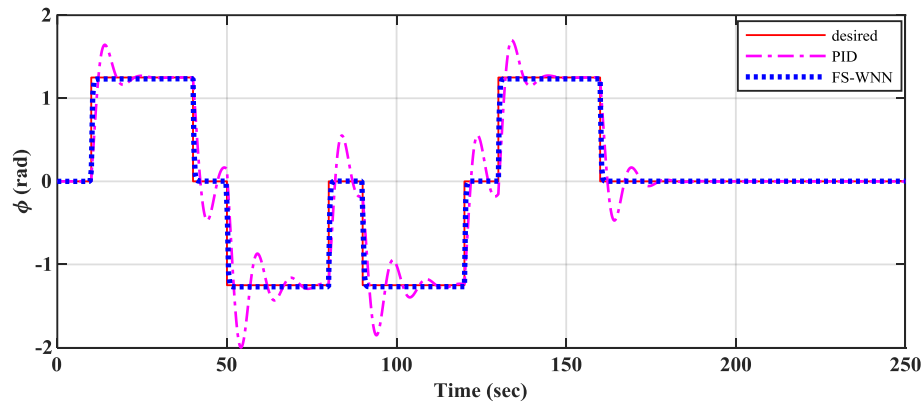


Figure 6.21: Performances de poursuite de roulis par le contrôleur FS-WNN hybride-variable et le PID pour $\hat{\alpha}(0) = 0.1$

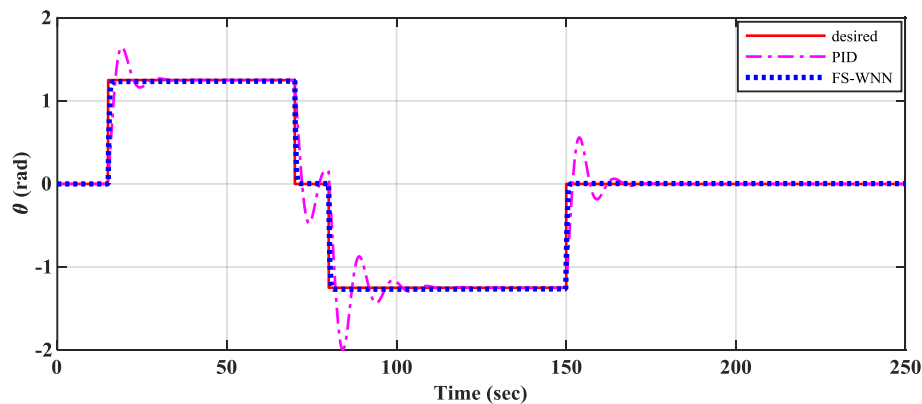


Figure 6.22: Performances de poursuite de tangage par le contrôleur FS-WNN hybride-variable et le PID pour $\hat{\alpha}(0) = 0.1$

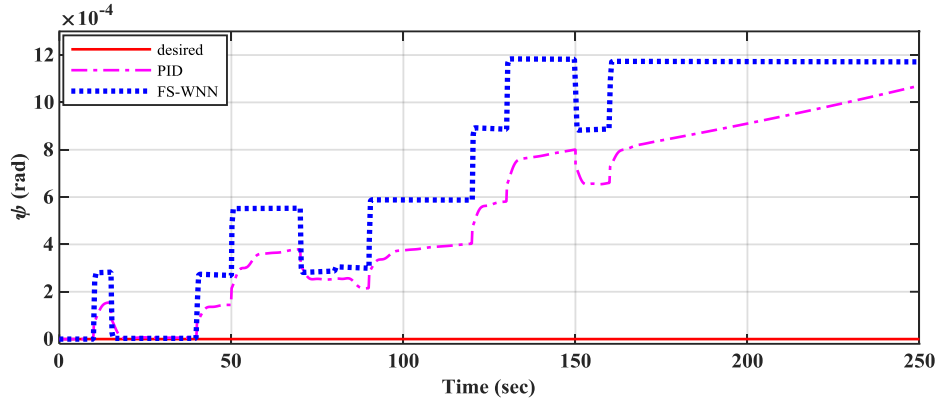


Figure 6.23 : Performances de poursuite de lacet par le contrôleur FS-WNN hybride-variable et le PID pour $\hat{\alpha}(0) = 0.1$

En observant les performances de poursuite du roulis, tangage et lacet par le contrôleur FS-WNN hybride-variable et le contrôleur PID, on trouve que le contrôleur FS-WNN hybride-variable a pu assurer une excellente poursuite des trajectoires désirées face au contrôleur PID pour les trois angles d'attitude dans le cas de $\hat{\alpha}(0) = 0.1$ comme dans le cas de $\hat{\alpha}(0) = 0.9$. Cela signifie que le contrôleur FS-WNN hybride-variable est capable d'assurer les performances souhaités en ajustant le paramètre de modulation α en ligne. Les forces générées par les moteurs de quadrotor sont illustrées dans Figure 6.25. Les trajectoires des paramètre de modulation α sont données dans la Figure 6.26. Une étude comparative selon l'erreur moyenne quadratique MSE entre le PID, le FS-WNN et le VFS-WNN est donnée dans le Tableau 6.2.

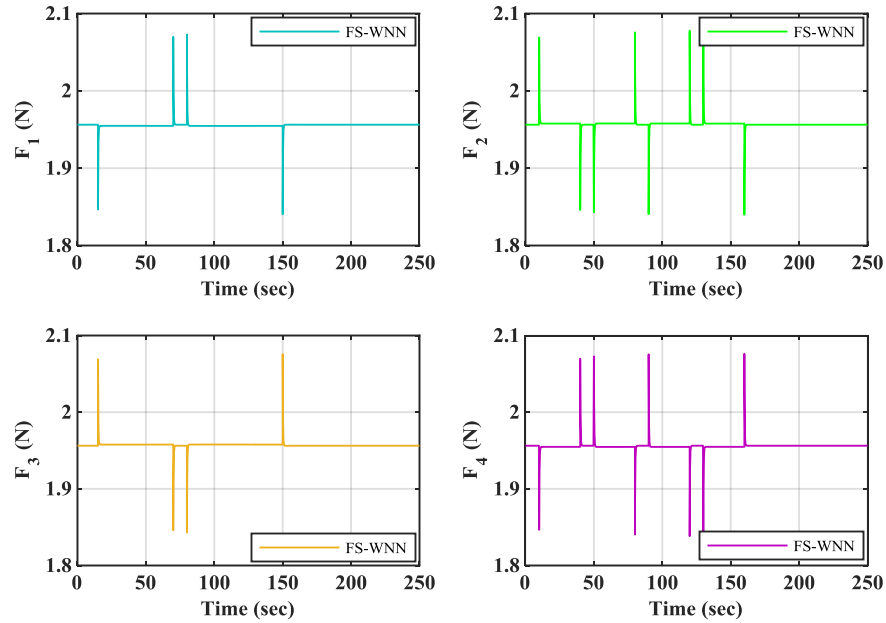


Figure 6.24: Forces de commande du contrôleur FS-WNN hybride-variable

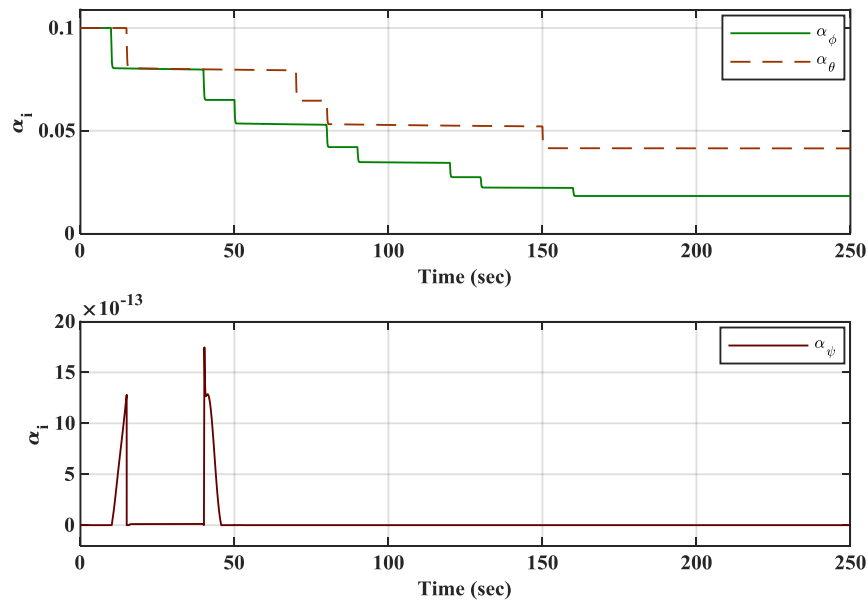


Figure 6.25: Paramètres de modulation du contrôleur FS-WNN hybride pour le mouvement de roulis et tangage (en haut) et pour le lacet (en bas)

A partir de ces Figures, on constate que le degré de contribution le plus élevé est attribué au contrôleur WNN. On remarque aussi que à chaque mouvement, le mécanisme d'adaptation intervient pour éliminer le dépassement en multipliant la réponse du WNN par un gain positif < 1 .

Tableau 6.2 : Comparaison de MSE de l'erreur de poursuite des trois contrôleurs (PID, FS-WNN, VFS-WNN)

	Critère de comparaison		
	$MSE(\varphi)$	$MSE(\theta)$	$MSE(\psi)$
PID	0.0298	0.0158	0.2108×10^{-6}
FS-WNN ($\alpha=0.5$)	0.0085	0.0055	0.8684×10^{-6}
VFS-WNN ($\alpha=0.1$)	0.0032	0.0016	0.3731×10^{-6}
VFS-WNN ($\alpha=0.9$)	0.0036	0.0018	0.2828×10^{-6}

La comparaison des performances des trois contrôleurs (PID, FS-WNN, VFS-WNN) en fonction de MSE de leurs erreurs de poursuite est donnée dans le Tableau 6.2, on remarque que le contrôleur VFS-WNN hybride-variable avec paramètre de modulation adaptatif $\alpha=0.1$ a assurée l'erreur de poursuite la plus petites entre les trois contrôleurs. Ensuite, on trouve dans la deuxième position le contrôleur VFS-WNN hybride-variable avec paramètre de modulation adaptatif $\alpha=0.9$. Dans la troisième position le VFS-WNN hybride avec un paramètre de modulation fixe $\alpha=0.5$. Et en dernier, on trouve le PID qui a présenté l'erreur de poursuite la plus élevée.

5. Conclusion

Dans ce chapitre, un contrôleur FS-WNN hybride-variable a été proposé pour le contrôle du quadrotor. En différence avec le contrôleur PID qui dépend du modèle de système, le contrôleur FS-WNN hybride-variable utilise le FS-WNN hybride-variable développé dans le chapitre 05 pour estimer directement et en ligne la loi de commande. Les lois d'adaptation des paramètres du FS-WNN hybride-variable ont été dérivées par la méthode de Lyapunov pour assurer la stabilité du système en boucle fermée. Le contrôleur FS-WNN hybride-variable proposé surmonte le problème de la connaissance préalable des dynamiques du modèle de système à commander. Le contrôleur FS-WNN hybride-variable a également une architecture simple et donc facile à concevoir et à mettre en œuvre dans des applications réelles. Une application par simulation de contrôle d'un quadrotor par le FS-WNN hybride-variable proposé et par un PID a été développée. Les résultats de simulation ont été présentés pour démontrer l'efficacité du contrôleur FS-WNN hybride-variable par rapport au contrôleur PID dans le contrôle des systèmes MIMO non linéaires et incertains. Le contrôleur FS-WNN hybride-variable a montré une excellente poursuite et un signal de commande

plus lisse que celui du contrôleur FS-WNN hybride avec des paramètres de modulation fixes. De même, le contrôleur FS-WNN hybride-variable n'a pas engendré des dépassements contrairement au contrôleur PID. On constate que le contrôleur FS-WNN hybride-variable est plus performant que le contrôleur FS-WNN hybride et le contrôleur PID.

Conclusion générale

Cette thèse a été consacrée à l'identification et la commande intelligente des systèmes non linéaires incertains. Nous avons présenté, dans le cadre de cette thèse, la technique de contrôle adaptatif intelligent des systèmes SISOs non linéaires incertains et l'importance d'un choix approprié de la fonction d'activation ondelette sur les performances du réseau WNN. Nous avons ainsi prouvé qu'il est possible d'améliorer la précision et la robustesse du contrôleur neuronal à base d'ondelettes tout en diminuant sa complexité (taille, paramètres à ajuster, ... etc.) avec un bon choix de la famille d'ondelettes dans la conception du contrôleur WNN. Ensuite, nous avons présenté la technique de contrôle adaptatif neuronal à base d'ondelettes des systèmes MIMO incertains. Où nous avons utilisé le réseau de neurones à base d'ondelettes dans la conception d'un contrôleur adaptatif neuronal indirect à base d'ondelettes pour la commande des systèmes MIMO non linéaires incertains. Les résultats de simulation sur un robot manipulateur ont montré une très bonne poursuite grâce à la grande capacité du réseau WNN à adapter en ligne les paramètres de translation et de dilatation des fonctions ondelettes dans les neurones cachés.

Dans le même contexte de contrôle intelligent des systèmes incertains, nous avons établi une étude détaillée de contrôle adaptatif indirect des systèmes MIMO incertains en utilisant les systèmes flous. Les systèmes flous ont des caractéristiques intéressantes dans l'approximation des fonctions non linéaires face aux incertitudes et l'imprécision. L'analyse des performances du contrôleur flou adaptatif indirect par simulation sur un robot manipulateur MIMO a montré des très bonnes performances de poursuite avec une grande robustesse et une insensibilité face aux variations paramétriques contrairement aux réseaux WNNs. La complémentarité des deux approches nous a motivé pour combiner les deux approches différentes dans un nouveau réseau FS-WNN hybride

Nous avons développé dans le cadre de cette thèse un nouveau approximateur FS-WNN hybride pour la commande des systèmes MIMO non linéaires incertains. L'approximateur FS-WNN hybride a été synthétisé par l'hybridation du système flou avec le réseau de neurones à base d'ondelettes en utilisant la technique de modulation comme un moyen d'hybridation. La technique de modulation utilisée nous a permis de fixer le degré de contribution de chaque système intelligent

selon un choix de concepteur. Les résultats de simulation ont prouvé que le contrôleur FS-WNN hybride adaptatif indirect est plus performant que le contrôleur WNN adaptatif indirect et le contrôleur FS adaptatif indirect. Ce qui peut être justifié comme suit : le contrôleur FS-WNN hybride a exploité la multiplicité des approximations différentes de la même fonction pour réduire l'erreur d'approximation, et a bénéficié de la grande capacité d'adaptation du réseau WNN et de la robustesse du système FS pour mieux contrôler le système incertain et cela même après l'application des variations paramétriques et des perturbations.

Cependant, pour résoudre le problème de choix des paramètres de modulation, nous avons développé un algorithme d'optimisation qui se charge de la recherche et de l'adaptation des paramètres optimaux de modulation en ligne. L'intérêt de cet algorithme est l'automatisation du choix des paramètres de modulation et l'ajustement en ligne de ces paramètres selon les besoins de contrôle. L'algorithme proposé offre un gain de temps et d'effort au concepteur et permet d'atteindre les capacités maximales de l'approximateur FS-WNN hybride-variable en comparaison avec l'approximateur FS-WNN hybride avec des paramètres de modulation fixes.

Par la suite, nous avons utilisé le nouveau approximateur FS-WNN hybride-variable dans le développement d'un nouveau schéma de commande adaptative directe pour contrôler l'attitude d'un quadrotor. Les quadrotors sont des systèmes MIMO incertains naturellement instables et représentent un vrai challenge au niveau de la modélisation et le contrôle due à leurs dynamiques complexes. Ce qui fait de l'approximateur FS-WNN hybride-variable un choix parfait comme un système intelligent pour le contrôle du quadrotor. Le schéma de commande proposé a prouvé son efficacité par rapport aux techniques de commande adaptatives qui se basent sur un seul approximateur intelligent dans le design du contrôleur en terme de précision et de robustesse.

Travaux futurs de recherche

Ce qui suit présente des recommandations pour de futures recherches et des perspectives envisageables :

1. Nous avons considéré, dans ce travail, la classe des systèmes non linéaires incertains continus. Dans d'autres applications, il sera nécessaire de reconsidérer le problème dans le cadre de la commande discrète; ce qui peut changer certaines étapes de conception, comme les lois d'ajustement.

2. Dans le cadre de ce travail, on a toujours supposé que le vecteur d'états est intégralement disponible à la mesure. Il est plus judicieux de considérer la synthèse de la loi de commande avec un observateur d'état pour estimer le vecteur des états du système.
3. Les approches développées ont montré leur efficacité en simulation. Des tests de validation sur des systèmes robotiques réels sont fortement recommandés.
4. On a considéré dans notre travail l'hybridation en utilisant la technique de modulation adaptative. Cependant, une estimation des paramètres de modulation en utilisant d'autres techniques d'optimisation des paramètres de modulation comme les algorithmes génétiques est aussi envisageable.

Annexe A

Linéarisation de Taylor :

Cas de $F(x)$:

On a :

$$\tilde{\Psi}_F(\mathbf{x}, \tilde{\mathbf{C}}_F) = \mathbf{\Omega}_F^T \tilde{\mathbf{C}}_F + O_F \quad (\text{A.1})$$

Avec :

$$\mathbf{\Omega}_F = \left. \frac{\partial \psi_F}{\partial \mathbf{C}_F} \right|_{\mathbf{C}_F = \tilde{\mathbf{C}}_F} = \left[\left. \frac{\partial \psi_1}{\partial \mathbf{C}_F} \frac{\partial \psi_2}{\partial \mathbf{C}_F} \dots \frac{\partial \psi_{n_c}}{\partial \mathbf{C}_F} \right]^T \right|_{\mathbf{C}_F = \tilde{\mathbf{C}}_F} \in R^{n_c \times n_c} \quad (\text{A.2})$$

Avec:

$$\psi_F = \begin{bmatrix} \psi_F^1 \\ \vdots \\ \psi_F^{n_c} \end{bmatrix}_{n_c \times 1} \quad (\text{A.3})$$

Et:

$$\mathbf{C}_F = \begin{bmatrix} C_F^{11} & \dots & C_F^{1n} \\ \vdots & \ddots & \vdots \\ C_F^{n_c 1} & \dots & C_F^{n_c n} \end{bmatrix}_{n_c \times n} \quad (\text{A.4})$$

Avec n_c est le nombre des fonctions d'activations dans le réseau, et n est le nombre des variables d'entrée du réseau. La dérivée partielle de (A.1) donne :

$$\Omega_F = \frac{\partial \psi_F}{\partial C_F} = \frac{\partial \begin{bmatrix} \psi_F^1 \\ \vdots \\ \psi_F^{n_c} \end{bmatrix}_{n_c \times 1}}{\partial \begin{bmatrix} c_F^{11} & \dots & c_F^{1n} \\ \vdots & \ddots & \vdots \\ c_F^{n_c 1} & \dots & c_F^{n_c n} \end{bmatrix}_{n_c \times n}} = \frac{\partial \begin{bmatrix} \psi_F^1 \\ \vdots \\ \psi_F^{n_c} \end{bmatrix}_{n_c \times 1}}{\partial \begin{bmatrix} c_F^{11} \\ \vdots \\ c_F^{n_c n} \end{bmatrix}_{(n_c \cdot n) \times 1}} = \begin{bmatrix} \frac{\partial \psi_F^1}{c_F^{11}} & \dots & \frac{\partial \psi_F^1}{c_F^{1n}} \\ \vdots & \ddots & \vdots \\ \frac{\partial \psi_F^{n_c}}{c_F^{n_c 1}} & \dots & \frac{\partial \psi_F^{n_c}}{c_F^{n_c n}} \end{bmatrix}_{(n_c \cdot n) \times n_c} \quad (\text{A.5})$$

$$\Omega_F = \begin{bmatrix} \frac{\partial \psi_F^1}{\partial c_F^{11}} & 0 & 0 & \dots \\ \frac{\partial \psi_F^1}{\partial c_F^{12}} & 0 & 0 & \dots \\ \vdots & 0 & 0 & \dots \\ \frac{\partial \psi_F^1}{\partial c_F^{1n}} & 0 & 0 & \dots \\ 0 & \frac{\partial \psi_F^2}{\partial c_F^{21}} & 0 & \vdots \\ 0 & \frac{\partial \psi_F^2}{\partial c_F^{22}} & 0 & \vdots \\ \vdots & \vdots & \vdots & \vdots \end{bmatrix}_{(n_c \cdot n) \times n_c} \quad (\text{A.6})$$

Alors, on aura :

$$\Omega_F^T = \begin{bmatrix} \frac{\partial \psi_F^1}{\partial c_F^{11}} & \frac{\partial \psi_F^1}{\partial c_F^{12}} & \dots & \frac{\partial \psi_F^1}{\partial c_F^{1n}} & 0 & 0 & \dots \\ 0 & 0 & 0 & 0 & \frac{\partial \psi_F^2}{\partial c_F^{21}} & \frac{\partial \psi_F^2}{\partial c_F^{22}} & \dots \\ 0 & 0 & 0 & 0 & 0 & 0 & \dots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix}_{n_c \times (n_c \cdot n)} \quad (\text{A.7})$$

Sachant que :

$$\hat{C}_F = \begin{bmatrix} c_F^{11} \\ \vdots \\ c_F^{n_c n} \end{bmatrix}_{(n_c \cdot n) \times 1} \quad (\text{A.8})$$

De même pour $G(x)$:

$$\tilde{\Psi}_G(x, \tilde{C}_G) = \Omega_G^T \cdot \tilde{C}_G + O_G \quad (\text{A.9})$$

$$\mathbf{\Omega}_G^T = \left. \frac{\partial \psi_G}{\partial \mathbf{C}_G} \right|_{\mathbf{C}_G = \hat{\mathbf{C}}_G} = \left[\frac{\partial \psi_1}{\partial \mathbf{C}_G} \quad \frac{\partial \psi_2}{\partial \mathbf{C}_G} \quad \dots \quad \frac{\partial \psi_{n_c}}{\partial \mathbf{C}_G} \right]^T \bigg|_{\mathbf{C}_G = \hat{\mathbf{C}}_G} \in R^{(2.n_c) \times (2.n_c.n)} \quad (\text{A.10})$$

On a :

$$\psi_G = \begin{bmatrix} \psi_{G^{11}}^1 & 0 \\ \vdots & \vdots \\ \psi_{G^{11}}^{n_c} & 0 \\ 0 & \psi_{G^{22}}^1 \\ \vdots & \vdots \\ 0 & \psi_{G^{22}}^{n_c} \end{bmatrix}_{(2.n_c) \times 2} \quad (\text{A.11})$$

Et on a :

$$\mathbf{C}_{G^{11}} = \begin{bmatrix} C_{G^{11}}^{11} & \dots & C_{G^{11}}^{1n} \\ \vdots & & \vdots \\ C_{G^{11}}^{n_c 1} & \dots & C_{G^{11}}^{n_c n} \end{bmatrix}_{n_c \times n}, \quad \mathbf{C}_{G^{12}} = \begin{bmatrix} C_{G^{12}}^{11} & \dots & C_{G^{12}}^{1n} \\ \vdots & & \vdots \\ C_{G^{12}}^{n_c 1} & \dots & C_{G^{12}}^{n_c n} \end{bmatrix}_{n_c \times n} \quad (\text{A.12})$$

$$\mathbf{C}_{G^{21}} = \begin{bmatrix} C_{G^{21}}^{11} & \dots & C_{G^{21}}^{1n} \\ \vdots & & \vdots \\ C_{G^{21}}^{n_c 1} & \dots & C_{G^{21}}^{n_c n} \end{bmatrix}_{n_c \times n}, \quad \mathbf{C}_{G^{22}} = \begin{bmatrix} C_{G^{22}}^{11} & \dots & C_{G^{22}}^{1n} \\ \vdots & & \vdots \\ C_{G^{22}}^{n_c 1} & \dots & C_{G^{22}}^{n_c n} \end{bmatrix}_{n_c \times n}$$

En regroupant les matrices $\mathbf{C}_{G^{11}}$, $\mathbf{C}_{G^{12}}$, $\mathbf{C}_{G^{21}}$, $\mathbf{C}_{G^{22}}$ dans une seule matrice \mathbf{C}_g , on aura :

$$\mathbf{C}_g = \begin{bmatrix} C_{G^{11}}^{11} & 0 \\ \vdots & \vdots \\ C_{G^{11}}^{n_c n} & 0 \\ 0 & C_{G^{22}}^{11} \\ \vdots & \vdots \\ 0 & C_{G^{22}}^{n_c n} \end{bmatrix}_{(2.n_c.n) \times 2} \quad (\text{A.13})$$

On calcule la matrice des dérivées partielles de $\psi_{\mathbf{G}^{11}}$ par rapport à $\mathbf{C}_{\mathbf{G}^{11}}$, on obtient :

$$\mathbf{\Omega}_{\mathbf{G}^{11}} = \frac{\partial \psi_{\mathbf{G}^{11}}}{\partial \mathbf{C}_{\mathbf{G}^{11}}} = \frac{\partial \begin{bmatrix} \psi_{\mathbf{G}^{11}}^1 \\ \vdots \\ \psi_{\mathbf{G}^{11}}^{n_c} \end{bmatrix}_{n_c \times 1}}{\partial \begin{bmatrix} c_{\mathbf{G}^{11}}^{11} & \dots & c_{\mathbf{G}^{11}}^{1n} \\ \vdots & & \vdots \\ c_{\mathbf{G}^{11}}^{n_c 1} & \dots & c_{\mathbf{G}^{11}}^{n_c n} \end{bmatrix}_{n_c \times n}} = \frac{\partial \begin{bmatrix} \psi_{\mathbf{G}^{11}}^1 \\ \vdots \\ \psi_{\mathbf{G}^{11}}^{n_c} \end{bmatrix}_{n_c \times 1}}{\partial \begin{bmatrix} c_{\mathbf{G}^{11}}^{11} \\ \vdots \\ c_{\mathbf{G}^{11}}^{n_c n} \end{bmatrix}_{(n_c \cdot n) \times 1}} = \begin{bmatrix} \frac{\partial \psi_{\mathbf{G}^{11}}^1}{c_{\mathbf{G}^{11}}^{11}} & \dots & \frac{\partial \psi_{\mathbf{G}^{11}}^{n_c}}{c_{\mathbf{G}^{11}}^{n_c n}} \end{bmatrix}_{(n_c \cdot n) \times n_c} \quad (\text{A.14})$$

$$\mathbf{\Omega}_{\mathbf{G}^{11}} = \begin{bmatrix} \frac{\partial \psi_{\mathbf{G}^{11}}^1}{\partial c_{\mathbf{G}^{11}}^{11}} & 0 & 0 & \dots \\ \frac{\partial \psi_{\mathbf{G}^{11}}^1}{\partial c_{\mathbf{G}^{11}}^{12}} & 0 & 0 & \dots \\ \vdots & 0 & 0 & \dots \\ \frac{\partial \psi_{\mathbf{G}^{11}}^1}{\partial c_{\mathbf{G}^{11}}^{1n}} & 0 & 0 & \dots \\ 0 & \frac{\partial \psi_{\mathbf{G}^{11}}^2}{\partial c_{\mathbf{G}^{11}}^{21}} & 0 & \vdots \\ 0 & \frac{\partial \psi_{\mathbf{G}^{11}}^2}{\partial c_{\mathbf{G}^{11}}^{22}} & 0 & \vdots \\ \vdots & \vdots & \vdots & \vdots \end{bmatrix}_{(n_c \cdot n) \times n_c} \quad (\text{A.15})$$

On calcule la matrice des dérivées partielles de $\psi_{\mathbf{G}^{12}}$ par rapport à $\mathbf{C}_{\mathbf{G}^{12}}$, on obtient :

$$\mathbf{\Omega}_{\mathbf{G}^{12}} = \frac{\partial \psi_{\mathbf{G}^{12}}}{\partial \mathbf{C}_{\mathbf{G}^{12}}} = \frac{\partial \begin{bmatrix} \psi_{\mathbf{G}^{12}}^1 \\ \vdots \\ \psi_{\mathbf{G}^{12}}^{n_c} \end{bmatrix}_{n_c \times 1}}{\partial \begin{bmatrix} c_{\mathbf{G}^{12}}^{11} & \dots & c_{\mathbf{G}^{12}}^{1n} \\ \vdots & & \vdots \\ c_{\mathbf{G}^{12}}^{n_c 1} & \dots & c_{\mathbf{G}^{12}}^{n_c n} \end{bmatrix}_{n_c \times n}} = \frac{\partial \begin{bmatrix} \psi_{\mathbf{G}^{12}}^1 \\ \vdots \\ \psi_{\mathbf{G}^{12}}^{n_c} \end{bmatrix}_{n_c \times 1}}{\partial \begin{bmatrix} c_{\mathbf{G}^{12}}^{11} \\ \vdots \\ c_{\mathbf{G}^{12}}^{n_c n} \end{bmatrix}_{(n_c \cdot n) \times 1}} = \begin{bmatrix} \frac{\partial \psi_{\mathbf{G}^{12}}^1}{c_{\mathbf{G}^{12}}^{11}} & \dots & \frac{\partial \psi_{\mathbf{G}^{12}}^{n_c}}{c_{\mathbf{G}^{12}}^{n_c n}} \end{bmatrix}_{(n_c \cdot n) \times n_c} \quad (\text{A.16})$$

$$\Omega_{\mathbf{G}^{12}} = \begin{bmatrix} \frac{\partial \psi_{\mathbf{G}^{12}}^1}{\partial c_{\mathbf{G}^{12}}^{11}} & 0 & 0 & \dots \\ \frac{\partial \psi_{\mathbf{G}^{12}}^1}{\partial c_{\mathbf{G}^{12}}^{12}} & 0 & 0 & \dots \\ \vdots & 0 & 0 & \dots \\ \frac{\partial \psi_{\mathbf{G}^{12}}^1}{\partial c_{\mathbf{G}^{12}}^{1n}} & 0 & 0 & \dots \\ 0 & \frac{\partial \psi_{\mathbf{G}^{12}}^2}{\partial c_{\mathbf{G}^{12}}^{21}} & 0 & \vdots \\ 0 & \frac{\partial \psi_{\mathbf{G}^{12}}^2}{\partial c_{\mathbf{G}^{12}}^{22}} & 0 & \vdots \\ \vdots & \vdots & \vdots & \vdots \end{bmatrix}_{(n_c \cdot n) \times n_c} \quad (\text{A.17})$$

On calcule la matrice des dérivées partielles de $\psi_{\mathbf{G}^{21}}$ par rapport à $\mathbf{C}_{\mathbf{G}^{21}}$, on obtient :

$$\Omega_{\mathbf{G}^{21}} = \frac{\partial \psi_{\mathbf{G}^{21}}}{\partial \mathbf{C}_{\mathbf{G}^{21}}} = \frac{\partial \begin{bmatrix} \psi_{\mathbf{G}^{21}}^1 \\ \vdots \\ \psi_{\mathbf{G}^{21}}^{n_c} \end{bmatrix}_{n_c \times 1}}{\partial \begin{bmatrix} c_{\mathbf{G}^{21}}^{11} & \dots & c_{\mathbf{G}^{21}}^{1n} \\ \vdots & & \vdots \\ c_{\mathbf{G}^{21}}^{n_c 1} & \dots & c_{\mathbf{G}^{21}}^{n_c n} \end{bmatrix}_{n_c \times n}} = \frac{\partial \begin{bmatrix} \psi_{\mathbf{G}^{21}}^1 \\ \vdots \\ \psi_{\mathbf{G}^{21}}^{n_c} \end{bmatrix}_{n_c \times 1}}{\partial \begin{bmatrix} c_{\mathbf{G}^{21}}^{11} \\ \vdots \\ c_{\mathbf{G}^{21}}^{n_c n} \end{bmatrix}_{(n_c \cdot n) \times 1}} = \begin{bmatrix} \frac{\partial \psi_{\mathbf{G}^{21}}^1}{c_{\mathbf{G}^{21}}^{11}} \\ \vdots \\ \frac{\partial \psi_{\mathbf{G}^{21}}^{n_c}}{c_{\mathbf{G}^{21}}^{n_c n}} \end{bmatrix} \dots \begin{bmatrix} \frac{\partial \psi_{\mathbf{G}^{21}}^{n_c}}{c_{\mathbf{G}^{21}}^{11}} \\ \vdots \\ \frac{\partial \psi_{\mathbf{G}^{21}}^{n_c}}{c_{\mathbf{G}^{21}}^{n_c n}} \end{bmatrix}_{(n_c \cdot n) \times n_c} \quad (\text{A.18})$$

$$\Omega_{\mathbf{G}^{21}} = \begin{bmatrix} \frac{\partial \psi_{\mathbf{G}^{21}}^1}{\partial c_{\mathbf{G}^{21}}^{11}} & 0 & 0 & \dots \\ \frac{\partial \psi_{\mathbf{G}^{21}}^1}{\partial c_{\mathbf{G}^{21}}^{12}} & 0 & 0 & \dots \\ \vdots & 0 & 0 & \dots \\ \frac{\partial \psi_{\mathbf{G}^{21}}^1}{\partial c_{\mathbf{G}^{21}}^{1n}} & 0 & 0 & \dots \\ 0 & \frac{\partial \psi_{\mathbf{G}^{21}}^2}{\partial c_{\mathbf{G}^{21}}^{21}} & 0 & \vdots \\ 0 & \frac{\partial \psi_{\mathbf{G}^{21}}^2}{\partial c_{\mathbf{G}^{21}}^{22}} & 0 & \vdots \\ \vdots & \vdots & \vdots & \vdots \end{bmatrix}_{(n_c \cdot n) \times n_c} \quad (\text{A.19})$$

On calcule la matrice des dérivées partielles de $\psi_{\mathbf{G}^{22}}$ par rapport à $\mathbf{C}_{\mathbf{G}^{22}}$, on obtient :

$$\mathbf{\Omega}_{\mathbf{G}^{22}} = \frac{\partial \psi_{\mathbf{G}^{22}}}{\partial \mathbf{C}_{\mathbf{G}^{22}}} = \frac{\partial \begin{bmatrix} \psi_{\mathbf{G}^{22}}^1 \\ \vdots \\ \psi_{\mathbf{G}^{22}}^{n_c} \end{bmatrix}_{n_c \times 1}}{\partial \begin{bmatrix} c_{\mathbf{G}^{22}}^{11} & \dots & c_{\mathbf{G}^{22}}^{1n} \\ \vdots & & \vdots \\ c_{\mathbf{G}^{22}}^{n_c 1} & \dots & c_{\mathbf{G}^{22}}^{n_c n} \end{bmatrix}_{n_c \times n}} = \frac{\partial \begin{bmatrix} \psi_{\mathbf{G}^{22}}^1 \\ \vdots \\ \psi_{\mathbf{G}^{22}}^{n_c} \end{bmatrix}_{n_c \times 1}}{\partial \begin{bmatrix} c_{\mathbf{G}^{22}}^{11} \\ \vdots \\ c_{\mathbf{G}^{22}}^{n_c n} \end{bmatrix}_{(n_c \cdot n) \times 1}} = \begin{bmatrix} \frac{\partial \psi_{\mathbf{G}^{22}}^1}{\partial c_{\mathbf{G}^{22}}^{11}} & \dots & \frac{\partial \psi_{\mathbf{G}^{22}}^{n_c}}{\partial c_{\mathbf{G}^{22}}^{11}} \\ \vdots & & \vdots \\ \frac{\partial \psi_{\mathbf{G}^{22}}^1}{\partial c_{\mathbf{G}^{22}}^{n_c n}} & \dots & \frac{\partial \psi_{\mathbf{G}^{22}}^{n_c}}{\partial c_{\mathbf{G}^{22}}^{n_c n}} \end{bmatrix}_{(n_c \cdot n) \times n_c} \quad (\text{A.20})$$

$$\mathbf{\Omega}_{\mathbf{G}^{22}} = \begin{bmatrix} \frac{\partial \psi_{\mathbf{G}^{22}}^1}{\partial c_{\mathbf{G}^{22}}^{11}} & 0 & 0 & \dots \\ \frac{\partial \psi_{\mathbf{G}^{22}}^1}{\partial c_{\mathbf{G}^{22}}^{12}} & 0 & 0 & \dots \\ \vdots & 0 & 0 & \dots \\ \frac{\partial \psi_{\mathbf{G}^{22}}^1}{\partial c_{\mathbf{G}^{22}}^{1n}} & 0 & 0 & \dots \\ 0 & \frac{\partial \psi_{\mathbf{G}^{22}}^2}{\partial c_{\mathbf{G}^{22}}^{21}} & 0 & \vdots \\ 0 & \frac{\partial \psi_{\mathbf{G}^{22}}^2}{\partial c_{\mathbf{G}^{22}}^{22}} & 0 & \vdots \\ \vdots & \vdots & \vdots & \vdots \end{bmatrix}_{(n_c \cdot n) \times n_c} \quad (\text{A.21})$$

En regroupant les matrices $\mathbf{\Omega}_{\mathbf{G}^{11}}$, $\mathbf{\Omega}_{\mathbf{G}^{12}}$, $\mathbf{\Omega}_{\mathbf{G}^{21}}$, $\mathbf{\Omega}_{\mathbf{G}^{22}}$ dans une seule matrice $\mathbf{\Omega}_{\mathbf{G}}$, on aura :

$$\mathbf{\Omega}_{\mathbf{G}} = \begin{bmatrix} \frac{\partial \psi_{\mathbf{G}^{11}}^1}{\partial c_{\mathbf{G}^{11}}^{11}} & \frac{\partial \psi_{\mathbf{G}^{11}}^2}{\partial c_{\mathbf{G}^{11}}^{11}} & \dots & \frac{\partial \psi_{\mathbf{G}^{11}}^{n_c}}{\partial c_{\mathbf{G}^{11}}^{11}} & \frac{\partial \psi_{\mathbf{G}^{12}}^1}{\partial c_{\mathbf{G}^{12}}^{11}} & \frac{\partial \psi_{\mathbf{G}^{12}}^2}{\partial c_{\mathbf{G}^{12}}^{11}} & \dots & \frac{\partial \psi_{\mathbf{G}^{12}}^{n_c}}{\partial c_{\mathbf{G}^{12}}^{11}} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \frac{\partial \psi_{\mathbf{G}^{11}}^1}{\partial c_{\mathbf{G}^{11}}^{n_c n}} & \frac{\partial \psi_{\mathbf{G}^{11}}^2}{\partial c_{\mathbf{G}^{11}}^{n_c n}} & \dots & \frac{\partial \psi_{\mathbf{G}^{11}}^{n_c}}{\partial c_{\mathbf{G}^{11}}^{n_c n}} & \frac{\partial \psi_{\mathbf{G}^{12}}^1}{\partial c_{\mathbf{G}^{12}}^{n_c n}} & \frac{\partial \psi_{\mathbf{G}^{12}}^2}{\partial c_{\mathbf{G}^{12}}^{n_c n}} & \dots & \frac{\partial \psi_{\mathbf{G}^{12}}^{n_c}}{\partial c_{\mathbf{G}^{12}}^{n_c n}} \\ \frac{\partial \psi_{\mathbf{G}^{21}}^1}{\partial c_{\mathbf{G}^{21}}^{11}} & \frac{\partial \psi_{\mathbf{G}^{21}}^2}{\partial c_{\mathbf{G}^{21}}^{11}} & \dots & \frac{\partial \psi_{\mathbf{G}^{21}}^{n_c}}{\partial c_{\mathbf{G}^{21}}^{11}} & \frac{\partial \psi_{\mathbf{G}^{22}}^1}{\partial c_{\mathbf{G}^{22}}^{11}} & \frac{\partial \psi_{\mathbf{G}^{22}}^2}{\partial c_{\mathbf{G}^{22}}^{11}} & \dots & \frac{\partial \psi_{\mathbf{G}^{22}}^{n_c}}{\partial c_{\mathbf{G}^{22}}^{11}} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \frac{\partial \psi_{\mathbf{G}^{21}}^1}{\partial c_{\mathbf{G}^{21}}^{n_c n}} & \frac{\partial \psi_{\mathbf{G}^{21}}^2}{\partial c_{\mathbf{G}^{21}}^{n_c n}} & \dots & \frac{\partial \psi_{\mathbf{G}^{21}}^{n_c}}{\partial c_{\mathbf{G}^{21}}^{n_c n}} & \frac{\partial \psi_{\mathbf{G}^{22}}^1}{\partial c_{\mathbf{G}^{22}}^{n_c n}} & \frac{\partial \psi_{\mathbf{G}^{22}}^2}{\partial c_{\mathbf{G}^{22}}^{n_c n}} & \dots & \frac{\partial \psi_{\mathbf{G}^{22}}^{n_c}}{\partial c_{\mathbf{G}^{22}}^{n_c n}} \end{bmatrix}_{(2 \cdot n_c \cdot n) \times (2 \cdot n_c)} \quad (\text{A.22})$$

En calculant le transpose de Ω_g , on obtient :

$$\Omega_G^T = \begin{bmatrix} \frac{\partial \psi_{G^{11}}^1}{\partial c_{G^{11}}^{11}} & \dots & \frac{\partial \psi_{G^{11}}^1}{\partial c_{G^{11}}^{n_c n}} & \frac{\partial \psi_{G^{12}}^1}{\partial c_{G^{12}}^{11}} & \dots & \frac{\partial \psi_{G^{12}}^1}{\partial c_{G^{12}}^{n_c n}} \\ \frac{\partial \psi_{G^{11}}^2}{\partial c_{G^{11}}^{11}} & \dots & \frac{\partial \psi_{G^{11}}^2}{\partial c_{G^{11}}^{n_c n}} & \frac{\partial \psi_{G^{12}}^2}{\partial c_{G^{12}}^{11}} & \dots & \frac{\partial \psi_{G^{12}}^2}{\partial c_{G^{12}}^{n_c n}} \\ \vdots & & \vdots & \vdots & & \vdots \\ \frac{\partial \psi_{G^{11}}^{n_c}}{\partial c_{G^{11}}^{11}} & \dots & \frac{\partial \psi_{G^{11}}^{n_c}}{\partial c_{G^{11}}^{n_c n}} & \frac{\partial \psi_{G^{12}}^{n_c}}{\partial c_{G^{12}}^{11}} & \dots & \frac{\partial \psi_{G^{12}}^{n_c}}{\partial c_{G^{12}}^{n_c n}} \\ \frac{\partial \psi_{G^{21}}^1}{\partial c_{G^{21}}^{11}} & \dots & \frac{\partial \psi_{G^{21}}^1}{\partial c_{G^{21}}^{n_c n}} & \frac{\partial \psi_{G^{22}}^1}{\partial c_{G^{22}}^{11}} & \dots & \frac{\partial \psi_{G^{22}}^1}{\partial c_{G^{22}}^{n_c n}} \\ \frac{\partial \psi_{G^{21}}^2}{\partial c_{G^{21}}^{11}} & \dots & \frac{\partial \psi_{G^{21}}^2}{\partial c_{G^{21}}^{n_c n}} & \frac{\partial \psi_{G^{22}}^2}{\partial c_{G^{22}}^{11}} & \dots & \frac{\partial \psi_{G^{22}}^2}{\partial c_{G^{22}}^{n_c n}} \\ \vdots & & \vdots & \vdots & & \vdots \\ \frac{\partial \psi_{G^{21}}^{n_c}}{\partial c_{G^{21}}^{11}} & \dots & \frac{\partial \psi_{G^{21}}^{n_c}}{\partial c_{G^{21}}^{n_c n}} & \frac{\partial \psi_{G^{22}}^{n_c}}{\partial c_{G^{22}}^{11}} & \dots & \frac{\partial \psi_{G^{22}}^{n_c}}{\partial c_{G^{22}}^{n_c n}} \end{bmatrix}_{(2.n_c) \times (2.n_c.n)}$$

Sachant que :

$$\hat{C}_G = \begin{bmatrix} c_{G^{11}}^{11} & 0 \\ \vdots & \vdots \\ c_{G^{11}}^{n_c n} & 0 \\ 0 & c_{G^{22}}^{11} \\ \vdots & \vdots \\ 0 & c_{G^{22}}^{n_c n} \end{bmatrix}_{(2.n_c.n) \times 2}$$

Annexe B

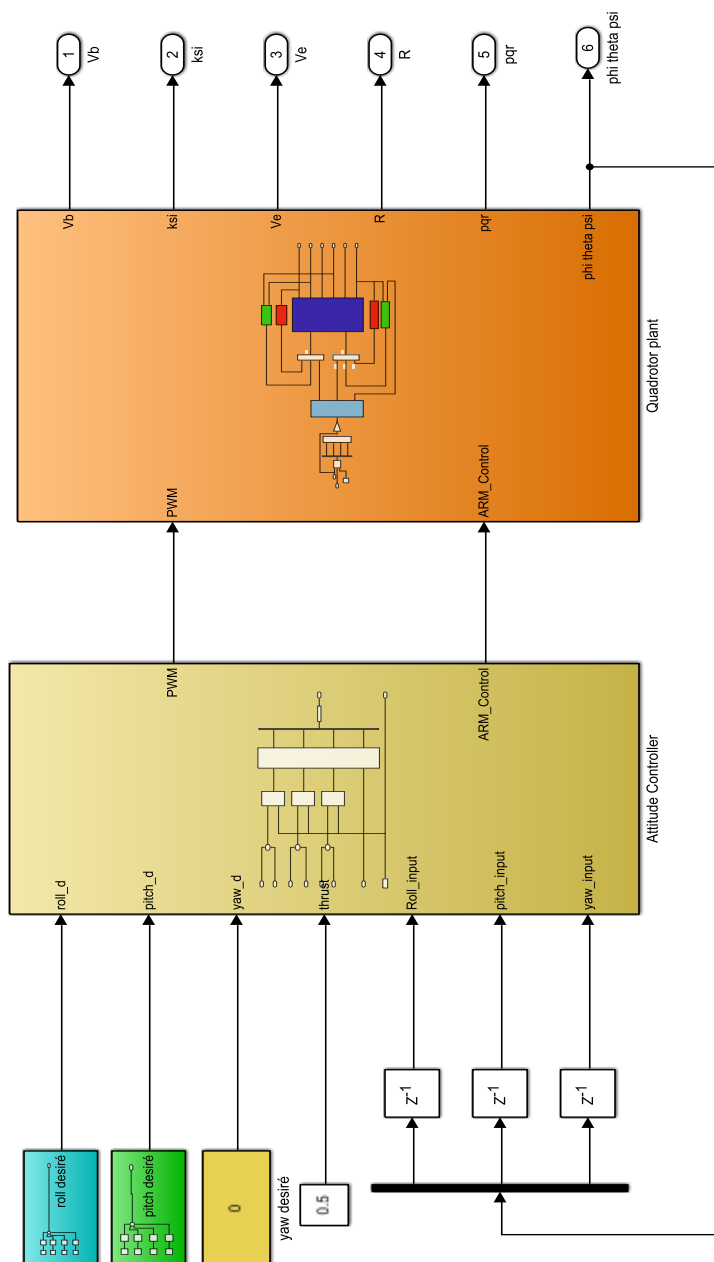


Figure 6.26: Schéma de commande de l'attitude de quadrotor dans le Toolbox Matlab Px4 Pixhawk

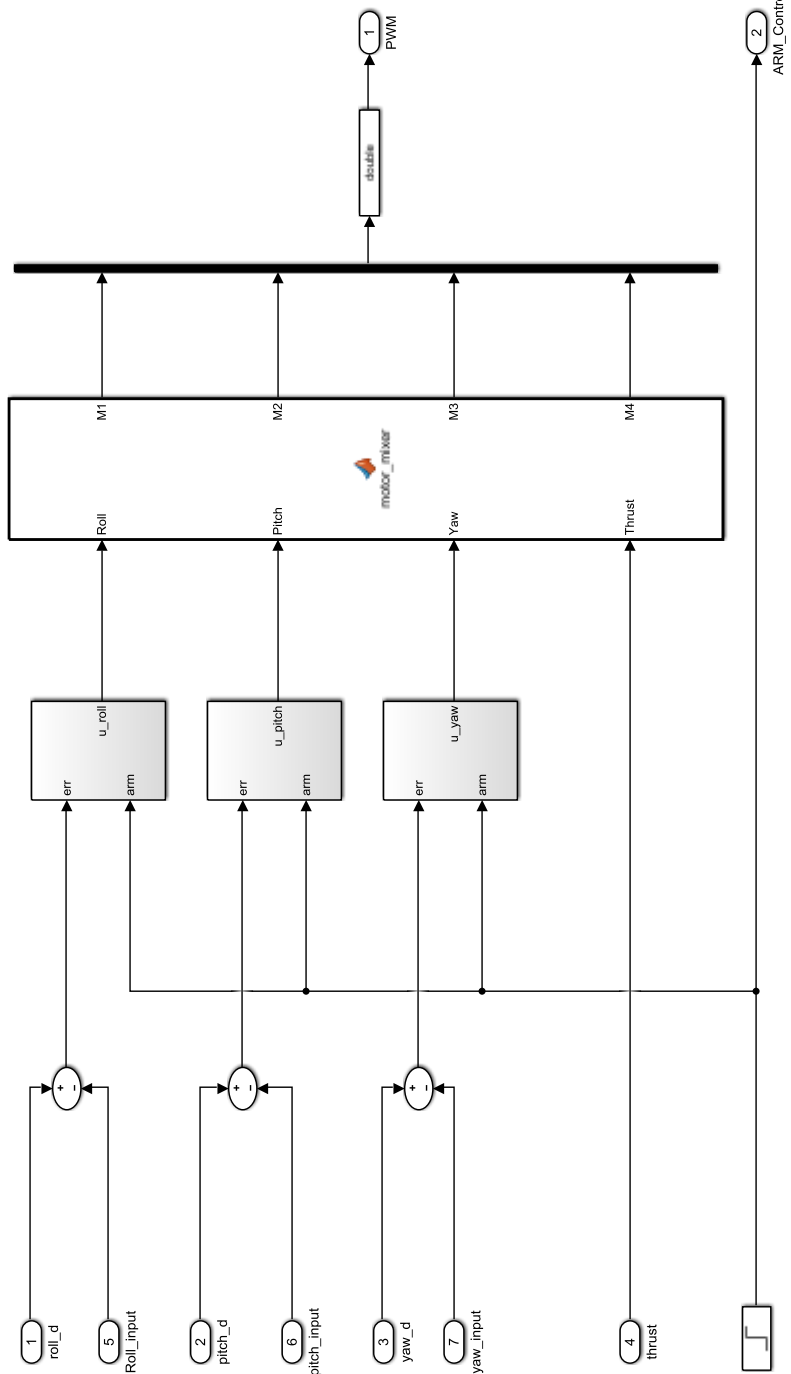


Figure 6.27: Schéma de contrôleur de l'attitude de quadrotor dans le Toolbox Matlab Px4 Pixhawk

Biblio-webographie

- [1] H. Czichos, « Technology in the 21st Century – The role of materials and materials testing », *Mater. Test.*, vol. 63, n° 1, p. 5-9, janv. 2021, doi: 10.1515/mt-2020-0015.
- [2] X. Guan, Z. Zhong, R. Liu, et J. Zhan, « Virtual subjective and objective evaluation method of traffic jam assist », *Proc. Inst. Mech. Eng. Part J. Automob. Eng.*, 2022, doi: 10.1177/09544070221078398.
- [3] H. Shraim, A. Awada, et R. Youness, « A survey on quadrotors: Configurations, modeling and identification, control, collision avoidance, fault diagnosis and tolerant control », *IEEE Aerosp. Electron. Syst. Mag.*, vol. 33, n° 7, p. 14-33, juill. 2018, doi: 10.1109/MAES.2018.160246.
- [4] E. Spyrakos-Papastavridis et J. S. Dai, « Minimally Model-Based Trajectory Tracking and Variable Impedance Control of Flexible-Joint Robots », *IEEE Trans. Ind. Electron.*, vol. 68, n° 7, p. 6031-6041, juill. 2021, doi: 10.1109/TIE.2020.2994886.
- [5] A. Bazhanov, R. Vashchenko, et V. Rubanov, « Development of control system for a complex technological object using fuzzy behavior charts », *Heliyon*, vol. 6, n° 2, p. e03393, févr. 2020, doi: 10.1016/j.heliyon.2020.e03393.
- [6] T. Gurriet, A. Singletary, J. Reher, L. Ciarletta, E. Feron, et A. Ames, « Towards a Framework for Realizable Safety Critical Control through Active Set Invariance », in *2018 ACM/IEEE 9th International Conference on Cyber-Physical Systems (ICCPS)*, avr. 2018, p. 98-106. doi: 10.1109/ICCPS.2018.00018.
- [7] A. Rehan, I. Boiko, et Y. Zweiri, « Optimal non-parametric tuning of PID controllers based on classification of shapes of oscillations in modified relay feedback test », *J. Frankl. Inst.*, vol. 358, n° 2, p. 1448-1474, janv. 2021, doi: 10.1016/j.jfranklin.2020.12.007.
- [8] B. Ghosh et É. André, « Offline and online monitoring of scattered uncertain logs using uncertain linear dynamical systems », arXiv, arXiv:2204.11505, avr. 2022. doi: 10.48550/arXiv.2204.11505.
- [9] J. Hong, S. Laflamme, J. Dodson, et B. Joyce, « Introduction to State Estimation of High-Rate System Dynamics », *Sensors*, vol. 18, n° 1, Art. n° 1, janv. 2018, doi: 10.3390/s18010217.
- [10] A. Wu, H. Liu, et Z. Zeng, « Observer Design and H_∞ Performance for Discrete-Time Uncertain Fuzzy-Logic Systems », *IEEE Trans. Cybern.*, vol. 51, n° 5, p. 2398-2408, mai 2021, doi: 10.1109/TCYB.2019.2948562.
- [11] M. G. Bellemare *et al.*, « Autonomous navigation of stratospheric balloons using reinforcement learning », *Nature*, vol. 588, n° 7836, Art. n° 7836, déc. 2020, doi: 10.1038/s41586-020-2939-8.
- [12] G. Shi *et al.*, « Neural Lander: Stable Drone Landing Control Using Learned Dynamics », in *2019 International Conference on Robotics and Automation (ICRA)*, mai 2019, p. 9784-9790. doi: 10.1109/ICRA.2019.8794351.
- [13] A. Yahiaoui, J. Hensen, L. Soethout, et D. Van Paassen, « Model based optimal control for integrated building systems », in *Proceedings of the 6th International Postgraduate Research Conference in the Built and Human Environment*, 2006, p. 322-332.

- [14] Z. Shi, L. Zhao, Y. Zhao, et F. Gao, « Transient Performance Analysis of Adaptive Control with Control Architecture Modification », in *2019 Chinese Automation Congress (CAC)*, nov. 2019, p. 45-50. doi: 10.1109/CAC48633.2019.8996671.
- [15] Z. Zhang, Z. Yang, R. D. Sisson, et J. Liang, « Improving ceramic additive manufacturing via machine learning-enabled closed-loop control », *Int. J. Appl. Ceram. Technol.*, vol. 19, n° 2, p. 957-967, 2022.
- [16] W. Wang et Y. Li, « Distributed Fuzzy Optimal Consensus Control of State-Constrained Nonlinear Strict-Feedback Systems », *IEEE Trans. Cybern.*, 2022.
- [17] I. Sengupta, S. Gupta, D. Deb, et S. Ozana, « Dynamic Stability of an Electric Monowheel System Using LQG-Based Adaptive Control », *Appl. Sci.*, vol. 11, n° 20, Art. n° 20, janv. 2021, doi: 10.3390/app11209766.
- [18] J.-T. Huang et Y.-W. Jiang, « Robust Composite Adaptive Control of Linearisable Systems With Improved Performance », *IEEE Access*, vol. 9, p. 88037-88047, 2021, doi: 10.1109/ACCESS.2021.3090088.
- [19] P. A. Fisher et A. M. Annaswamy, « Discrete-Time Adaptive Control of a Class of Nonlinear Systems Using High-Order Tuners ». arXiv, 26 avril 2022.
- [20] B. T. Lopez et J.-J. E. Slotine, « Universal Adaptive Control of Nonlinear Systems », *IEEE Control Syst. Lett.*, vol. 6, p. 1826-1830, 2022, doi: 10.1109/LCSYS.2021.3133359.
- [21] D. H. Nguyen, M. H. Lowenberg, et S. A. Neild, « Identifying limits of linear control design validity in nonlinear systems: a continuation-based approach », *Nonlinear Dyn.*, vol. 104, n° 2, p. 901-921, avr. 2021, doi: 10.1007/s11071-021-06341-2.
- [22] Z. Liu *et al.*, « Fuzzy adaptive control technique for a new fractional-order supply chain system », *Phys. Scr.*, vol. 96, n° 12, p. 124017, 2021.
- [23] Z. Chen, J. Wang, K. Ma, P. Zhu, B. He, et C. Zhang, « Novel fuzzy event-triggered adaptive control for nonlinear systems with input hysteresis », *Soft Comput.*, vol. 25, n° 8, p. 6619-6631, 2021.
- [24] W. Liu et T. Zhao, « An active disturbance rejection control for hysteresis compensation based on Neural Networks adaptive control », *ISA Trans.*, vol. 109, p. 81-88, 2021.
- [25] M. Kamiński et K. Szabat, « Adaptive control structure with neural data processing applied for electrical drive with elastic shaft », *Energies*, vol. 14, n° 12, p. 3389, 2021.
- [26] L. Lu, P. Jin, G. Pang, Z. Zhang, et G. E. Karniadakis, « Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators », *Nat. Mach. Intell.*, vol. 3, n° 3, Art. n° 3, mars 2021, doi: 10.1038/s42256-021-00302-5.
- [27] N. Wang, Z. Fu, F. Tao, S. Song, et M. Ma, « Adaptive fuzzy fixed-time control for a class of strict-feedback stochastic nonlinear systems », *Syst. Sci. Control Eng.*, vol. 10, n° 1, p. 142-153, déc. 2022, doi: 10.1080/21642583.2022.2048320.
- [28] T. P. Mote et S. D. Lokhande, « Temperature control system using ANFIS », *Int. J. Soft Comput. Eng.*, vol. 2, n° 1, p. 2231-2307, 2012.
- [29] P. R. Pande, P. L. Paikrao, et D. S. Chaudhari, « Digital ANFIS Model Design », vol. 3, n° 1, p. 5, 2013.
- [30] K. W. Wong, T. Gedeon, A. Chong, et S. Khor, « Intelligent Data Analysis Using Neuro-Fuzzy Approach », p. 6.
- [31] H. Han, H. Liu, J. Li, et J. Qiao, « Cooperative fuzzy-neural control for wastewater treatment process », *IEEE Trans. Ind. Inform.*, vol. 17, n° 9, p. 5971-5981, 2020.

- [32] P. S. P. Eboule et A. N. Hasan, « Accurate fault detection and location in power transmission line using concurrent neuro fuzzy technique », *Przeegląd Elektrotechniczny*, vol. 97, 2021.
- [33] J.-S. R. Jang, « Fuzzy modeling using generalized neural networks and Kalman filter algorithm », in *Proceedings of the ninth National conference on Artificial intelligence - Volume 2*, Anaheim, California, juill. 1991, p. 762-767.
- [34] Q. Li, G. Han, W. Zhang, et Y. Yang, « Adaptive neuro-fuzzy sliding mode control guidance law with impact angle constraint », *IET Control Theory Appl.*, vol. 9, n° 14, p. 2115-2123, sept. 2015, doi: 10.1049/iet-cta.2014.1206.
- [35] A. Hajian et P. Styles, « Neuro-fuzzy Systems », in *Application of Soft Computing and Intelligent Methods in Geophysics*, A. Hajian et P. Styles, Éd. Cham: Springer International Publishing, 2018, p. 375-415. doi: 10.1007/978-3-319-66532-0_5.
- [36] R. Hou, Y. Hou, C. Wang, Q. Gao, et H. Sun, « A Hybrid Wavelet Fuzzy Neural Network and Switching Particle Swarm Optimization Algorithm for AC Servo System », *Math. Probl. Eng.*, vol. 2016, p. e9724917, déc. 2016, doi: 10.1155/2016/9724917.
- [37] S. Ganjefar et M. Tofighi, « A fuzzy wavelet neural network stabilizer design using genetic algorithm for multi-machine systems », *Przeegląd Elektrotechniczny*, n° R. 89, nr 5, p. 19-25, 2013.
- [38] M. N. Mohd Salleh et K. Hussain, « A review of training methods of ANFIS for applications in business and economics », *Int. J. U- E- Serv. Sci. Technol.*, vol. 9, n° 7, Art. n° 7, 2016.
- [39] W. P. D. Wilde, S. Hernandez, et S. Kravanja, *High Performance and Optimum Design of Structures and Materials III*. WIT Press, 2018.
- [40] H. Marzi, A. Haj Darwish, et H. Helfawi, « Training ANFIS Using the Enhanced Bees Algorithm and Least Squares Estimation », *Intell. Autom. Soft Comput.*, vol. 23, n° 2, p. 227-234, avr. 2017, doi: 10.1080/10798587.2016.1196880.
- [41] J. Zhang, « Improved on-line process fault diagnosis through information fusion in multiple neural networks », *Comput. Chem. Eng.*, vol. 30, n° 3, p. 558-571, janv. 2006, doi: 10.1016/j.compchemeng.2005.11.002.
- [42] M. P. Perrone et L. N. Cooper, « When networks disagree: Ensemble methods for hybrid neural networks », in *How We Learn; How We Remember: Toward an Understanding of Brain and Neural Systems*, vol. Volume 10, WORLD SCIENTIFIC, 1995, p. 342-358. doi: 10.1142/9789812795885_0025.
- [43] Sung-Bae Cho, « Fuzzy aggregation of modular neural networks with ordered weighted averaging operators », *Int. J. Approx. Reason.*, vol. 13, n° 4, p. 359-375, nov. 1995, doi: 10.1016/0888-613X(95)00059-P.
- [44] S. A. Emami et A. Banazadeh, « Online Identification of Aircraft Dynamics in the Presence of Actuator Faults », *J. Intell. Robot. Syst.*, vol. 96, n° 3-4, p. 541-553, déc. 2019, doi: 10.1007/s10846-019-00998-z.
- [45] S. A. Emami et A. Roudbari, « Multimodel ELM-Based Identification of an Aircraft Dynamics in the Entire Flight Envelope », *IEEE Trans. Aerosp. Electron. Syst.*, vol. 55, n° 5, p. 2181-2194, oct. 2019, doi: 10.1109/TAES.2018.2883848.
- [46] C. C. Coello, G. B. Lamont, et D. A. van Veldhuizen, *Evolutionary Algorithms for Solving Multi-Objective Problems*. Springer Science & Business Media, 2007.
- [47] M. P. Perrone, « When Networks Disagree: Ensemble Methods for Hybrid Neural Networks », p. 17.

- [48] H. F. Ghavidel, « Robust control of large-scale nonlinear systems by a hybrid adaptive fuzzy observer design with input saturation », *Soft Comput.*, vol. 22, n° 19, p. 6473-6487, oct. 2018, doi: 10.1007/s00500-017-2699-z.
- [49] Y. Bibi, O. Bouhali, et T. Bouktir, « Hybrid fuzzy direct/indirect adaptive controller for uncertain nonlinear systems », *Trans. Inst. Meas. Control*, vol. 42, n° 15, p. 3012-3023, nov. 2020, doi: 10.1177/0142331220939728.
- [50] Y.-S. Huang, L. Han, et Z. Wang, « Combined indirect & direct adaptive fuzzy decentralised control of large-scale systems with strong interconnection and its application », *Int. J. Syst. Sci.*, vol. 52, n° 3, p. 455-467, févr. 2021, doi: 10.1080/00207721.2020.1829741.
- [51] K. Kahili, O. Bouhali, et N. Rizoug, « Wavelet Functions Performances in Intelligent Direct Adaptive Control of Uncertain Nonlinear Systems », in *2018 International Conference on Electrical Sciences and Technologies in Maghreb (CISTEM)*, oct. 2018, p. 1-6. doi: 10.1109/CISTEM.2018.8613605.
- [52] K. Kahili, O. Bouhali, N. Rizoug, et F. Khenfri, « Hybrid FS–WNN approximator in indirect adaptive control of uncertain non-linear MIMO systems », *IET Control Theory Appl.*, vol. 14, n° 4, p. 539-548, 2020, doi: 10.1049/iet-cta.2019.0078.
- [53] K. Kahili, O. Bouhali, F. Khenfri, et N. Rizoug, « Novel Hybrid Intelligent Backstepping Controller for Chaotic Systems », in *2020 7th International Conference on Control, Decision and Information Technologies (CoDIT)*, 2020, vol. 1, p. 1185-1190.
- [54] K. Kahili, O. Bouhali, F. Khenfri, et N. Rizoug, « Robust Intelligent Self-tuning PID Controller for the Body-rate Stabilization of Quadrotors », in *IECON 2019-45th Annual Conference of the IEEE Industrial Electronics Society*, 2019, vol. 1, p. 5281-5286.
- [55] Q. Zhang et A. Benveniste, « Wavelet networks », *IEEE Trans. Neural Netw.*, vol. 3, n° 6, p. 889-898, nov. 1992, doi: 10.1109/72.165591.
- [56] Z. Zainuddin et O. Pauline, « Modified wavelet neural network in function approximation and its application in prediction of time-series pollution data », *Appl. Soft Comput.*, vol. 11, n° 8, p. 4866-4874, déc. 2011, doi: 10.1016/j.asoc.2011.06.013.
- [57] Q. Ye et L. Wei, « The Prediction of Stock Price Based on Improved Wavelet Neural Network », *Open J. Appl. Sci.*, vol. 05, n° 04, Art. n° 04, avr. 2015, doi: 10.4236/ojapps.2015.54012.
- [58] Z. Xia, H. Zhang, K. Shen, Y. Qu, et Z. Jiang, « Wavelet analysis of extended X-ray absorption fine structure data: Theory, application », *Phys. B Condens. Matter*, vol. 542, p. 12-19, août 2018, doi: 10.1016/j.physb.2018.04.039.
- [59] D. P. Yadav, « A brief description of wavelet and wavelet transforms and their applications », *Int. J. Stat. Appl. Math.*, vol. 3, n° 1, p. 266-271, janv. 2018.
- [60] L. Debnath et F. A. Shah, « The Wavelet Transforms », in *Lecture Notes on Wavelet Transforms*, L. Debnath et F. A. Shah, Éd. Cham: Springer International Publishing, 2017, p. 93-122. doi: 10.1007/978-3-319-59433-0_3.
- [61] B. Bathellier, D. Van De Ville, T. Blu, M. Unser, et A. Carleton, « Wavelet-based multi-resolution statistics for optical imaging signals: Application to automated detection of odour activated glomeruli in the mouse olfactory bulb », *NeuroImage*, vol. 34, n° 3, p. 1020-1035, févr. 2007, doi: 10.1016/j.neuroimage.2006.10.038.
- [62] S. Mezghani, « Identification of relevant wavelet functions for multiscale characterization of manufactured surfaces using a genetically optimized neural network », *Int. J. Adv. Manuf. Technol.*, vol. 96, n° 5, p. 1891-1903, mai 2018, doi: 10.1007/s00170-018-1756-9.

- [63] F. F. M. El-Sousy et K. A. Abuhasel, « Adaptive nonlinear disturbance observer using double loop self-organizing recurrent wavelet-neural-network for two-axis motion control system », in *2016 IEEE Industry Applications Society Annual Meeting*, oct. 2016, p. 1-14. doi: 10.1109/IAS.2016.7731869.
- [64] A. Solhmirzaei, S. Azadi, et R. Kazemi, « Road profile estimation using wavelet neural network and 7-DOF vehicle dynamic systems », *J. Mech. Sci. Technol.*, vol. 26, n° 10, p. 3029-3036, oct. 2012, doi: 10.1007/s12206-012-0812-x.
- [65] Q. Fang, Q. Zhu, et F. Qiao, *Advancements in Smart City and Intelligent Building: Proceedings of the International Conference on Smart City and Intelligent Building (ICSCIB 2018)*. Springer, 2019.
- [66] E. Papavasileiou et B. Jansen, « The importance of the activation function in NeuroEvolution with FS-NEAT and FD-NEAT », in *2017 IEEE Symposium Series on Computational Intelligence (SSCI)*, nov. 2017, p. 1-7. doi: 10.1109/SSCI.2017.8285328.
- [67] M. Y. Pir, F. A. Shah, et M. Asger, « Comparative Study of Different Wavelet Based Neural Network Models for IIP Growth Forecasting Using Different Yield Spreads », vol. 4, n° 6, p. 9, 2017.
- [68] M. M. Khan, A. Mendes, et S. K. Chalup, « Evolutionary Wavelet Neural Network ensembles for breast cancer and Parkinson's disease prediction », *PLoS One*, vol. 13, n° 2, p. e0192192, 2018, doi: 10.1371/journal.pone.0192192.
- [69] G. Wang, L. Guo, et H. Duan, « Wavelet Neural Network Using Multiple Wavelet Functions in Target Threat Assessment », *Sci. World J.*, vol. 2013, p. e632437, févr. 2013, doi: 10.1155/2013/632437.
- [70] G. Wang, L. Guo, et H. Duan, « Wavelet Neural Network Using Multiple Wavelet Functions in Target Threat Assessment », *The Scientific World Journal*, 2013.
- [71] M. Benosman, « Model-based vs data-driven adaptive control: An overview », *Int. J. Adapt. Control Signal Process.*, vol. 32, n° 5, p. 753-776, 2018, doi: 10.1002/acs.2862.
- [72] D. Zeng *et al.*, « Research on Improved Auto-Tuning of a PID Controller Based on Phase Angle Margin », *Energies*, 2019, doi: 10.3390/EN12091704.
- [73] L. A. Vázquez, F. Jurado, C. E. Castañeda, et A. Y. Alanis, « Real-Time Implementation of a Neural Integrator Backstepping Control via Recurrent Wavelet First Order Neural Network », *Neural Process. Lett.*, vol. 49, n° 3, p. 1629-1648, juin 2019, doi: 10.1007/s11063-018-9893-6.
- [74] L. Yang et H. Chen, « Fault diagnosis of gearbox based on RBF-PF and particle swarm optimization wavelet neural network », *Neural Comput. Appl.*, vol. 31, n° 9, p. 4463-4478, sept. 2019, doi: 10.1007/s00521-018-3525-y.
- [75] A. Kratsios, « The Universal Approximation Property: Characterizations, Existence, and a Canonical Topology for Deep-Learning », *ArXiv191003344 Cs Math Stat*, févr. 2020.
- [76] R. Ortega, V. Nikiforov, et D. Gerasimov, « On modified parameter estimators for identification and adaptive control. A unified framework and some new schemes », *Annu. Rev. Control*, vol. 50, p. 278-293, janv. 2020, doi: 10.1016/j.arcontrol.2020.06.002.
- [77] H.-G. Han, X.-L. Wu, Z. Liu, et J.-F. Qiao, « Design of Self-Organizing Intelligent Controller Using Fuzzy Neural Network », *IEEE Trans. Fuzzy Syst.*, vol. 26, n° 5, p. 3097-3111, oct. 2018, doi: 10.1109/TFUZZ.2017.2785812.
- [78] C.-M. Lin, L.-Y. Chen, et C.-H. Chen, « RCMAC Hybrid Control for MIMO Uncertain Nonlinear Systems Using Sliding-Mode Technology », *IEEE Trans. Neural Netw.*, vol. 18, n° 3, p. 708-720, mai 2007, doi: 10.1109/TNN.2007.891198.

- [79] N. Goléa, G. Debbache, et A. Goléa, « Neural network-based adaptive sliding mode control for uncertain non-linear MIMO systems », *Int. J. Model. Identif. Control*, vol. 16, n° 4, p. 334, 2012, doi: 10.1504/IJMIC.2012.048266.
- [80] W. Zheng et M. Chen, « Tracking Control of Manipulator Based on High-Order Disturbance Observer », *IEEE Access*, vol. 6, p. 26753-26764, 2018, doi: 10.1109/ACCESS.2018.2834978.
- [81] M. R. Parsaei, H. R. Boveiri, R. Javidan, et R. Khayami, « Telesurgery QoS improvement over SDN based on a Type-2 fuzzy system and enhanced cuckoo optimization algorithm », *Int. J. Commun. Syst.*, vol. 33, n° 11, p. e4426, 2020, doi: 10.1002/dac.4426.
- [82] O. T. Nkamgang, D. Tchiotso, B. S. Tchinda, et H. B. Fotsin, « A neuro-fuzzy system for automated detection and classification of human intestinal parasites », *Inform. Med. Unlocked*, vol. 13, p. 81-91, janv. 2018, doi: 10.1016/j.imu.2018.10.007.
- [83] Y. Kurniasari, B. Suseta, N. Hendiyani, et A. M. Abadi, « Classification of Open Unemployment Rate in Indonesia with Mamdani Fuzzy Inference System », *J. Phys. Conf. Ser.*, vol. 1581, n° 1, p. 012010, juill. 2020, doi: 10.1088/1742-6596/1581/1/012010.
- [84] M. L. Imrane, A. Melingui, J. J. B. Mvogo Ahanda, F. Biya Motto, et R. Merzouki, « Artificial potential field neuro-fuzzy controller for autonomous navigation of mobile robots », *Proc. Inst. Mech. Eng. Part J. Syst. Control Eng.*, vol. 235, n° 7, p. 1179-1192, août 2021, doi: 10.1177/0959651820974831.
- [85] W. Fang, F. Chao, C.-M. Lin, L. Yang, C. Shang, et C. Zhou, « An Improved Fuzzy Brain Emotional Learning Model Network Controller for Humanoid Robots », *Front. Neurorobotics*, vol. 13, 2019,
- [86] R. Kruse, S. Mostaghim, C. Borgelt, C. Braune, et M. Steinbrecher, « Hybrid Systems for Tuning and Learning Fuzzy Systems », in *Computational Intelligence: A Methodological Introduction*, R. Kruse, S. Mostaghim, C. Borgelt, C. Braune, et M. Steinbrecher, Éd. Cham: Springer International Publishing, 2022, p. 471-487. doi: 10.1007/978-3-030-42227-1_21.
- [87] A. Ismail et A. Ismail, « MODELING, SIMULATION AND PERFORMANCE ANALYSIS OF THE THREE PHASE PERMANENT MAGNET BRUSHLESS DIRECT CURRENT (PMBLDC) MOTOR USING NEURO-FUZZY CONTROLLERBASED GENETIC ALGORITHM », *Int. J. Curr. Res.*, vol. 14, p. 12, janv. 2022, doi: 10.24941/ijcr.42862.01.2022.
- [88] N. Walia, H. Singh, et A. Sharma, « ANFIS: Adaptive Neuro-Fuzzy Inference System- A Survey », *Int. J. Comput. Appl.*, vol. 123, n° 13, p. 32-38, août 2015, doi: 10.5120/ijca2015905635.
- [89] Y. Qin, R. Langari, et L. Gü, « A new modeling algorithm based on ANFIS and GMDH », *J. Intell. Fuzzy Syst.*, vol. 29, n° 4, p. 1321-1329, oct. 2015, doi: 10.3233/IFS-141443.
- [90] M. S. Iraj, « Multi-layer architecture for adaptive fuzzy inference system with a large number of input features », *Cogn. Syst. Res.*, vol. 42, p. 23-41, mai 2017, doi: 10.1016/j.cogsys.2016.11.006.
- [91] I. Chairez, « Multiple DNN identifier for uncertain nonlinear systems based on Takagi-Sugeno inference », *Fuzzy Sets Syst.*, vol. 237, p. 118-135, févr. 2014, doi: 10.1016/j.fss.2013.05.001.
- [92] C.-C. Tsai, F.-C. Tai, Y.-L. Chang, et C.-T. Tsai, « Adaptive Predictive PID Control Using Fuzzy Wavelet Neural Networks for Nonlinear Discrete-Time Time-Delay Systems », *Int. J. Fuzzy Syst.*, vol. 19, n° 6, p. 1718-1730, déc. 2017, doi: 10.1007/s40815-017-0405-z.

- [93] R. Hou, L. Wang, Q. Gao, Y. Hou, et C. Wang, « Indirect adaptive fuzzy wavelet neural network with self- recurrent consequent part for AC servo system », *ISA Trans.*, vol. 70, p. 298-307, sept. 2017, doi: 10.1016/j.isatra.2017.04.010.
- [94] X. Bu et H. Lei, « A fuzzy wavelet neural network-based approach to hypersonic flight vehicle direct nonaffine hybrid control », *Nonlinear Dyn.*, vol. 94, n° 3, p. 1657-1668, nov. 2018, doi: 10.1007/s11071-018-4447-z.
- [95] H. K. Nigam, R. N. Mohapatra, et K. Murari, « Wavelet approximation of a function using Chebyshev wavelets », *J. Inequalities Appl.*, vol. 2020, n° 1, p. 187, juill. 2020, doi: 10.1186/s13660-020-02453-2.
- [96] S. Dian, Y. Hu, T. Zhao, et J. Han, « Adaptive backstepping control for flexible-joint manipulator using interval type-2 fuzzy neural network approximator », *Nonlinear Dyn.*, vol. 97, n° 2, p. 1567-1580, juill. 2019, doi: 10.1007/s11071-019-05073-8.
- [97] N. Vahab et M. D. Nima, « Robust path tracking of a quadrotor using adaptive fuzzy terminal sliding mode control », *Control Engineering Practice*, p.47-63, vol. 110, 2021.
- [98] J. A. C. González, O. Salas-Peña, et J. De León-Morales, « Observer-based super twisting design: A comparative study on quadrotor altitude control », *ISA Trans.*, vol. 109, p. 307-314, mars 2021, doi: 10.1016/j.isatra.2020.10.026.
- [99] C.-H. Wang, T.-C. Lin, et H.-L. Liu, « Combined direct/indirect adaptive fuzzy-neural networks control with state observer and supervisory controller for nonlinear dynamic systems », in *10th IEEE International Conference on Fuzzy Systems. (Cat. No.01CH37297)*, déc. 2001, vol. 2, p. 880-883 vol.3. doi: 10.1109/FUZZ.2001.1009096.
- [100] S.-B. Cho, « Fuzzy aggregation of modular neural networks with ordered weighted averaging operators », *Int J Approx Reason*, vol. 13, n° 4, p. 359-375, nov. 1995, doi: 10.1016/0888-613X(95)00059-P.
- [101] T. Murofushi et M. Sugeno, « Fuzzy Measures and Fuzzy Integrals », p. 40.
- [102] W.-Y. Wang, Y.-H. Chien, C.-K. Yang, et T.-T. Lee, « Time-efficient structure for observer-based direct/indirect fuzzy-neural controller », in *2013 10th IEEE International Conference On Networking, Sensing And Control (ICNSC)*, avr. 2013, p. 404-408. doi: 10.1109/ICNSC.2013.6548772.
- [103] P. Burggräf, A. R. Pérez Martínez, H. Roth, et J. Wagner, « Quadrotors in factory applications: design and implementation of the quadrotor's P-PID cascade control system », *SN Appl. Sci.*, vol. 1, n° 7, p. 722, juin 2019, doi: 10.1007/s42452-019-0698-7.
- [104] « Commercial Drone Market Size & Share Report, 2021-2028 ». <https://www.grandviewresearch.com/industry-analysis/global-commercial-drones-market>
- [105] A. A. Pereira, J. P. Espada, R. G. Crespo, et S. R. Aguilar, « Platform for controlling and getting data from network connected drones in indoor environments », *Future Gener. Comput. Syst.*, vol. 92, n° C, p. 656-662, mars 2019, doi: 10.1016/j.future.2018.01.011.
- [106] M. Maboudi, M. Homaei, S. Song, S. Malihi, M. Saadatseresht, et M. Gerke, « A Review on Viewpoints and Path-planning for UAV-based 3D Reconstruction », p. 33.
- [107] V. N. Sankaranarayanan, « Adaptive Controllers for Quadrotors Carrying Unknown Payloads ». arXiv, 30 août 2021.
- [108] M. N. Shauqee, P. Rajendran, et N. M. Suhadis, « An effective proportional-double derivative-linear quadratic regulator controller for quadcopter attitude and altitude control », *Automatika*, vol. 62, n° 3-4, p. 415-433, oct. 2021, doi: 10.1080/00051144.2021.1981527.

- [109] S. I. Abdelmaksoud, M. Mailah, et A. M. Abdallah, « Robust Intelligent Self-Tuning Active Force Control of a Quadrotor With Improved Body Jerk Performance », *IEEE Access*, vol. 8, p. 150037-150050, 2020, doi: 10.1109/ACCESS.2020.3015101.
- [110] K. Yi, X. Liang, Y. He, L. Yang, et J. Han, « Active-Model-Based Control for the Quadrotor Carrying a Changed Slung Load », *Electronics*, vol. 8, n° 4, Art. n° 4, avr. 2019, doi: 10.3390/electronics8040461.
- [111] L. Wang, H. He, Z. Zeng, et M.-F. Ge, « Model-independent formation tracking of multiple Euler–Lagrange systems via bounded inputs », *IEEE Trans. Cybern.*, vol. 51, n° 5, p. 2813-2823, 2019.
- [112] B. Mu, K. Zhang, et Y. Shi, « Integral Sliding Mode Flight Controller Design for a Quadrotor and the Application in a Heterogeneous Multi-Agent System », *IEEE Trans. Ind. Electron.*, vol. 64, n° 12, p. 9389-9398, déc. 2017, doi: 10.1109/TIE.2017.2711575.
- [113] M. Islam, M. Okasha, E. Sulaeman, S. Fatai, et A. Legowo, « Performance evaluation of linear quadratic regulator and linear quadratic gaussian controllers on quadrotor platform », *Int. J. Recent Technol. Eng.*, vol. 7, p. 191-195, mars 2019.
- [114] S. Wang et Y. Yang, « Quadrotor aircraft attitude estimation and control based on Kalman filter », in *Proceedings of the 31st Chinese Control Conference*, juill. 2012, p. 5634-5639.
- [115] C. Acosta Lúa, C. C. Vaca García, S. Di Gennaro, B. Castillo-Toledo, et M. E. Sánchez Morales, « Real-Time Hovering Control of Unmanned Aerial Vehicles », *Math. Probl. Eng.*, vol. 2020, p. e2314356, juill. 2020, doi: 10.1155/2020/2314356.
- [116] G. Qi, S. Ma, X. Guo, X. Li, et J. Guo, « High-order differential feedback control for quadrotor UAV: Theory and experimentation », *Electronics*, vol. 9, n° 12, p. 2001, 2020.
- [117] M. Jafari et H. Xu, « Intelligent Control for Unmanned Aerial Systems with System Uncertainties and Disturbances Using Artificial Neural Network », *Drones*, vol. 2, n° 3, Art. n° 3, sept. 2018, doi: 10.3390/drones2030030.
- [118] N. Dalwadi, D. Deb, et S. M. Muyeen, « Adaptive backstepping controller design of quadrotor biplane for payload delivery », *IET Intell. Transp. Syst.*, vol. n/a, n° n/a, doi: 10.1049/itr2.12171.
- [119] F. Qiao, J. Shi, X. Qu, et Y. Lyu, « Adaptive Back-stepping Neural Control for an Embedded and Tilttable V-tail Morphing Aircraft », *Int. J. Control Autom. Syst.*, vol. 20, n° 2, p. 678-690, 2022.
- [120] T. Madani et A. Benallegue, « Adaptive Control via Backstepping Technique and Neural Networks of a Quadrotor Helicopter », *IFAC Proc. Vol.*, vol. 41, n° 2, p. 6513-6518, janv. 2008, doi: 10.3182/20080706-5-KR-1001.01098.
- [121] T. Dierks et S. Jagannathan, « Output feedback control of a quadrotor UAV using neural networks », *IEEE Trans. Neural Netw.*, vol. 21, n° 1, p. 50-66, 2009.
- [122] T. Dierks et S. Jagannathan, « Neural network control of quadrotor UAV formations », in *2009 American Control Conference*, juin 2009, p. 2990-2996. doi: 10.1109/ACC.2009.5160591.
- [123] H. M. Guzey, T. Dierks, S. Jagannathan, et L. Acar, « Modified Consensus-based Output Feedback Control of Quadrotor UAV Formations Using Neural Networks », *J. Intell. Robot. Syst.*, vol. 94, n° 1, p. 283-300, avr. 2019, doi: 10.1007/s10846-018-0961-y.
- [124] Z. Jia, L. Wang, J. Yu, et X. Ai, « Distributed adaptive neural networks leader-following formation control for quadrotors with directed switching topologies », *ISA Trans.*, vol. 93, p. 93-107, oct. 2019, doi: 10.1016/j.isatra.2019.02.030.

- [125] W. Jasim et D. Gu, « Robust Team Formation Control for Quadrotors », *IEEE Trans. Control Syst. Technol.*, vol. 26, n° 4, p. 1516-1523, juill. 2018, doi: 10.1109/TCST.2017.2705072.
- [126] H. Gao, C. Liu, D. Guo, et J. Liu, « Fuzzy adaptive PD control for quadrotor helicopter », in *2015 IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems (CYBER)*, 2015, p. 281-286.
- [127] F. J. Niroumand, A. Fakharian, et M. S. Seyedsajadi, « Fuzzy integral backstepping control approach in attitude stabilization of a quadrotor UAV », in *2013 13th Iranian Conference on Fuzzy Systems (IFSC)*, 2013, p. 1-6.
- [128] M. Pazooki et A. H. Mazinan, « Hybrid fuzzy-based sliding-mode control approach, optimized by genetic algorithm for quadrotor unmanned aerial vehicles », *Complex Intell. Syst.*, vol. 4, n° 2, p. 79-93, juin 2018, doi: 10.1007/s40747-017-0051-y.
- [129] M. Idrissi, M. Salami, et F. Annaz, « A Review of Quadrotor Unmanned Aerial Vehicles: Applications, Architectural Design and Control Algorithms », *J. Intell. Robot. Syst.*, vol. 104, n° 2, p. 22, févr. 2022, doi: 10.1007/s10846-021-01527-7.