



**Université Batna 2 – Mostefa Ben Boulaïd**  
**Faculté de Technologie**  
**Département de L'Electronique**



**Thèse**

Préparée au sein du Laboratoire d'Automatique Avancée et d'Analyse des Système  
**LAAAS**

Présentée pour l'obtention du diplôme de :  
**Doctorat en Sciences en Electronique**  
**Option : Robotique**

Sous le Thème :

**Modélisation et commande des drones**

Présentée par :

**BOUHENTALA Moufid**

**Devant le jury composé de :**

M. SAIDI Lamir	Prof.	Université de Batna	Président
M. CHAFAA Khiereddine	Prof.	Université de Batna	Rapporteur
M. KHIREDINE Mohammed Salah	Prof.	Université de Batna	Co-Rapporteur
M. TOUBA Mostefa Mohamed	MCA.	Université de Biskra	Examineur
M. AOUCHE Abdelaziz	MCA	Université de Tébessa	Examineur
M. DJEDDI Abdelghani	MCA.	Université de Tébessa	Examineur
M. BENBRAHIM Meriem	MCB.	Université de Batna	Invité

**30/01/2023**

# INTRODUCTION GENERALE

Ces dernières années, l'intérêt pour les Véhicules Aériens Autonomes (*Unmanned Aerial Vehicle, UAV*), s'est constamment accru. Ces véhicules aériens, aussi appelés Drones, ont été employés, au début, principalement dans le domaine militaire, dans des missions de reconnaissance, de surveillance et aussi dans un rôle offensif. Au fur et à mesure que les technologies informatiques, de conception de capteurs et d'actionneurs progressent, les drones se sont aussi perfectionnés.

De nos jours, nous pouvons trouver des applications civiles telles que, la surveillance de trafic routier, la surveillance et la protection de l'environnement, la recherche et le secours de blessés ou la gestion de grandes infrastructures à savoir, les lignes haute tension, les barrages et les ponts.

La recherche en robotique aérienne a principalement impliqué des hélicoptères et des architectures "VTOL" (Vertical Take-off and Landing). Les principaux avantages de ces plateformes sont leur manœuvrabilité et leur capacité à effectuer des vols stationnaires. L'hélicoptère à quatre rotors (Quadrirotor) a attiré particulièrement l'attention dans la communauté robotique et automatique. Ce véhicule est capable d'atterrir et de décoller à la verticale et en raison de sa symétrie, il est simple à modéliser et à construire. Néanmoins, malgré une complexité moindre de cet engin par rapport aux hélicoptères conventionnels, la commande de l'hélicoptère à quatre rotors est un défi intéressant en raison des non-linéarités, des forces aérodynamiques et de son faible moment d'inertie.

En général, la position dans l'espace de ces engins est commandée par un opérateur. Le faire voler d'une façon stable et autonome est une tâche délicate qui nécessite des pilotes expérimentés et afin de réduire la difficulté de piloter, une commande d'attitude peut être ajoutée. Cette dernière doit être implantée dans un calculateur embarqué et capable de le stabiliser automatiquement. Ainsi, le pilote ne doit plus fournir que la consigne d'attitude et la commande embarquée doit être capable d'amener le véhicule à cette position angulaire et le maintenir en dépit de l'existence de perturbations externes.

Les travaux présentés dans cette thèse, consistent en la modélisation et le développement d'une commande d'un hélicoptère à quatre rotors, afin d'améliorer la poursuite de la trajectoire et la commande d'attitude.

Néanmoins, maintenir des performances désirées nécessite une commande adaptative permettant de fournir une approche systématique pour l'ajustement automatique d'un régulateur en temps réel. Ce type de commande est aussi caractérisé par la présence d'une boucle d'identification en temps réel. Mais, les approximateurs universels comme les systèmes flous nécessitent une initialisation des paramètres ajustées.

Par conséquent, notre contribution réside dans le développement des techniques de modélisation et de commande floue de type Takagi-Sugeno (TS).

Au niveau de l'identification, nous allons utiliser les méthodes floues de clustering des données entrée-sortie pour construire des fonctions dynamiques non linéaires. Le type d'approche suivi, peut être considéré comme une approche de type boîte grise qui représente le système dynamique non linéaire comme un modèle à base de règles, concaténant un ensemble de sous modèles localement linéaires sous la forme d'auto-régression non-linéaire (NARX). A partir de cette représentation, nous développons une commande de type mode glissant. Nous nous sommes intéressés, en particulier, à la commande adaptative par retour d'état basée sur des modèles TS affines.

L'organisation de cette thèse est la suivante :

Le **Chapitre 1** consiste en une introduction générale sur les drones et plus particulièrement sur les drones miniaturés à voilure tournante. Nous présentons les applications et les configurations de véhicules les plus répandues avec un état de l'art des diverses stratégies de commande employées dans la littérature. Nous présentons le modèle du quadrirotor qui sera utilisé pour la synthèse d'algorithmes de commande.

Dans le **Chapitre 2**, nous traitons le problème d'identification des modèles flous à partir des données entrée-sortie. Nous introduisons d'abord la modélisation floue de systèmes, en se focalisant particulièrement sur le modèle de type Takagi-Sugeno. Pour la construction de tels modèles nous abordons d'une part, les différentes méthodes de clustering flou et, d'autre part, nous présentons l'application d'une méthodologie de construction de modèle flou.

Le **Chapitre 3**, traite le cadre formel permettant la synthèse du modèle flou de type-2. Ensuite, nous présentons l'approche proposée pour la modélisation floue type-2 qui sera utilisée pour approximer les différentes fonctions. Le modèle flou TS est construit en trois

étapes : 1) Génération de clusters de données floues, 2) Estimation des fonctions d'appartenance, en utilisant un nouvel algorithme de détection d'enveloppe (EDA), et enfin, la 3<sup>ème</sup> étape est destinée au calcul des paramètres des fonctions conséquentes en utilisant les moindres carrés pour les antécédents et le Filtre de Kalman pour les conséquents.

Le chapitre est terminé par des simulations pour montrer l'efficacité de l'approche proposée.

Dans le **Chapitre 4**, nous considérons la possibilité qu'offrent les modèles flous de type TS pour la commande des systèmes non linéaires, en utilisant la commande adaptative floue. Nous commençons par un aperçu général sur la synthèse (stabilisation) des contrôleurs basés sur l'approximation floue TS des fonctions inconnues. Ensuite, nous développons spécifiquement une commande Adaptative floue par mode glissant par retour d'état pour le système de quadrirotor. L'élaboration de la commande consiste en premier lieu, en l'identification des fonctions inconnues du système et son implémentation dans notre contrôleur adaptative en second lieu. La commande est testée par des simulations de différents scenarios pour sa validation.

Enfin, nous terminons ce travail par une conclusion générale et quelques prospectives qui nous semblent les plus intéressantes à étudier.

# Dédicaces

À ma mère,

Ma source éternelle d'inspiration et de bénédiction,

À mon père,

Mon maître de passion, de patience et de persévérance,

À mes frères, À mes nièces et mes neveux,

À tous mes amis

---

# Remerciements

Le travail présenté dans ce mémoire a été effectué, à la faculté de Technologie, au sein de l'université de Mostefa Ben Boulaid (BATNA 2).

Je remercie ALLAH le tout puissant pour la volonté et la patience qu'il m'a donné tout au long de mes études.

Je tiens à exprimer mes profonds remerciements à Monsieur CHAFAA Khereddine, Directeur de thèse et Professeur à l'université de Batna 2 et Monsieur KHAIRDDINE Mohamed Salah, Professeur à l'université de Batna 2, pour avoir dirigé et co-dirigé ce travail, pour leurs nombreux conseils ainsi que leur soutien tout au long de cette thèse.

J'ai été honoré que Monsieur SAIDI Amir, Professeur à l'Université de Batna 2, ait accepté d'examiner ce travail et de présider le jury de soutenance.

Mes profonds remerciements à Messieurs TOUBA Mostefa, Maître de Conférences à l'université de Biskra, AOUCHE Abdelaziz et DJEDDI Abdelghani, Maîtres de Conférences à l'université de Tebessa qui ont accepté d'examiner mon manuscrit de thèse.

Je remercie Madame BENBRAHIM, Maître de conférences à l'Université de Batna 2, pour son soutien et aide durant la préparation de cette thèse.

Que tous les membres, collègues et amis de l'université de Batna 2 trouvent ici le témoignage de mon amitié et de ma reconnaissance, en particulier Smail, Ghoulam, Yacin et Sif. Enfin, je ne saurais terminer sans remercier toute ma famille, qui m'a accompagné tout au long de mes études par son amour inconditionnel et son soutien constant.

---

# SOMMAIRE

Remerciements

Dédicaces

INTRODUCTION

1

**1 ETAT DE L'ART ET MODÉLISATION DES QUADRIROTORS.....4**

**1.1 Introduction .....4**

**1.2 applications des drones ..... 5**

**1.3 Catégorie des drones .....6**

1.3.1 Classification selon leur taille .....6

1.3.2 classifications selon leur configuration .....7

**1.4. Configuration de véhicule Quadrirotor .....8**

**1.5. Etat de l'art sur la commande des quadrirotors..... 9**

**1.6 Modélisation de Quadrirotor .....12**

1.6.1 Fonctionnement et Conception de base..... 13

1.6.2 Etude cinématique .....14

1.6.3 Forces et Couples .....16

1.6.4 Mouvements angulaires .....16

1.6.5 Etude dynamique du système..... 18

**1.7 Conclusion .....24**

**2 IDENTIFICATION FLOUS A PARTIR DE DONNEES ..... 25**

**2.1 Introductions .....26**

**2.2 Modélisation floue de systèmes .....26**

2.2.1. Structure générale et différents types de modèles flous .....26

2.2.2. Le modèle flou de type Takagi-Sugeno TS .....29

**2.3 Identification (Construction) de modèles flous .....31**

2.3.1. Structure du modèle flou.....31

2.3.2. L'identification basé sur les données .....31

**2.4 Coalescence floue (clustering flou) .....32**

2.4.1. Notion de base.....32

2.4.2. Coalescence floue .....32

**2.5 Méthodes de clustering flou.....33**

2.5.1 Méthode FCM..... 35

2.5.2 Méthode GK .....37

2.5.3	Méthode GG.....	40
<b>2.6</b>	<b>Méthodologie générale .....</b>	<b>41</b>
2.6.1	Conception d'identification et acquisition des données.....	42
2.6.2	Sélection de la structure entrée-sortie .....	42
2.6.3	clustering flou .....	42
2.6.4	Obtention du modèle TS.....	43
2.6.5	Validation du modèle TS.....	43
2.6.6	Obtention de la base de règles du modèle flou TS.....	43
2.6.7	Génération des fonctions d'appartenance des antécédents .....	44
<b>2.7</b>	<b>Paramètres des conséquents .....</b>	<b>46</b>
2.7.1	utilisant la technique de moindre carre.....	46
2.7.2	utilisant le filtre de KALMAN .....	47
<b>2.8</b>	<b>Validation du modèle .....</b>	<b>49</b>
<b>2.9</b>	<b>Conclusion .....</b>	<b>49</b>
<b>3</b>	<b>IDENTIFICATION DES MODELES FLOUS TYPE-2 .....</b>	<b>51</b>
<b>3.1</b>	<b>Introduction.....</b>	<b>51</b>
<b>3.2</b>	<b>Logique floue type-2.....</b>	<b>51</b>
<b>3.3</b>	<b>Systèmes à logique floue Takagi-Sugeno de type-2 .....</b>	<b>56</b>
<b>3.4</b>	<b>Algorithme d'envelopper les données projetées .....</b>	<b>57</b>
3.4.1.	Conception des fonctions d'appartenance de type-2 .....	57
3.4.2.	Détections des enveloppes et algorithme d'estimation .....	59
<b>3.5</b>	<b>Modélisation flou de type-2 .....</b>	<b>62</b>
<b>3.6</b>	<b>Simulation .....</b>	<b>64</b>
3.6.1.	Approximation d'une fonction Non linéaire.....	64
3.6.2.	Modélisation de procédés de neutralisation du pH .....	67
3.6.3	Modélisation d'un système no linéaire dynamique.....	69
3.6.4	Identification et commande d'un système dynamique .....	70
<b>3.7</b>	<b>Conclusion .....</b>	<b>76</b>
<b>4</b>	<b>COMMANDE ADAPTATIVE FLOUE D'UN QUADROTOR.....</b>	<b>77</b>
<b>4.1.</b>	<b>Introduction .....</b>	<b>77</b>
<b>4.2.</b>	<b>Commande par mode glissant.....</b>	<b>77</b>
<b>4.3.</b>	<b>Commande adaptative floue.....</b>	<b>79</b>
4.3.1	Approximation .....	80
<b>4.4.</b>	<b>Conception de la loi de commande .....</b>	<b>81</b>

4.4.1 terme de contrôle d'équivalence .....	82
4.4.2 les lois d'adaptation .....	83
4.4.3 modifications par projection .....	84
4.4.4. Terme de mode glissant .....	85
4.4.5. Stabilité .....	86
<b>4.5. Application au Quadrirotor.....</b>	<b>88</b>
4.5.1 Identification du système .....	89
4.5.2. Commande proposée .....	92
4.5.3. Résultat et discussion .....	98
<b>4.6. Conclusion .....</b>	<b>107</b>

**CONCLUSION ET PERSPECTIVE**

## TABLE DES FIGURES

Fig. 1.1 répartition des Application des drones.....	5
Fig. 1.2 Ingénuité à l'intérieur du simulateur spatial du NASA. ....	6
Fig. 1.3 Catégories des Drones selon la taille.....	7
Fig. 1.4– Hélicoptère à quatre rotors de <i>Bothezat</i> .....	8
Fig. 1.5 Drone à quatre rotors.....	9
Fig. 1.6 Fonctionnement de l'hélicoptère à quatre rotors .....	15
Fig. 1.7 a) Mouvement de tangage du quadrirotor b) Mouvement de roulis du quadrirotor....	17
Fig. 2.1 Représentation générale d'un système flou. ....	28
Fig. 2.2 Résultats de l'algorithme FCM par l'ensemble de données de motorcycle. ....	37
Fig.2.3. Interprétation géométrique d'une matrice de covariance d'un cluster.....	39
Fig. 2.4 Résultats de l'algorithme de GK par l'ensemble de données synthétiques.....	40
Fig. 2.5 Résultats de l'algorithme de Gath–Geva par l'ensemble de données synthétiques....	41
Fig. 2.6 Illustration du mécanisme de projection.....	45
Fig. 2.7. Image complète du KF .....	49
Fig. 3.1 Structure d'un système flou type-2 , .....	52
Fig. 3.2 Ensemble flou type-2 .....	52
Fig. 3.3 Illustration des points de commutation pour le calcul de $y_l$ et $y_r$ .....	55
Fig. 3.4 Approximation des données bruitées projetées par une fonction d'appartenance.....	58
Fig. 3.5 (a) Détection d'enveloppes (b) Régression gaussienne.....	58
Fig. 3.6 Algorithme EDA2 (a) avant réajustement (b) après réajustement. ....	61
Fig. 3.7 Algorithme proposé par la méthode 2 .....	62
Fig. 3.8 Résultats obtenu après l'application de EDA .....	62
Fig. 3.9 (a) GK clustering (b) détection (c) Estimation .....	64
Fig. 3.10 (a) EDA1 détection des enveloppes (b) EDA2 corrections des intersections.....	65
Fig. 3.11 Fonctions de conséquents .....	65
Fig. 3.12 Modélisation et comparaison type-1/typ-2 .....	66
Fig. 3.13 Approximation de la fonction sinc(x) .....	67
Fig. 3.14 Clusters et projection des MFs de données normalisées de Neutralisation du pH....	68
Fig. 3.15 Identification T1FM / IVFM du processus de pH Neutralisation . ....	69
Fig. 3.16 Identification du modèle dynamique type-1/type-2 .....	70
Fig. 3.17 Commande floue indirecte proposée.....	72
Fig. 3.18 Présentation de données utilisées pour la construction du modèle.....	73
Fig. 3.19 Clusters et projections $\mu_{ij}$ de données.....	73

Fig. 3.20 Fonctions d'appartenance de système de type-1 et de type-2.....	73
Fig. 3.21 Résultats de la sortie identifiée.....	74
Fig. 3.22 Adaptation des paramètres de contrôleur basé sur le modèle flou type-2.....	75
Fig. 3.23 Estimation obtenue avec et sans adaptation.....	75
Fig. 3.24 Comparaison entre la commande adaptative type-1 et type-2.....	75
Fig. 4.1 Illustration des projections des lois d'adaptation .....	85
Fig. 4.2 Commande floue indirecte proposée.....	87
Fig. 4.3 Projection et construction des fonctions type-1 et type-2.....	90
Fig. 4.4 Fonctions d'appartenance de type-2/type-1 correspondant au $\hat{f}_1(\underline{x})$ .....	90
Fig. 4.5 Fonctions d'appartenance de type-2/type-1 correspondant au $\hat{f}_2(\underline{x})$ .....	91
Fig. 4.6 Fonctions d'appartenance de type-2/type-1 correspondant au $\hat{f}_3(\underline{x})$ .....	91
Fig. 4.7 comparaison entre $\hat{f}_1(\underline{x})$ et $f_1(\underline{x})$ .....	92
Fig. 4.8 comparaison entre $\hat{f}_2(x)$ et $f_2(x)$ .....	92
Fig. 4.9 comparaison entre $\hat{f}_3(x)$ et $f_3(x)$ .....	92
Fig 4.10 structure globale du contrôleur proposé.....	98
Fig. 4.11 Réponses du système avec une trajectoire sinusoïdale.....	99
Fig. 4.12 Trajectoire de l'erreur de suivi sur les droites de glissement .....	100
Fig. 4.13 Réponses du système avec une trajectoire sinusoïdale .....	100
Fig. 4.14 Signaux des vitesses des quatre moteurs (rad/s <sup>2</sup> ) .....	101
Fig. 4.15 Adaptations des Paramètres $\theta_{g_1}, \theta_{g_2}$ et $\theta_{g_3}$ .....	101
Fig. 4.16 Adaptations des Paramètres $\theta_{f_1}, \theta_{f_2}$ et $\theta_{f_3}$ .....	102
Fig. 4.17 réponse du système avec une trajectoire sinusoïdale .....	103
Fig. 4.18 Angles correspondants aux réponses de position.....	103
Fig 4.19 CMG adaptative en présence de perturbation $\delta J_r = 10\% J_r$ .....	104
Fig 4.20 CMG adaptative en présence de perturbation $\delta J_r = 15\% J_r$ .....	104
Fig 4.21 CMG adaptative en présence des perturbations $\delta J_r = 20\% J_r$ .....	105
Fig 4.22 Adaptation des paramètres des fonctions $f_i(x)$ .....	105
Fig 4.23 Les signaux de commande $U_1, U_2, U_3$ et $U_4$ .....	106
Fig. 4.24 Angles correspondant avec perturbation $\delta\varphi = \delta\theta = 0.25rad$ .....	106
Fig 4.25 Réponse de système utilisant la CMG adaptative basée sur le modèle flou.....	107

## Etat de l'art et modélisation des quadrirotors

### 1.1. Introduction

Les objets volants ont toujours exercé une grande fascination sur l'homme en encourageant toutes sortes de recherches et de développement. Le défi scientifique dans la conception et le contrôle dans des environnements encombrés et le manque de solutions existantes a été très motivant. D'autre part, le vaste champ d'applications dans les marchés militaires et civils a été encourageant pour leur diffusion. Les limitations des robots de terrain en sol et les progrès récents dans la technologie ont poussés les chercheurs vers le développement de nouveaux concepts de mobilité. Cela comprend les systèmes de vol sur lequel on pourrait appliquer les techniques déjà mises au point sur les robots au sol. Cependant, la tâche n'est pas facile en raison de plusieurs problèmes ouverts. Dans le domaine des technologies de détection, l'industrie peut actuellement fournir une nouvelle génération de micro-unité de mesure inertielle intégrée, composée généralement d'un System Micro-Electro Mécanique (MEMS), capteurs inertiels et magnéto-résistive. La dernière technologie de stockage d'énergie à haute densité technologie a été développée à l'origine pour des applications portatives et est maintenant largement utilisée dans la robotique aérienne. La réduction des coûts et de la taille de ces systèmes est très intéressante pour le marché civil. En même temps, limite ses performances et le contrôle devient difficile.

Dans ce chapitre nous allons commencer par les différentes applications des drones, ensuite nous présentons les différentes catégories de Véhicules Aériens Autonomes (UAV). Ensuite nous allons concentrer notre attention sur le mini hélicoptère à quatre rotors, considéré comme un Micro-UAV. Ce véhicule aérien est capable d'atterrir et de décoller à la verticale. Nous allons présenter des généralités sur les travaux concernant la commande de l'hélicoptère à quatre rotors et finalement nous allons présenter la modélisation de drone.

### 1.2 Applications des drones

L'utilisation des drones miniaturés est encore limitée à ce jour, essentiellement à cause des réglementations restreignant ou interdisant leur utilisation au sein de l'espace aérien.

Néanmoins, le nombre d'applications envisageables pour ce type de véhicule, ainsi que l'intérêt qu'il suscite chez de nombreux industriels, permettent d'espérer un futur assouplissement de cette réglementation. Ces applications relèvent tant du domaine civil que militaire figure ( fig.1.1) [3].

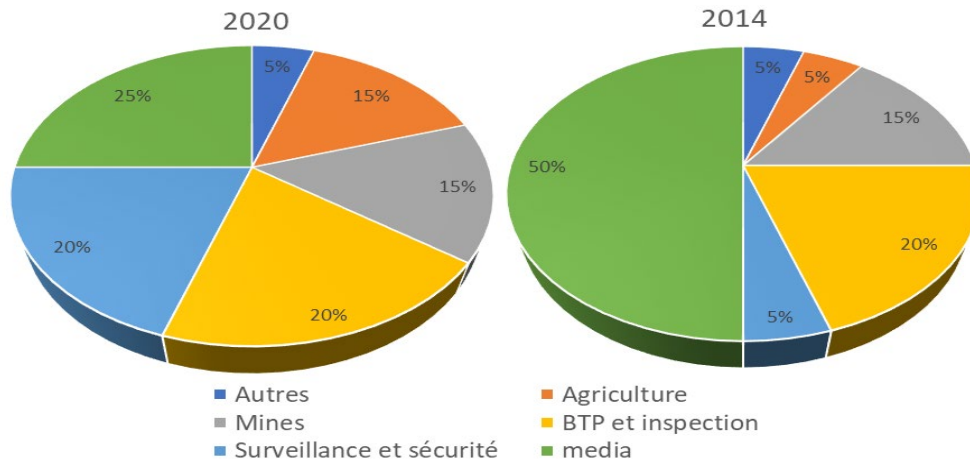


Fig.1.1 Répartition des Application des drones

Les applications militaires potentielles sont, par exemple, l'aide au fantassin, la reconnaissance de zones à risques, la cartographie de régions ou de bâtiments, la localisation de victimes lors de conflit, etc. Equipé d'une caméra, le véhicule réalise alors une fonction d'œil déporté. L'absence d'humain à bord permet son utilisation dans des contextes dangereux où la perte du véhicule peut être envisagée, d'autant plus facilement que son coût est bas. La faible taille du véhicule peut de plus rendre possible son utilisation par une seule personne, et faciliter ainsi son transport dans des zones de conflit. Enfin, la capacité de décollage et d'atterrissage vertical autorise un déploiement rapide depuis des zones très restreintes.

Dans le domaine civil, leurs applications sont nombreuses dans la localisation de victimes lors de catastrophes naturelles, surveillance en milieu urbain, analyse ou collecte d'échantillons en atmosphère polluée, exploration de zones contaminées, supervision du trafic routier, prises de vues de bâtiments ou d'ouvrages d'art pour la maintenance, épandage agricole, inspection de lignes électriques ou de voies ferrées, etc. Le drone est principalement utilisé dans sa fonction d'œil déporté, pour laquelle la capacité de vol stationnaire est mise à profit, et pour des missions difficiles d'accès pour l'homme ou dans des milieux dangereux pour lui (sites pollués ou irradiés).

Parmi les applications récentes des drones on trouve aussi l'exploration spatiale. L'hélicoptère de Mars, Ingénuité (figure 1.2) est un petit avion autonome qui a transporté à la

surface de la planète rouge attaché au ventre du rover *persévérance*. L'Ingénuité ouvre ainsi un nouvel avenir dans l'exploration du système martien.



**Fig 1.2** Ingénuité à l'intérieur du simulateur spatial du NASA.  
Photo : NASA/JPL-CALTECH

Les drones miniatures à voilure tournante trouvent également leur place comme plateformes expérimentales peu onéreuses et faciles à mettre en œuvre. Ils servent ainsi à mettre en pratique et à illustrer par l'expérience des travaux de recherche dans plusieurs domaines à savoir : la robotique, l'automatique, traitement du signal, l'informatique temps réel, la vision, etc.

### 1.3 Catégories de drones

Si les drones sont presque à la portée de tous pour la conception, la technologie et la production, les algorithmes de commande et d'observation relèvent des recherches évoluées. Ceci explique la majorité des projets déclarés à l'étude ou déjà en phase de développement émanant plusieurs pays, ces deniers ont amorcé le virage vers le développement technologique et accordent une place croissante au concept de véhicule sans pilote et autonome [1].

#### 1.3.1 Selon leur taille

Les drones peuvent être classés en fonction de leur taille, de leur application ou des deux en même temps. Sachant bien que la taille constitue le critère dominant. (Voir la figure 1.3). Ces catégories sont résumées ci-après [1] :

**Micro-UAV** : cette catégorie d'UAV est caractérisée par le fait qu'une seule personne suffit aux opérations. Ce type de drones est en général propulsé électriquement. Ainsi, ils permettent de faire des vols à l'intérieur. Le cout peu élevé des matériaux et des composants rend cette catégorie attractive pour réaliser des applications civiles peu couteuses.

**Mini-UAV** : les opérations de cette catégorie de drones exigent une équipe fournie en plus que celle de micro-UAV. Ces drones volent en général à une vitesse de 70 km/h à une altitude d'environ 3,5 km. Leur durée d'intervention est d'environ 4h avec une charge utile de 4kg.

**MALE/HALE** :(Medium Altitude High Endurance) ces drones sont utilisés pour des vols de plus longue durée à une altitude opérationnelle de 10 à 15 Km et à une altitude d'environ 20 km. Ces deux types de drones font partie de la classe de grande taille pouvant accomplir des missions atteignant 2 jours. Les représentants les plus connus sont le 'Predator' (MALE) et le Global Hawk (HALE) des forces armées Nord-Américaines.

**UCAV** : ce type de drones (*Unmanned Combat Air Vehicles*) est celui qui se trouve entre les mains des forces armées jouant un rôle offensif. A l'heure actuelle, ces drones se trouvent au stade des études ou de la réalisation de prototypes, par exemple le X-45 de Boeing.

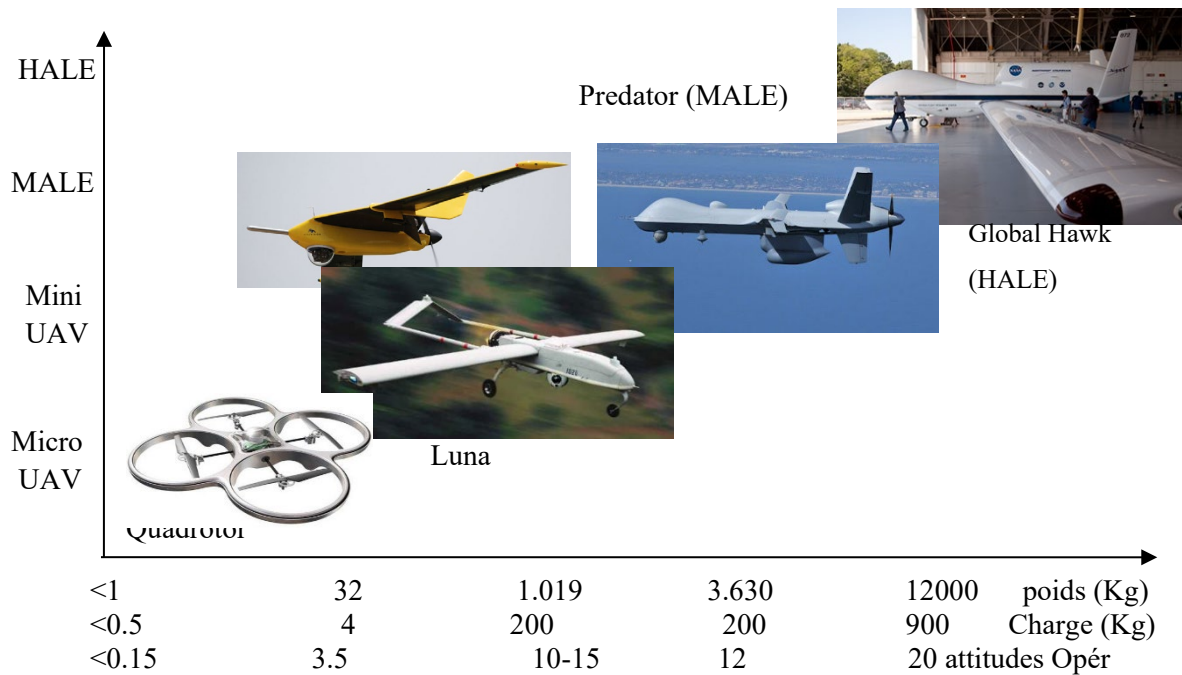


Fig.1.3 Catégories des Drones selon la taille

### 1.3.2 Selon leur configuration

Le fonctionnement aérodynamique du drone fournit une autre possibilité de classification, principalement en trois familles : ceux à voilures fixes, ceux à voilures tournantes et les aéronefs à ailes battantes (ornicoptères).

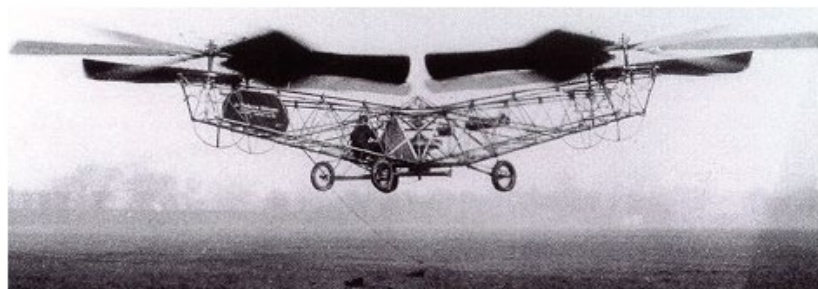
[2] montre que la recherche en robotique aérienne s'est principalement focalisée sur des drones à voilures tournantes : des hélicoptères et des architectures "VTOL". Ces véhicules capables de décoller et d'atterrir à la verticale, ont été employés pour des applications militaires et civiles tels que ; le secours et la recherche de blessés dans un désastre, la surveillance environnementale, la cartographie aérienne, la surveillance du trafic routier, la cinématographie ou encore l'anthropologie.

Il existe de nombreuses configurations de véhicules de drones miniatures à voilure tournante, suivant le nombre de rotors, leur position et la présence ou non d'autres actionneurs. Dans la suite nous nous intéressons à la configuration de véhicule de **Quadrirotor**.

### 1.4 Configuration de véhicule quadrirotor.

Le véhicule est équipé de quatre rotors disposés en croix. Les rotors avant et arrière tournent dans le sens contraire des rotors latéraux. Les rotors étant à pas fixe, le pilotage en tangage roulis lacet est réalisé par différence de vitesse de rotation, La simplicité mécanique de ce véhicule en fait une configuration très abordable pour la réalisation de plateformes expérimentales à faible coût.

Cependant, le concept de l'hélicoptère à quatre rotors n'est pas nouveau. Le premier hélicoptère à quatre rotors a été dessiné et construit en 1922 par George de Bothezat. Il était destiné à l'armée américaine. Cet immense aéronef pesait 1600 kilogrammes et était propulsé par un seul moteur. Malgré le succès de son vol, les performances de cet appareil n'ont pas été celles attendues par l'armée et par conséquent, le projet a été abandonné (figure 1.4).



**Fig. 1.4** Hélicoptère à quatre rotors de *Bothezat*. Photo: NPGallery Search (nps.gov)

Depuis quelques années, ce même concept a commencé à être exploité, mais avec une taille réduite. Depuis 2002, plusieurs centres de recherche et de groupes de spécialistes en

aéronautique ont commencé à publier les premiers résultats à propos de la modélisation et de la commande de cet hélicoptère à quatre rotors.

L'hélicoptère à quatre rotors est souvent appelé **quadrirotor** ou encore **quadrocopter**. Il possède certains avantages par rapport aux hélicoptères conventionnels. En raison de sa symétrie, ce véhicule est dynamiquement élégant, simple à dessiner et à construire. En fait, il est plus aisé de réaliser un vol stationnaire avec quatre forces de poussées opérant à une même distance du centre de masse qu'avec une seule force de poussée agissant sur le centre de masse. De plus, les pales peuvent être protégées par un carénage, ce qui rend leur utilisation plus sûre que celle des hélicoptères conventionnels, qui sont dangereux à utiliser dans des espaces réduits à cause des pales tournantes non protégées.

La miniaturisation des capteurs, des composants électroniques, l'augmentation des capacités de traitement des calculateurs et le progrès réalisé dans le développement des batteries à polymères de nouvelle génération rendent aujourd'hui possible l'implantation de systèmes embarqués, miniaturisés et autonomes (fig 1.5). Cette plateforme est idéale pour comprendre certains concepts liés au vol de ce type d'aéronefs. Bien que le constructeur mentionne que cet aéronef soit beaucoup plus facile à piloter qu'un hélicoptère conventionnel.



**Fig. 1.5** Drone à quatre rotors. Photo : Julia Hautojärvi

### 1.5 Etat de l'art sur la commande des quadrirotors

Les drones possèdent quelques caractéristiques spécifiques qui rendent difficile la conception d'algorithmes de commande. En effet, ces systèmes sont des systèmes sous actionnés sensibles aux perturbations aérodynamiques et leur dynamique est fortement non linéaire et présente un couplage important entre les différentes variables d'états et les entrées. Afin de diminuer ces effets indésirables, le développement des stratégies de commandes des véhicules aériens autonomes a fait l'objet de plusieurs centaines de publications au cours des

dernières décennies. A cet effet, cette section est destinée à présenter les architectures de commande les plus utilisées dans les drones quadrirotor.

### 1.5.1. Commandes linéaires

Les commandes linéaires tels que : le contrôle quadratique linéaire LQR, le contrôle PD ou PID, l'algorithme  $H_{\text{inf}}$  et l'ordonnancement du gain sont les techniques de contrôle les plus couramment et conventionnellement appliquées. Bouabdallah et al. [4][5] ont étendu leur travail sur le projet OS4 où une étude comparative entre les régulateurs classiques PID et les régulateurs optimaux LQ adaptatifs a été fait. Ils ont constaté que ce dernier a donné des résultats moyens dus aux imperfections de modélisation.

Stepaniak [6] a présenté et discuter en détail la modélisation et l'identification de ce type de systèmes, mais sans prendre en considération l'aspect de leurs commandes. On trouve aussi dans ce travail des directives sur l'implantation pratiques de ce dernier sur les quadrirotors.

Tayebi et McGilvray[7] ont réalisé une étude approfondie et avancée sur les techniques de contrôle et de stabilisation de l'attitude des quadrirotors. Les résultats expérimentaux ont été obtenus à partir d'une version modifiée de drone Draganflyer III.

Une comparaison intéressante entre quatre techniques de commande : LQR, LQR avec gain programmé, linéarisation par retour d'états et mode glissant, a été faite par Kim et al. [8]. Où ils ont constaté d'une manière expérimentale que la commande LQR avec gain programmé était la plus robuste vis-à-vis de la présence des incertitudes de modélisation. Tandis que, pour un système modélisé avec plus de précision, une meilleure performance a été obtenue avec l'approche du mode glissant.

Dans les travaux de Pounds et al. [9], un modèle général est employé en prenant en compte quelques effets ignorés précédemment tel que le battement des pales. Ainsi, comme continuité à leur travail, les auteurs ont utilisé le benchmark X-4 flyer Mark II, avec une commande discrète de type PID [10][11].

### 1.5.2. Commande à Mode Glissant (Sliding Mode Control SMC)

C'est une technique de commande de commutation où les états du système sont amenés vers une surface de glissement [12][13]. Dans les travaux de [14], une commande en mode

glissant pour stabiliser les sous-systèmes sous-actionné du quadrirotor à l'aide d'un contrôleur PID, a été proposée. Ils ont validé la robustesse du contrôleur en traitant des incertitudes paramétriques.

Swamp [15] a introduit un contrôle de mode glissant de second ordre conçu sur la base de la théorie de Lyapunov pour stabiliser le quadrirotor. Ce contrôleur a démontré des résultats prometteurs par rapport au mode glissant conventionnel et a assuré également la robustesse.

La technique SMC a suscité une grande attention pour la conception de contrôleur robuste dans la non-linéarité d'ordre élevé vis-à-vis des perturbations et des incertitudes paramétriques. Cependant, il peut provoquer un problème de vibration qui se produit en raison de la commutation continue du modèle contrôlé. En conséquence, cela peut provoquer une perte d'énergie, une dynamique non modélisée et une instabilité du système parfois dangereuse pour le système [16][17].

### **1.5.3. Commande adaptative**

Ce type de commande offre un mécanisme de réglage paramétrique pour contrôler un système. Le contrôle adaptatif est largement appliqué dans le quadrirotor, et la plupart des cas, il est couplé à d'autres contrôleurs comme le réseau de neurones [18-20], SMC [21-23], le contrôle flou [24], le backstepping [25-28], et la commande LQR [29]. Koshkouei et Zinober [30] ont combiné plusieurs techniques comme le backstepping adaptatif avec la commande par mode glissant afin d'obtenir leurs avantages.

### **1.5.4. Commande intelligente**

On trouve dans la littérature des quadrirotors des commandes basées sur l'intelligence artificielle. Les méthodes de commande par les réseaux de neurone ont été proposées dans de nombreux articles [31][32][33]. Ce type de commande doit d'abord passer par une étape d'apprentissage en temps différé. Cette méthode a prouvé sa robustesse contre les bruits et les perturbations. Dans [31], la disponibilité d'un contrôleur neuronal a été mathématiquement prouvée, et plus tard, les résultats de la simulation ont été révélés dans [32]. Un contrôleur neural adaptatif formant le réseau neuronal en ligne a été introduit dans [33] où la stabilité a été confirmée par des résultats de simulation.

La commande floue peut être considérée comme l'un des contrôleurs les plus appropriés pour un système non linéaire avec incertitudes [34-40]. Le contrôleur flou a suscité un grand intérêt parmi les chercheurs car il offre la solution la mieux adaptée lorsque le système est complexe, mal défini et incertain. La partie la plus intéressante de ce contrôleur est sa mise en œuvre des connaissances des experts linguistiquement et imiter le raisonnement humain pour simplifier tout système complexe. Néanmoins, il n'est pas facile de concevoir la règle de contrôle linguistique et d'analyser la stabilité du système. De plus, il nécessite beaucoup de temps pour le réglage des paramètres.

### 1.6 Modélisation de quadrirotor

La modélisation des quadrirotors a été intensivement étudiée ces dernières années. Cette étape est considérée comme le premier pas pour élaborer les lois de commande. Le modèle dynamique qui sera obtenu représentera d'une part la relation entre les forces et les couples aérodynamiques provoqués par la rotation des rotors et l'engin, et d'autre part des accélérations de translation et de rotation du centre de masse du quadrirotor. Pour les drones on peut distinguer deux classes de modélisation : modélisation par la méthode d'Euler-Lagrange [10][1] et modélisation par la méthode de Newton-Euler [12][1].

#### 1.6.1 Fonctionnement et Concepts de base

Les moteurs arrière et avant ( $M_1, M_3$ ) tournent dans le sens contraire des aiguilles d'une montre lorsque les moteurs droit et gauche ( $M_2, M_4$ ) tournent dans le sens des aiguilles d'une montre. Chaque actionneur produit une force  $F_i$  parallèle à son axe de rotation, ainsi qu'un couple résistant  $\Gamma_i$  opposé au sens de rotation. Figure 1.8.

La force totale (poussée totale) exercée sur l'hélicoptère (parallèle à l'axe  $z_b$ ) est la somme des quatre forces générées par chaque actionneur ( $F_T = F_1 + F_2 + F_3 + F_4$ ).

##### a) Mouvements angulaires

La combinaison des forces  $F_i$  et des couples résistants  $\Gamma_i$  donne les mouvements angulaires autour des axes principaux de l'hélicoptère :

**Mouvement de tangage ( $\theta$ )** : ce mouvement est assuré par la différence des forces ( $F_1, F_3$ ) produites par les actionneurs avant et arrière. Cette différence de forces produit un couple  $\Gamma_\theta$  autour de l'axe  $y_B$

**Mouvement de roulis ( $\phi$ )** : ce mouvement est assuré par la différence des forces ( $F_2, F_4$ ) produites par les actionneurs droit et gauche. Cette différence de forces produit un couple  $\Gamma_\phi$  autour de l'axe  $x_b$

**Mouvement de lacet ( $\psi$ )** : ce mouvement est assuré par la somme des couples de trainée ( $D_i$ ) produits par les quatre actionneurs. Etant donné que les sens de rotation des actionneurs ( $M_1, M_3$ ) et ( $M_2, M_4$ ) sont opposés, nous pouvons régler la somme des quatre couples résistants. Quand les quatre rotors tournent à la même vitesse, ils sont soumis au même couple résistant dont la somme est nulle. Par conséquent, il n'y a pas de rotation autour de l'axe  $z_e$ . Par contre, si nous provoquons une différence de vitesse entre les moteurs tournant en sens opposé, les couples résistants provoquent un couple  $\Gamma_\psi$  autour de l'axe  $z_e$ , provoquant ainsi la rotation de l'engin. Notons que la force de poussée  $F_T$  est celle qui garantit la sustentation de l'hélicoptère. Néanmoins, lorsque l'hélicoptère effectue une inclinaison ( $\theta \neq 0$  & ou  $\phi \neq 0$ ), cette force donne lieu à une translation horizontale.

En fait, l'hélicoptère à quatre rotors est un système mécanique sous-actionné, i.e. le déplacement horizontal dépend directement de la position angulaire.

### b) Types de vols

De manière générale et d'après la description des mouvements angulaires, nous pouvons classer les principaux vols en trois types:

**Vol stationnaire** : l'hélicoptère peut rester en vol stationnaire à une certaine hauteur constante par rapport au sol, en ayant une vitesse de translation nulle. Il est en montée verticale après avoir franchi le seuil définissant l'effet du sol

**Translations verticales** : elles sont définies lorsque l'hélicoptère se déplace suivant l'axe  $z_e$ . En l'absence de perturbations, la force de poussée  $F_T$  est toujours verticale et en montée elle est toujours supérieure au poids de l'hélicoptère ( $F_T > mg$ ) tandis qu'en descente elle est inférieure au poids ( $F_T < mg$ ).

**Translations Horizontales** : elles sont définies de façon similaire aux translations verticales, mais cette fois-ci dans le plan  $x_e y_e$ . Lorsqu'une translation est effectuée suivant la direction  $x_e$  et la force de poussée maintient le système à une hauteur constante par rapport au sol, le système effectue un vol connu sous le nom de "vol en palier".

D'après la description des mouvements angulaires et des déplacements verticaux et horizontaux, nous pouvons remarquer que le déplacement de l'engin dépend directement de son orientation (attitude). De ce fait, le déplacement peut être commandé de façon performante selon l'orientation en donnant origine à une commande semi-automatique de la position.

### 1.6.2 Etude cinématique

Comme le quadrirotor est considéré comme un corps rigide, nous définissons deux référentiels : un référentiel terrestre fixe  $E(o_E, x_e, y_e, z_e)$  et un référentiel mobile fixé au centre de masse du quadrirotor  $B(o_B, x_b, y_b, z_b)$ . Le référentiel  $E$  est utilisé pour définir la position linéaire ( $\Gamma^E$  [m]) et la position angulaire ( $\Theta^E$  [rad]) du quadrirotor. Le référentiel  $B$  est utilisé pour définir la vitesse linéaire ( $V^B$  [ms-1]), la vitesse angulaire ( $\omega^B$  [rad s-1]), les forces ( $F^B$  [N]) et les couples ( $\tau^B$  [Nm]).

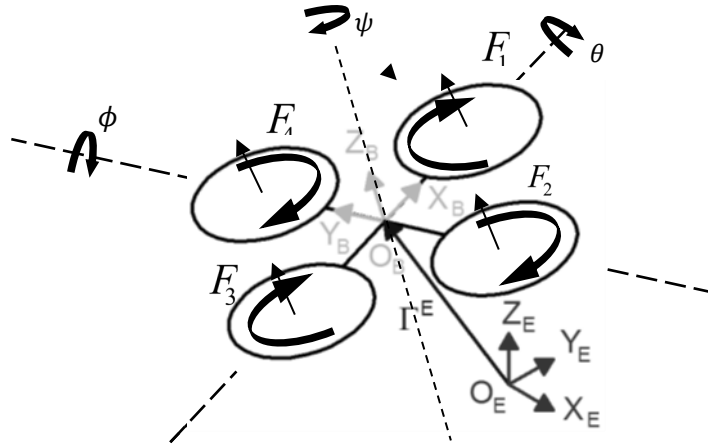
La position linéaire  $\Gamma^E$  de l'hélicoptère est déterminée par les coordonnées du vecteur entre l'origine du référentiel B et l'origine du référentiel E par rapport au référentiel E conformément à l'équation (1.1), ce qui est montré sur la figure (Fig. 1.6)

$$\Gamma^E = [X \quad Y \quad Z] \tag{1.1}$$

La position angulaire (ou de l'attitude)  $\Theta^E$  de l'hélicoptère est définie par l'orientation du référentiel B par rapport au référentiel E. Ceci est donné par trois rotations consécutives sur les grands axes qui tiennent la référentiel E vers le référentiel B. Dans ce travail, l'ensemble «roll-tangage-lacet» des angles d'Euler ont été utilisés. L'équation (1.2) montre le vecteur d'attitude :

$$\Theta^E = [\phi \quad \theta \quad \psi]^T \tag{1.2}$$

Pour le passage entre les deux référentiels, la communauté d'aéronautique utilise généralement la rotation premièrement autour de l'axe  $z_B$ , ensuite autour de l'axe  $y_B$  et finalement autour de l'axe  $x_B$ . Ces rotations sont exprimées respectivement par trois matrices de rotation  $R(x_B, \phi)$ ,  $R(y_B, \theta)$  et  $R(z_B, \psi)$  :



**Fig. 1.6** Fonctionnement de l'hélicoptère à quatre rotors  
Par rapport aux Référentiels terrestre et mobile

$$R(x_B, \phi) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & c_\phi & -s_\phi \\ 0 & s_\phi & c_\phi \end{pmatrix} \quad R(y_B, \theta) = \begin{pmatrix} c_\theta & 0 & s_\theta \\ 0 & 1 & 0 \\ -s_\theta & 0 & c_\theta \end{pmatrix} \quad R(z_B, \psi) = \begin{pmatrix} c_\psi & -s_\psi & 0 \\ s_\psi & c_\psi & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Le passage du référentiel fixe E vers la référentiel B se fait à travers la matrice de rotation totale

$R_\Theta = R_\Theta(\phi, \theta, \psi) = R(z_B, \psi)R(y_B, \theta)R(x_B, \phi)$  :

$$R_\Theta = \begin{pmatrix} c_\psi c_\theta & c_\psi s_\theta s_\phi - s_\psi c_\phi & c_\psi s_\theta c_\phi + s_\psi s_\phi \\ s_\psi c_\theta & s_\psi s_\theta s_\phi + c_\psi c_\phi & s_\psi s_\theta c_\phi - c_\psi s_\phi \\ -s_\theta & c_\theta s_\phi & c_\theta c_\phi \end{pmatrix} \quad (1.3)$$

où  $c_\theta$  et  $s_\theta$  désignent  $\cos \theta$  et  $\sin \theta$ , respectivement.

### 1.6.3 Forces aérodynamiques et couples externes

Dans cette section nous présentons les forces aérodynamiques et les couples externes qui sont appliqués sur le quadrirotor. Pour les forces, nous avons le poids du quadrirotor plus les forces produites par la rotation des rotors. La vitesse de rotation des rotors génère des forces aérodynamiques pour décoller, tourner et faire atterrir le quadrirotor. Les qualités aérodynamiques du rotor sont déterminées par le profil des pales. Ce profil est une coupe de la pale perpendiculaire à l'axe de la pale. La force aérodynamique peut être décomposée en deux forces projetées suivant deux axes :

- La traînée  $D_i$  parallèle au plan de rotation de la pale.
- La portance  $F_i$  suivant la perpendiculaire du plan de rotation de la pale (figure 1.8).

La traînée correspond à la résistance à l'avancement dans l'air (à cause de l'accélération de la masse d'air soumise à la rotation des pales), et grâce à la portance que le quadrirotor compense le poids, décolle et se maintient dans l'air.

Le calcul de la portance d'un rotor constitue la première étape dans le processus d'estimation des efforts agissant sur la structure de l'engin. D'après [11], dans les conditions proches du vol stationnaire, ou le pas collectif de chaque rotor est fixé à une valeur constante, d'où  $F_i$  s'écrira comme suit :

$$F_i = b \Omega_i^2 \quad (1.5)$$

Avec  $b$  est appelée facteur de la portance.

On définit  $U_1$  la portance totale provoquée par les quatre moteurs est définit comme :

$$U_1 = \sum_{i=1}^4 F_i = b(\Omega_1^2 + \Omega_2^2 + \Omega_3^2 + \Omega_4^2) \quad (1.6)$$

On peut écrire cette force dans la référentiel B comme :

$$\mathbf{F}^B = \begin{bmatrix} 0_{2 \times 1} \\ b(\Omega_1^2 + \Omega_2^2 + \Omega_3^2 + \Omega_4^2) \end{bmatrix} \quad (1.7)$$

Pour la traînée  $D_i$ , [11] a proposé la forme suivante :

$$D_i = C_d \rho \Omega_i^2 = k_d \Omega_i^2 \quad (1.8)$$

Où  $C_d$  représente le coefficient de traînée et  $\rho$  représente la masse volumique de l'air. L'action de l'air sur chaque rotor, crée un couple de contre-rotation agissant suivant l'axe de pivot de chaque rotor. Dans notre étude, le pas collectif est fixe, l'expression de couple-résistant s'écrit [11][1] :

$$\Gamma_i = d \Omega_i^2 \quad (1.11)$$

où  $d$  est une constante positive appelée facteur de traînée.

## 1.6.4 Mouvements angulaires

L'étude des portances et traînées de chaque rotor nous aide à identifier les couples actifs et par conséquent les mouvements de rotation du quadrirotor dans le référentiel mobile  $B(o_B, x_B, y_B, z_B)$  (figure 1.7):

La différence entre les portances gauche et droite produit le couple  $\Gamma_\phi$  responsable du mouvement de roulis

$$\Gamma_\phi = l(F_4 - F_2) = k_l l(\Omega_4^2 - \Omega_2^2) = bl(\Omega_4^2 - \Omega_2^2) \quad (1.12)$$

Le mouvement de tangage est produit par le couple  $\Gamma_\theta$  causé par la dissymétrie entre les portances produites par le rotor avant et arrière :

$$\Gamma_\theta = l(F_3 - F_1) = k_l l(\Omega_3^2 - \Omega_1^2) = bl(\Omega_3^2 - \Omega_1^2) \quad (1.13)$$

Le mouvement de lacet  $\Gamma_\psi$  est le résultat de la somme des couples résistants générés par les traînées  $D_i, i = 1, \dots, 4$  autour du centre de l'engin puisque l'axe de rotation des pales est loin du centre du quadrirotor :

$$\Gamma_\psi = \Gamma_2 + \Gamma_4 - \Gamma_1 - \Gamma_3 = d(\Omega_2^3 + \Omega_4^3 - \Omega_1^3 - \Omega_3^3) \quad (1.14)$$

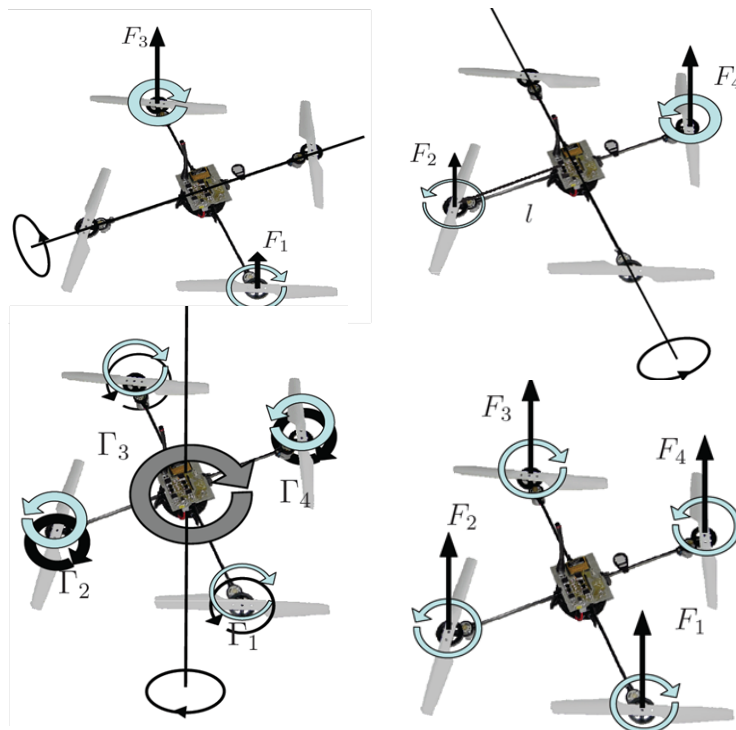


Fig. 1.7 a) Mouvement de tangage du quadrirotor b) Mouvement de roulis du quadrirotor

Ces trois couples forment le couple actif  $\Gamma_a$  :

$$\Gamma_a = \begin{bmatrix} \Gamma_\phi(\Omega_2, \Omega_4) \\ \Gamma_\theta(\Omega_1, \Omega_3) \\ \Gamma_\psi(\Omega_1, \Omega_2, \Omega_3, \Omega_4) \end{bmatrix} = \begin{pmatrix} U_2 \\ U_3 \\ U_4 \end{pmatrix} \quad (1.15)$$

### 1.6.5 Etude Dynamique et approche appliquée

Dans cette section nous présentons la dérivation des équations du modèle spécifique à l'architecture du quadrirotor à partir du formalisme de Newton-Euler. Deux référentiels doivent être définis : Le référentiel inertiel de masse (référentiel E) et Le référentiel du corps fixe (référentiel B).

La cinématique d'un corps rigide générique à 6 ddl est décrite par l'équation (1.17).

$$\dot{\xi} = J_\theta v \quad (1.17)$$

où  $\dot{\xi}$  [...] est le vecteur de vitesse généralisée par rapport au référentiel E.

$v$  [...] est le vecteur vitesse par rapport au référentiel B

$J_\theta$  [-] est la matrice généralisée.

$\xi$  [...] est composé de la position du quadrirotor linéaire  $\Gamma^E$  [m] et la position angulaire  $\Theta^E$  [rad] par rapport au référentiel E comme indiqué dans l'équation (1.18).

$$\xi = [\Gamma^E \ \Theta^E]^T = [X \ Y \ Z \ \phi \ \theta \ \psi]^T \quad (1.18)$$

De même  $v$  [...] est composé de la vitesse linéaire du quadrirotor linéaire  $V^B$  [ $ms^{-1}$ ] et la vitesse angulaire  $\omega^B$  [ $rads^{-1}$ ] par rapport à B comme indiqué dans l'équation (1.19).

$$v = [V^B \ \omega^B]^T = [u \ v \ w \ p \ q \ r]^T \quad (1.19)$$

Le vecteur  $\omega^B = (p, q, r)^T$  dans l'équation (1.19) représente le vecteur des vitesses angulaires du quadrirotor dans le référentiel mobile.

Le vecteur  $\omega$  est exprimé par rapport aux vitesses de rotations  $\dot{\phi}, \dot{\theta}, \dot{\psi}$  dans  $E(o_E, x_E, y_E, z_E)$

$$\omega = \begin{pmatrix} \dot{\phi} \\ 0 \\ 0 \end{pmatrix} + \mathbf{R}(x, \phi)^{-1} \begin{pmatrix} 0 \\ \dot{\theta} \\ 0 \end{pmatrix} + (\mathbf{R}(y, \theta)\mathbf{R}(x, \phi))^{-1} \begin{pmatrix} 0 \\ 0 \\ \dot{\psi} \end{pmatrix} \quad (1.20)$$

En plus, la matrice généralisée  $\mathbf{J}_\Theta$  est composée de 4 sous-matrices selon l'équation (1.21).

$$\mathbf{J}_\Theta = \begin{bmatrix} \mathbf{R}_\Theta & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & \mathbf{T}_\Theta \end{bmatrix} \quad (1.21)$$

Tandis que, la matrice de transfert  $\mathbf{T}_\Theta$  est défini selon les équations (1.22).

$$\mathbf{T}_\Theta = \begin{bmatrix} 1 & s_\phi t_\theta & c_\phi t_\theta \\ 0 & c_\phi & -s_\phi \\ 0 & \frac{s_\phi}{c_\theta} & \frac{c_\phi}{c_\theta} \end{bmatrix} \quad (1.22)$$

De même, cette notation sera adoptée par la suite :  $c_k = \cos k$ ,  $s_k = \sin k$  et  $t_k = \tan k$ .

La dynamique d'un corps rigide générique à 6ddl tient compte de la masse du corps  $m[\text{kg}]$  et sa matrice d'inertie  $I [\text{N ms}^2]$ . La dynamique par rapport au B est décrite par l'équation (1.23).

$$\begin{bmatrix} m \mathbf{I}_{3 \times 3} & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & I \end{bmatrix} \begin{bmatrix} \dot{\mathbf{V}}^B \\ \dot{\boldsymbol{\omega}}^B \end{bmatrix} + \begin{bmatrix} \boldsymbol{\omega}^B \times (m \mathbf{V}^B) \\ \boldsymbol{\omega}^B \times (I \boldsymbol{\omega}^B) \end{bmatrix} = \begin{bmatrix} \mathbf{F}^B \\ \boldsymbol{\tau}^B \end{bmatrix} \quad (1.23)$$

$\mathbf{I}_{3 \times 3}$  c'est la matrice identité.  $\dot{\mathbf{V}}_B [\text{ms}^{-2}]$  est le vecteur accélération linéaire du quadrirotor, tandis que  $\dot{\boldsymbol{\omega}}^B [\text{rads}^{-2}]$  est le vecteur d'accélération angulaire du quadrirotor. En outre,  $\mathbf{F}^B [\text{N}]$  est le vecteur des forces du quadrirotor et  $\boldsymbol{\tau}^B [\text{Nm}]$  est le vecteur des couples du quadrirotor.

Deux hypothèses ont été prises dans ce formalisme :

- l'origine du corps  $O_B$  du châssis fixe est confondue avec le centre de masse du corps COM.
- les axes de la référentiel B coïncident avec les axes principaux d'inertie du corps. Dans ce cas, la matrice d'inertie  $I$  est diagonale.

A partir du vecteur de force généralisée  $\Lambda$  qui est défini selon l'équation (1.24).

$$\mathbf{A} = [\mathbf{F}^B \ \boldsymbol{\tau}^B]^T = [F_x \ F_y \ F_z \ \tau_x \ \tau_y \ \tau_z]^T \quad (1.24)$$

L'équation (1.23) peut être réécrite sous la forme matricielle suivante :

$$\mathbf{M}_B \dot{\mathbf{v}} + \mathbf{C}_B(\mathbf{v})\mathbf{v} = \mathbf{A} \quad (1.25)$$

$\mathbf{M}_B$  est la matrice d'inertie du système et  $\mathbf{C}_B(\mathbf{v})$  est la matrice de Coriolis-centripète, les deux sont représentées par rapport au référentiel B (1.26) et (1.27) :

$$M_B = \begin{bmatrix} m\mathbf{I}_{3 \times 3} & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & \mathbf{I} \end{bmatrix} \quad (1.26)$$

$$C_B(\mathbf{v}) = \begin{bmatrix} \mathbf{0}_{3 \times 3} & -m\mathbf{S}(\mathbf{V}^B)_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & -\mathbf{S}(\mathbf{I}\boldsymbol{\omega}^B) \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & m w & -m v \\ 0 & 0 & 0 & -m w & 0 & m u \\ 0 & 0 & 0 & m v & -m u & 0 \\ 0 & 0 & 0 & 0 & I_{ZZ} r & -I_{YY} q \\ 0 & 0 & 0 & -I_{ZZ} r & 0 & I_{XX} p \\ 0 & 0 & 0 & I_{YY} q & -I_{XX} p & 0 \end{bmatrix} \quad (1.27)$$

Dans l'équation (1.27) l'opérateur antisymétrique  $\mathbf{S}(\cdot)$  (1.28) est adopté.

$$\mathbf{S}(\boldsymbol{\kappa}) = -\mathbf{S}^T(\mathbf{k}) = \begin{bmatrix} 0 & -k_3 & k_1 \\ k_3 & 0 & -k_1 \\ -k_2 & k_1 & 0 \end{bmatrix} \quad \mathbf{k} = \begin{bmatrix} k_1 \\ k_2 \\ k_3 \end{bmatrix} \quad (1.28)$$

Suite au vecteur des forces généralisées et selon la nature du quadrirotor trois éléments contribuent dans la dynamique de système :

**Vecteur de gravitation  $\mathbf{G}_B(\boldsymbol{\xi})$ .** Il est obtenu à partir de l'accélération due à la pesanteur  $g[m s^{-2}]$ . L'équation (1.29) montre les transformations pour obtenir  $\mathbf{G}_B(\boldsymbol{\xi})$ .

$$\mathbf{G}_B(\boldsymbol{\xi}) = \begin{bmatrix} \mathbf{F}_G^B \\ \mathbf{0}_{3 \times 1} \end{bmatrix} = \begin{bmatrix} \mathbf{R}_\Theta^{-1} \mathbf{F}_G^E \\ \mathbf{0}_{3 \times 1} \end{bmatrix} = \begin{bmatrix} \mathbf{R}_\Theta^T \begin{bmatrix} 0 \\ 0 \\ -mg \end{bmatrix} \\ \mathbf{0}_{3 \times 1} \end{bmatrix}$$

Où  $F_G^B$  [N] est le vecteur de force gravitationnelle par rapport au référentiel B et  $F_G^E$  [N] par rapport au référentiel E.

**Effets gyroscopiques :** ils sont produits par la rotation de l'hélice, dont deux tournants dans le sens horaire et les deux autres tournants dans le sens antihoraire, ce qui provoque un déséquilibre global lorsque la somme algébrique des vitesses de rotation n'est pas égale à zéro. Par conséquent, si les taux de roulis ou de tangage sont également différents de zéro, le quadrirotor éprouve un couple gyroscopique selon l'équation (1.30).

$$\mathbf{O}_B(v)\boldsymbol{\Omega} = \left[ \begin{array}{c} \mathbf{0}_{3 \times 1} \\ -\sum_{k=1}^4 J_{TP} \left( \omega^B \times \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \right) (-1)^k \Omega_k \end{array} \right] = \left[ \begin{array}{c} \mathbf{0}_{3 \times 1} \\ J_{TP} \begin{bmatrix} -q \\ p \\ 0 \end{bmatrix} \end{array} \right] \boldsymbol{\Omega} = J_{TP} \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ q & -q & q & -q \\ -p & p & -p & p \\ 0 & 0 & 0 & 0 \end{bmatrix} \boldsymbol{\Omega} \quad (1.30)$$

$\mathbf{O}_B(v)$  est la matrice gyroscopique et  $J_{TP}[Nms^2]$  est le moment de rotation totale d'inertie autour de l'axe de l'hélice.

L'équation (1.31) définit la vitesse des hélices  $\Omega$  [rads<sup>-1</sup>] en général et le vecteur vitesse des hélices  $\boldsymbol{\Omega}$  [rads<sup>-1</sup>] utilisé dans l'équation (1.30).

$$\Omega = -\Omega_1 + \Omega_2 - \Omega_3 + \Omega_4 \quad \boldsymbol{\Omega} = [\Omega_1 \quad \Omega_2 \quad \Omega_3 \quad \Omega_4]^T \quad (1.31)$$

Où  $\Omega_1$  est la vitesse de l'hélice avant,  $\Omega_2$  est la vitesse de l'hélice à droite,  $\Omega_3$  est la vitesse de l'hélice arrière et  $\Omega_4$  est la vitesse de l'hélice gauche.

**Vecteur de mouvement** il représente les forces et les couples directement produits par les entrées principales du mouvement (1.32)

$$\mathbf{U}_B(\boldsymbol{\Omega}) = \mathbf{E}_B \boldsymbol{\Omega}^2 = [\mathbf{F}_B \quad \boldsymbol{\Gamma}_B]^T = \begin{bmatrix} 0 \\ 0 \\ U_1 \\ U_2 \\ U_3 \\ U_4 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ b(\Omega_1^2 + \Omega_2^2 + \Omega_3^2 + \Omega_4^2) \\ bl(\Omega_4^2 - \Omega_2^2) \\ bl(\Omega_3^2 - \Omega_1^2) \\ d(\Omega_2^3 + \Omega_4^3 - \Omega_1^3 - \Omega_3^3) \end{bmatrix} \quad (1.32)$$

L'équation (1.32) montre l'action du vecteur de commande sur la dynamique de l'hélicoptère quadrirotor. Avec  $l$  la distance entre le centre du quadrirotor et le centre de l'hélice.  $U_1, U_2, U_3$  et  $U_4$  sont les composantes du vecteur de commande introduites dans (§ 1.6.4). Il est possible d'identifier une matrice de mouvement  $\mathbf{E}_B$  comme suit :

$$\mathbf{E}_B = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ b & b & b & b \\ 0 & -bl & 0 & bl \\ -bl & 0 & bl & 0 \\ -d & d & -d & d \end{bmatrix} \quad (1.33)$$

De l'équation (1.25), il est possible de décrire le modèle de la dynamique du quadrirotor compte tenu de ces trois derniers éléments, on obtient l'équation (1.34).

$$\mathbf{M}_B \dot{\mathbf{v}} + \mathbf{C}_B(\mathbf{v})\mathbf{v} = \mathbf{G}_B(\xi) + \mathbf{O}_B(\mathbf{v})\Omega + \mathbf{E}_B\Omega^2 \quad (1.34)$$

### Présentation du système d'états

La dynamique du quadrirotor est exprimée par l'équation (1.34), le réarrangement de cette équation a permis d'obtenir le vecteur d'accélération par rapport au référentiel B :

$$\dot{\mathbf{v}} = \mathbf{M}_B^{-1}(-\mathbf{C}_B(\mathbf{v})\mathbf{v} + \mathbf{G}_B(\xi) + \mathbf{O}_B(\mathbf{v})\Omega + \mathbf{E}_B\Omega^2) \quad (1.35)$$

L'expression (1.36) sur la forme d'un système d'équations. Avec les entrées de vitesse sont données par l'équation (1.37).

$$\begin{cases} \dot{u} = (v r - w q) + g s_\theta \\ \dot{v} = (w p - u r) - g c_\theta s_\phi \\ \dot{w} = (u q - v p) - g c_\theta c_\phi + \frac{U_1}{m} \\ \dot{p} = \frac{I_{YY} - I_{ZZ}}{I_{XX}} qr - \frac{J_{TP}}{I_{XX}} q \Omega + \frac{U_2}{I_{XX}} \\ \dot{q} = \frac{I_{ZZ} - I_{XX}}{I_{YY}} pr + \frac{J_{TP}}{I_{YY}} p \Omega + \frac{U_3}{I_{YY}} \\ \dot{r} = \frac{I_{XX} - I_{YY}}{I_{ZZ}} pq + \frac{U_4}{I_{ZZ}} \end{cases} \quad (1.36)$$

$$\begin{aligned} \Omega &= -\Omega_1 + \Omega_2 - \Omega_3 + \Omega_4 \\ U_1 &= b(\Omega_1^2 + \Omega_2^2 + \Omega_3^2 + \Omega_4^2) & U_3 &= lb(-\Omega_1^2 + \Omega_3^2) \\ U_2 &= lb(-\Omega_2^2 + \Omega_4^2) & U_4 &= d(-\Omega_1^2 + \Omega_2^2 - \Omega_3^2 + \Omega_4^2) \end{aligned} \quad (1.37)$$

Toutefois, il peut être utile d'exprimer la dynamique par rapport à un référentiel hybride (appelé H) composé d'équations linéaires par rapport au référentiel E et les équations angulaires par

rapport au référentiel B. Ce référentiel H est adopté, car il est facile de combiner la dynamique avec la commande.

L'équation (1.38) montre le vecteur vitesse généralisée du quadrirotor par rapport au référentiel H.

$$\zeta = [\dot{\Gamma}^E \ \omega^B]^T = [\dot{X} \ \dot{Y} \ \dot{Z} \ p \ q \ r]^T \quad (3.38)$$

La dynamique du système peut être réécrite sous forme de matrice selon l'équation suivante :

$$M_H \dot{\zeta} + C_H(\zeta) \zeta = G_H + O_H(\zeta) \Omega + E_H(\zeta) \Omega^2 \quad (1.39)$$

où  $\dot{\zeta}$  est un vecteur d'accélération généralisé du quadrirotor par rapport au référentiel H. Il suit maintenant les définitions de toutes les matrices et les vecteurs utilisés dans l'équation (1.39). Le système de la matrice d'inertie par rapport au référentiel H,  $M_H = M_B$

$$M_H = \begin{bmatrix} m \mathbf{I}_{3 \times 3} & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & \mathbf{I} \end{bmatrix} \quad (1.40)$$

Par contre la matrice de Coriolis  $C_H(\zeta)$  est différent à  $C_B(\zeta)$  et est défini selon l'équation (1.41).

$$C_H(\zeta) = \begin{bmatrix} \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & -S(\mathbf{I} \omega^B) \end{bmatrix} \quad (1.41)$$

avec, le vecteur de gravitation  $G_H$  est défini dans l'équation (1.42).

$$G_H = \begin{bmatrix} \mathbf{F}_G^E \\ \mathbf{0}_{3 \times 1} \end{bmatrix} = \begin{bmatrix} \mathbf{0}_{1 \times 2} \\ -mg \\ \mathbf{0}_{1 \times 3} \end{bmatrix} \quad (1.42)$$

Les effets gyroscopiques produits par la rotation de l'hélice sont invariables et la matrice gyroscopique  $O_H(\zeta)$  est définie par :

$$O_H(\zeta) \Omega = O_B(v) \Omega = \begin{bmatrix} \mathbf{0}_{3 \times 1} \\ J_{TP} \begin{bmatrix} -q \\ p \\ 0 \end{bmatrix} \end{bmatrix} \Omega = J_{TP} \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ q & -q & q & -q \\ -p & p & -p & p \\ 0 & 0 & 0 & 0 \end{bmatrix} \Omega \quad (1.43)$$

La matrice de mouvement par rapport au référentiel H.  $E_H(\zeta)$  :

$$\mathbf{E}_H(\xi) \boldsymbol{\Omega}^2 = \begin{bmatrix} \mathbf{R}_\Theta & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & \mathbf{I}_{3 \times 3} \end{bmatrix} \mathbf{E}_B \boldsymbol{\Omega}^2 = \begin{bmatrix} (s\psi s\phi + c\psi s\theta c\phi)U_1 \\ (-c\psi s\phi + s\psi s\theta c\phi)U_1 \\ (c\phi c\theta)U_1 \\ U_2 \\ U_3 \\ U_4 \end{bmatrix} \quad (1.44)$$

En réarrangeant l'équation (1.39), on obtient :

$$\dot{\xi} = \mathbf{M}_H^{-1}(-\mathbf{C}_H(\xi)\xi + \mathbf{G}_H + \mathbf{O}_H(\xi)\boldsymbol{\Omega} + \mathbf{E}_H(\xi)\boldsymbol{\Omega}^2) \quad (1.45)$$

La forme matricielle (1.45) réécrite sous forme d'un système d'équations d'état comme suit :

$$\begin{cases} \ddot{X} = (\sin\psi \sin\phi + \cos\psi \sin\theta \cos\phi)U_1/m \\ \ddot{Y} = (-\cos\psi \sin\phi + \sin\psi \sin\theta \cos\phi)U_1/m \\ \ddot{Z} = -\mathbf{g} + (\cos\phi \cos\theta)U_1/m \\ \dot{p} = qr(I_{YY} - I_{ZZ})/I_{XX} - q\Omega J_{TP}/I_{XX} + U_2/I_{XX} \\ \dot{q} = pr((I_{ZZ} - I_{XX})/I_{YY}) + p\Omega(J_{TP}/I_{YY}) + U_3/I_{YY} \\ \dot{r} = pq(I_{XX} - I_{YY})/I_{ZZ} + U_4/I_{ZZ} \end{cases} \quad (1.46)$$

## 1.7 Conclusion

Dans ce chapitre nous avons présenté les définitions, les catégories, les configurations, les applications et les travaux liés à ce domaine, en premier lieu. Ensuite, une description globale des quadrirotors a été présentée où nous avons analysé les forces aérodynamiques et les couples agissant sur la dynamique du quadrirotor. Nous avons montré la dynamique des quadrirotors, les différents mouvements et la façon dont ils parviennent à réaliser les vols stationnaires.

Finalement, nous avons étudié les modèles dynamiques du quadrirotor en utilisant la méthode de Newton-Euler. Ce modèle nous servira par la suite, pour comparer la performance d'un modèle flou et représenter le comportement dynamique non-linéaire entrée-sortie du système.

Nous abordons maintenant dans les sections suivantes, l'identification floues de type Takagi-Sugeno à partir des données. Cette technique est utilisée pour identifier les fonctions et les systèmes non linéaires. Le modèle obtenu est utilisé pour établir une commande adaptative appliquée sur le système Quadrirotor.

## Chapitre 2

# Identification de modèles flous Takagi-Sugeno à partir de données

### 2.1 Introduction

Le développement de modèles mathématiques des systèmes est un sujet central dans plusieurs disciplines des sciences et de l'ingénierie. Traditionnellement, la modélisation est vue comme la double conjonction entre la compréhension de la nature et du comportement d'un système ainsi que le traitement mathématique approprié qui conduisent à l'obtention d'un modèle utilisable. Néanmoins, le besoin d'une forte compréhension des éléments physiques de base constitue une grande restriction au niveau pratique quand on est confronté aux systèmes complexes, les systèmes à phénomène aérodynamique font souvent partie de ces systèmes [1].

Le problème de modélisation consiste alors à proposer la structure appropriée pour l'approximateur et d'estimer les paramètres du modèle, en utilisant habituellement des données entrée-sortie représentatives du comportement, afin de capturer correctement les dynamiques du système. Le fait que les humains puissent souvent gérer différents types de situations complexes en exploitant des informations dont beaucoup sont subjectives et imprécises à stimuler la recherche de paradigmes alternatifs de modélisation et de commande. Ainsi, le concept de la modélisation floue a trouvé ses origines dans la théorie des ensembles flous proposée en 1965 par Zadeh [41], comme une manière de traiter l'incertitude, fondée sur l'idée de définir des ensembles pouvant contenir des éléments de façon graduelle. Cette théorie a introduit une manière de formaliser les méthodes humaines de raisonnement.

Dans le but de faire face à un problème de commande complexe, on peut tirer profit de différents types de connaissance de l'expertise et de nombreuses données entrées-sorties sur le système. Pour prendre en compte toutes ces connaissances, l'approche de conception de systèmes flous basée sur le modèle est, en général, plus systématique du point de vue des performances numériques.

Dans ce chapitre nous présentons les éléments de base d'identification de modèles flous à partir des données entrée-sortie. Nous nous intéressons particulièrement aux modèles de type Takagi-Sugeno et à leur construction en utilisant des techniques de coalescence floue. Une grande partie de ce chapitre est basée sur [42].

### 2.2. Modélisation floue des systèmes

Les modèles flous peuvent être considérés comme des modèles logiques pour établir des relations qualitatives entre les variables du modèle. La nature des modèles flous permet l'utilisation d'information sous la forme des expressions du langage naturel, ce qui facilite la formalisation de la connaissance des experts ainsi que la transparence et l'interprétabilité du modèle. De plus, les modèles flous ont une structure mathématique flexible capable d'approximer un grand nombre de systèmes non linéaires complexes avec une bonne précision [42] [43][44].

#### 2.2.1. Structure générale des modèles flous

En général les systèmes flous s'appuient sur une représentation de la connaissance sous forme de règles « Si-Alors » dont l'expression générique suivante :

*Si antécédent Alors conséquent*

Selon la structure particulière de la proposition conséquent, on peut distinguer trois types de modèles flous basés sur des règles [42] ; modèle flou de type Mamdani, modèle flou relationnel et modèle flou de type Takagi-Sugeno (noté TS), le conséquent utilise des variables numériques plutôt que des variables linguistiques, sous la forme d'une constante, d'un polynôme ou de manière plus générale d'une fonction ou d'une équation différentielle dépendant des variables associées à la proposition antécédent.

Dans le cadre des applications de la logique floue en commande des systèmes, les types de modèles les plus souvent utilisés sont ceux de Mamdani et de Takagi-Sugeno.

Reprenons une règle linguistique  $R_i$  de la forme générale suivante :

$$R_i : \text{Si } x \text{ est } A_i \text{ Alors } y \text{ est } B_i, \quad i = 1, \dots, r \quad (2.1)$$

Où  $r$  dénote le nombre de règles du modèle.  $x \in X \subset \mathfrak{R}^p$  est la variable de l'antécédent, qui représente l'entrée du système et  $y \in Y \subset \mathfrak{R}^q$  est la variable du conséquent, qui représente la

sortie du système flou.  $X$  et  $Y$  correspondent respectivement aux univers de discours des variables d'entrée et de sortie.  $A_i$  et  $B_i$  sont des termes linguistiques qui correspondent aux valeurs qualitatives associées aux variables de base  $x$  et  $y$ .

Ces termes linguistiques sont décrits par des ensembles flous définis par des fonctions d'appartenance  $\mu$ , qui établissent une correspondance de chacun des univers de discours vers l'intervalle  $[0,1]$ , ainsi :

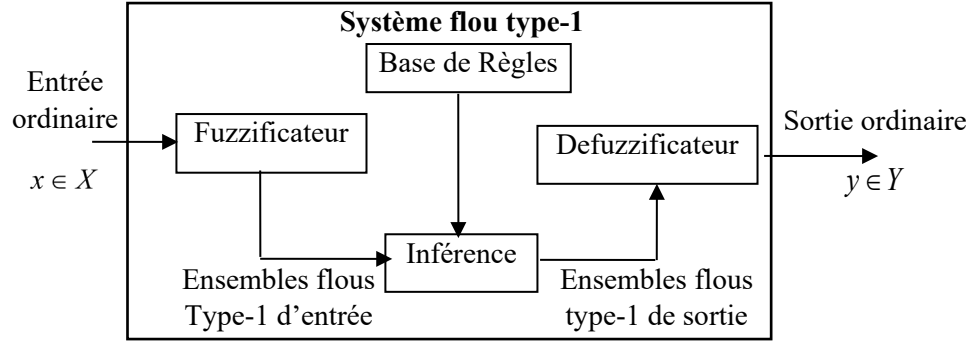
$$\mu_{A_i}(\mathbf{x}): X \rightarrow [0, 1] , \mu_{B_i}(\mathbf{y}): Y \rightarrow [0, 1] \quad (2.2)$$

La théorie des ensembles flous [41][42] permet une appartenance partielle d'un élément à un ensemble. Les ensembles flous  $A_i$  définissent des régions floues dans l'espace de l'antécédent pour lequel la prémisse du conséquent est valable. Les fonctions d'appartenance peuvent être définies par l'expert ou bien par expérimentation. Quand des données entrée-sortie du système sont disponibles, il est alors possible d'appliquer des méthodes pour l'obtention ou l'adaptation des fonctions d'appartenance.

Dans le cadre de la modélisation et la commande de systèmes, le fonctionnement interne des systèmes flous repose sur une structure, représentée par la figure (fig. 2.2), qui inclut :

- La fuzzification des variables d'entrée,
- L'inférence à partir d'une base de connaissance,
- La défuzzification, avec éventuellement un post-traitement d'information.

L'étape de fuzzification consiste à transformer les entrées numériques disponibles en parties floues. Celles-ci alimentent alors, le mécanisme d'inférence qui à partir d'une valeur d'entrée et selon la connaissance fournie par la base de connaissance, détermine la valeur correspondante de la sortie. Cette base de connaissance est composée par une base de règles et une base de données, et stocke les fonctions d'appartenance associées aux termes linguistiques employés dans le système flou. Cela constitue le fondement du raisonnement approximatif du système, car la combinaison des entrées avec les règles floues permet de tirer des conclusions. enfin, la défuzzification est l'opération inverse de la fuzzification, en convertissant les parties floues relatives aux sorties du mécanisme d'inférence en sorties numériques



**Fig. 2.1.** Représentation générale d'un système flou.

L'inférence floue se fait par une collection de plusieurs règles. Dans le cas d'une structure Mamdani, la forme la plus commune de ces règles est la forme dite conjonctive donnée par :

$$R_i: \text{Si } x_1 \text{ est } A_{i1} \text{ et } x_2 \text{ est } A_{i2} \text{ et } \dots \text{ et } x_p \text{ est } A_{ip} \text{ alors } y \text{ est } B_i \quad (2.3)$$

Pour la  $i^{\text{ème}}$  règle  $R_i$  du système, les  $p$  variables  $(x_1, \dots, x_p)$  de l'antécédent sont associées aux ensembles flous  $(A_{i1}, \dots, A_{ip})$  et le conséquent  $y$  est associé à un autre ensemble flou  $B_i$ . Les termes linguistiques représentés par  $A_i$  et  $B_i$  sont définis par des ensembles flous caractérisés par des fonctions d'appartenance  $\mu_{A_i}(x): \mathbb{R}^p \rightarrow [0,1]$  et  $\mu_{B_i}(y): \mathbb{R} \rightarrow [0, 1]$ .

Pour le cas des entrées numériques précises (*crisp*) et en utilisant la t-norme, le schéma de raisonnement peut être simplifié, aboutissant à celui bien connu dans la littérature appelé le max-min ou inférence de Mamdani.

L'inférence de Mamdani [42] de cas de la règle composée définie par l'expression (2.3) est :

-Pour chaque règle  $R_i$  du système ( $i = 1, \dots, r$ ), calculer le degré d'accomplissement  $\beta_i$  de l'antécédent :

$$\beta_i = \mu_{A_{i1}}(x_1) \wedge \mu_{A_{i2}}(x_2) \wedge \dots \wedge \mu_{A_{ip}}(x_p) \quad (2.4)$$

-L'ensemble flou de sortie  $B_{i'}$  est dérivé pour chaque règle, en utilisant la t-norme du minimum

$$\mu_{B_{i'}}(y) = \beta_i \wedge \mu_{B_i}(y), \forall y \in Y \quad (2.5)$$

-L'ensemble flou de sortie agrégé  $B'$  est calculé en prenant le maximum (union) des conclusions individuelles  $B_{i'}$  :

$$\mu_{B'}(y) = \max_{i=1,2,\dots,r} \mu_{B_i'}(y), \quad \forall y \in Y \quad (2.6)$$

L'expression (2.6) peut être généralisée en introduisant des opérateurs de type t-norme/t-conorme. Le sous-ensemble flou de sortie  $B'$  peut alors être obtenu par l'expression :

$$\mu_{B'}(y) = \underset{i=1,2,\dots,r}{\perp} T(\mu_{A_i}(x), I(\mu_{A_i}(x), \mu_{B_i}(y))), \quad \forall y \in Y \quad (2.7)$$

où  $\perp$  représente une t-conorme,  $T$  est une t-norme et  $I$  représente l'implication floue.

La défuzzification consiste en une transformation qui remplace le sous-ensemble flou de sortie de l'inférence par une valeur numérique unique. Il existe plusieurs techniques de défuzzification mais la plus utilisée est la méthode par centre de gravité, noté COG du sous-ensemble flou  $B'$  résultant de l'inférence :

$$Y_{COG}(B') = \frac{\int_a^b y \cdot \mu_{B'}(y) dy}{\int_a^b \mu_{B'}(y) dy} \quad (2.8)$$

### 2.2.2. Modèle flou type Takagi-Sugeno

Le modèle flou proposé par Takagi et Sugeno [44] est approprié pour l'approximation d'une classe générale de systèmes non linéaires. Le conséquent peut s'exprimer, sous forme d'une constante, d'un polynôme ou de manière plus générale d'une fonction ou d'une équation différentielle dépendant des variables associées à l'antécédent et peut s'écrire :

$$R_i : \text{Si } x \text{ est } A_i \text{ Alors } y_i = f_i(x), \quad i = 1, \dots, r \quad (2.9)$$

Typiquement les fonctions  $f_i(x)$  sont choisies comme des fonctions paramétrées appropriées.

La forme de paramétrisation souvent utilisée est la forme affine, donnée par :

$$y_i = \mathbf{a}_i^T \mathbf{x} + d_i \quad (2.10)$$

où  $\mathbf{a}_i \in \mathfrak{R}^p$  est un vecteur de paramètres et  $d_i$  est un scalaire. Ce modèle est appelé le modèle affine Takagi-Sugeno.

Les conclusions des règles dans ce modèle sont alors des hyperplans dans l'espace  $\mathfrak{R}^{p+1}$ . Le modèle global est composé par la succession des modèles locaux (linéaires) et peut être vu comme une approximation par morceaux d'une surface non linéaire correspondant à la sortie du système.

Pour le calcul du degré d'accomplissement  $\beta_i(x)$  de l'antécédent nous utilisons les opérateurs de la logique floue (§ 2.2.1). La sortie finale se calcule comme la moyenne des sorties correspondants aux règles  $R_i$ , pondérées par le degré d'accomplissement normalisé  $\lambda_i$ , selon l'expression [44] :

$$y = \frac{\sum_{i=1}^r \beta_i(\mathbf{x}) \cdot y_i}{\sum_{i=1}^r \beta_i(\mathbf{x})} = \sum_{i=1}^r \lambda_i y_i \quad \text{où} \quad \lambda_i = \frac{\beta_i(\mathbf{x})}{\sum_{i=1}^r \beta_i(\mathbf{x})}$$

Le modèle affine Takagi-Sugeno, avec une structure commune du conséquent, peut être exprimé comme un modèle pseudo-linéaire avec des paramètres dépendants des entrées :

$$y = \left( \sum_{i=1}^r \lambda_i(\mathbf{x}) a_i^T \right) \mathbf{x} + \sum_{i=1}^r \lambda_i(\mathbf{x}) d_i = \phi(x) \mathbf{x} + d(\mathbf{x}) \quad (2.11)$$

où  $\phi(x) = \sum_{i=1}^r \lambda_i(x) a_i$  et  $d(x) = \sum_{i=1}^r \lambda_i(x) d_i$

Dans le cadre d'identification des systèmes. Un système flou approxime typiquement un modèle de régression dynamique non linéaire de la forme :

$$y(k+1) = f(x(k)) \quad (2.12)$$

où  $f$  représente la fonction non linéaire et le vecteur de régression  $x(k)$  à l'instant  $k$

Les règles du modèle dynamique TS prennent alors la forme générale suivante :

$$R_i: \text{ Si } z_i(k) \text{ est } A_i \text{ Alors } y_i(k+1) = \theta_i^T w(k), \quad i = 1, \dots, r \quad (2.13)$$

où les variables de l'antécédent  $z(k)$  et du conséquent  $w(k)$  sont usuellement choisies à partir du vecteur de régression  $x(k)$ .

Le modèle a deux jeux de paramètres, où  $\theta_i$  présente les conséquents et  $A_i$  représente les fonctions d'appartenance pour les sous-ensembles flous. Ainsi, le modèle flou de type **TS** peut approximer les systèmes non linéaires avec des non linéarités douces et abruptes. Il permet de faire un choix spécifique des variables de l'antécédent et du conséquent et de modéliser des structures dynamiques qui changent en fonction de variables connues.

Dans le modèle dynamique représenté par les équations (2.13) est de type MISO. De ce fait, les systèmes MIMO peuvent être représentés de deux manières : soit la fonction non linéaire  $f(x)$  est une fonction vectorielle, soit le système MIMO est décomposé dans un ensemble de systèmes MISO couplés. Cependant, en modélisation floue l'approche de décomposition est la plus adoptée en général.

### 2.3. Identification de modèles flous

La construction de modèles flous peut se faire à partir de la connaissance préalable et les données. La connaissance préalable peut être plutôt d'une nature qualitative ou heuristique, issue de la connaissance des experts. La construction de modèles flous à partir de données (appelée identification floue) implique des méthodes basées sur la logique floue et le raisonnement approximatif, ainsi des idées issues d'autres domaines de l'intelligence artificielle [42]. La connaissance et les données dans un modèle flou peuvent être intégrées dans une collection de règles. De cette façon, une certaine structure du modèle est créée [45][46].

#### 2.3.1. Structure du modèle flou

En modélisation floue des systèmes, la sélection de la structure implique les choix de variables d'entrée et de sortie, structure des règles, nombre et type de fonctions d'appartenance pour chaque variable, et le type de mécanisme d'inférence.

Une fois que la structure est fixée, la performance de la méthode de modélisation peut être réglée avec précision en ajustant les paramètres. Les termes ajustables des modèles linguistiques sont les paramètres des fonctions d'appartenance, de l'antécédent et du conséquent ainsi que les règles.

#### 2.3.2. Identification basée sur des données entrée-sortie

On se propose donc d'obtenir un modèle flou à partir des données numériques issues du système à modéliser. L'objectif des entrées appliquées au système est de parcourir l'ensemble de l'espace dans lequel on recherche à modéliser le comportement du système. Selon les paramètres à régler et les stratégies utilisées, différentes méthodes ont été proposées dans la littérature : schéma de recherche en mosaïque [47], utilisation du gradient de la descente [43][48], utilisation du clustering et de gradient de descente [49] [50] et utilisation des stratégies évolutives [51] [52].

### 2.4 Modélisation basée sur des techniques de coalescence floue

Les méthodes d'identification basées sur la coalescence floue sont des méthodes qui ont des liens avec les domaines de l'analyse des données et de la reconnaissance des formes. Dans ces domaines, le concept d'une appartenance partielle est utilisé pour représenter le degré de similitude d'un objet avec un certain objet prototype. Le degré de similitude est habituellement calculé en utilisant une mesure appropriée de distance. Ainsi, ces objets, représentés comme un vecteur dans un espace de caractéristiques, peuvent être regroupés de manière que les vecteurs d'un même cluster soient aussi semblables que possible. Les paramètres du conséquent pour chaque règle peuvent être obtenus par estimation de type moindres carrés ou filtre de Kalman.

#### 2.4.1. Notions et Concepts de base

Les techniques de classification sont un outil important qui a trouvé des applications variées. D'un point de vue général, les techniques de classification ont pour but de regrouper les éléments d'un ensemble  $Z = \{z_1, z_2, \dots, z_N\}$  en un nombre optimal de classes selon leurs ressemblances.

Il existe un grand nombre de typologies de techniques de classification qui peuvent être regroupé en plusieurs catégories selon le type d'apprentissage tels que l'apprentissage supervisé, l'apprentissage par renforcement et l'apprentissage non supervisé. Ce dernier, consiste à établir des relations d'entrée/sortie en cherchant la formation de classes dans un ensemble d'observations sur lesquelles on ne dispose pas d'informations quant à leur appartenance aux éventuelles classes. Les méthodes de coalescence floue (regroupement ou clustering) font partie de ces méthodes, car elles n'utilisent pas des étiquettes préexistantes pour les classes [42][53].

#### 2.4.2 Coalescence floue

L'identification floue peut être vue comme une recherche de la décomposition du système. En exploitant le fait qu'en général, la complexité des systèmes n'est pas uniforme et comme il n'est pas possible de prévoir à l'avance qu'il y ait suffisamment de connaissance concernant cette tâche, nous considérons alors des méthodes de génération automatique de la décomposition, principalement à partir des données du système. Dans cette section nous abordons une classe appropriée d'algorithmes de coalescence floue (clustering) utilisée à ce propos.

## 2.5 Méthodes de coalescence floue

Au niveau de la modélisation, à chaque cluster correspond théoriquement un fonctionnement homogène du système qui peut se présenter sous la forme d'un modèle linéaire local.

Dans les méthodes de clustering flou, les données sont typiquement des observations (mesures) issues d'un certain processus physique. Chaque  $k$ -ième observation constitue un vecteur noté par  $z_k = [z_{1k}, z_{2k}, \dots, z_{nk}]^T$ , avec  $1 \leq k \leq N, z_k \in \mathfrak{R}^n$ , où  $N$  représente le nombre des observations et  $n$  correspond au nombre de variables mesurées, attributs ou caractéristiques dans l'espace de représentation  $\mathfrak{R}^n$ . Un ensemble de  $N$  observations est dénoté par  $Z = \{z_k | k = 1, 2, \dots, N\}$  et il est représenté comme une matrice sous la forme suivante :

$$Z = \begin{bmatrix} z_{11} & z_{12} & \dots & z_{1N} \\ z_{21} & z_{22} & \dots & z_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ z_{n1} & z_{n2} & \dots & z_{nN} \end{bmatrix} \quad (2.14)$$

Pour la reconnaissance des formes, les colonnes de cette matrice sont appelées les objets ou patrons et  $Z$  est appelée la matrice patronne. Quand le clustering est appliqué pour la modélisation et l'identification des systèmes dynamiques, les colonnes de  $Z$  contiennent les échantillons des signaux observés au cours du temps, et les lignes sont, par exemple, les variables physiques mesurées dans le système.

Afin de représenter la dynamique du système, il est nécessaire d'inclure typiquement les valeurs passées des variables dans la matrice  $Z$ .

Par exemple, dans le cas d'un système non linéaire représenté par un modèle NARX de deuxième ordre  $y(k+1) = f(y(k), y(k-1), u(k), u(k-1))$  les règles  $R_i$  d'un modèle dynamique affine Takagi-Sugeno ont la forme suivante :

$$\begin{aligned} R_i : & \text{ Si } y(k) \text{ est } A_{i1} \text{ et } y(k-1) \text{ est } A_{i2} \text{ et } u(k) \text{ est } B_{i1} \text{ et } u(k-1) \text{ est } B_{i2} \\ \text{ Alors } & y(k+1) = a_{i1}y(k) + a_{i2}y(k-1) + b_{i1}u(k) + b_{i2}u(k-1) + d \end{aligned} \quad (2.15)$$

La matrice des données prend la forme :

$$Z^T = \begin{bmatrix} y(1) & y(2) & u(1) & u(2) & \dots & y(3) \\ y(2) & y(3) & u(2) & u(3) & \dots & y(4) \\ \vdots & & \vdots & & \ddots & \vdots \\ y(N-2) & y(N-1) & u(N-2) & u(N-1) & \dots & y(N) \end{bmatrix}$$

Les algorithmes de clustering flou visent à réaliser une classification d'un ensemble de données en établissant une partition floue des observations en un certain nombre de classes. La notion de partition floue, qui s'est avéré d'une grande utilité pour le développement des techniques floues de classification, a été introduite par Ruspini [54]. Au sens de Ruspini (partition floue stricte), une  $c$ -partition floue d'un ensemble  $Z$  peut être obtenue en définissant des  $c$  sous-ensembles flous de  $Z$  tel que la somme des degrés d'appartenance pour chaque observation de  $Z$  soit unitaire. Ainsi, à chaque observation est associé un vecteur de degré d'appartenance aux différentes classes. Par conséquent, la juxtaposition de ces vecteurs pour l'ensemble des  $N$  observations de  $Z$  amène à la définition d'une matrice d'appartenance  $U$  (de dimension  $c \times N$ ), où l'élément  $\mu_{ik}$  représente le coefficient d'appartenance de l'observation  $z_k$  à la classe. Cette matrice établit une relation d'ordre floue et traduit l'idée d'une partition floue en  $c$  classes. La matrice d'appartenance  $U = [\mu_{ik}]$  est également appelée matrice de partition floue avec les propriétés suivantes :

$$\mu_{ik} \in [0, 1], \quad 1 \leq i \leq c, \quad 1 \leq k \leq N \quad (2.16)$$

$$0 < \sum_{k=1}^N \mu_{ik} < N, \quad 1 \leq i \leq c \quad (2.17)$$

$$\sum_{i=1}^c \mu_{ik} = 1, \quad 1 \leq k \leq N \quad (2.18)$$

L'expression (2.18) traduit une condition de normalisation d'inspiration probabiliste. Cependant, bien que cette approche de la définition d'une partition floue soit la plus utilisée, différents auteurs [55] [56] présentent des approches où cette contrainte est relaxée. Il s'agit d'approches qui conduisent à une partition dite possibiliste, dans laquelle en plus de respecter les expressions (2.16) et (2.17), la contrainte exprimée par (2.18) est remplacée par une autre moins restrictive :

$$\exists i, \mu_{ik} > 0, \quad 1 \leq k \leq N \quad (2.19)$$

La contrainte (2.19) doit assurer que chaque point soit assigné à au moins un des sous-ensembles flous avec une appartenance supérieure à zéro.

Dans les paragraphes suivants, trois (03) algorithmes de coalescence floue sont présenté, basés sur la minimisation d'une fonction objectif : L'algorithme des C-Moyennes Floues, (en anglais *Fuzzy c-Means* FCM) [57] [58], ainsi que deux de ses extensions, l'algorithme de Gustafson-Kessel (GK)[59] et l'algorithme de Geva Getha (GG)[60]. Le premier Algorithme permet la détection de classes hyper sphériques, tandis que le deuxième, détecte des classes hyper ellipsoïdales, typiquement mieux adaptées à la géométrie des observations.

### 2.5.1 Groupage FCM

Une grande famille d'algorithmes de clustering flou est basée sur la minimisation de la fonction objectif C-Means Floue formulée comme :

$$J_{FCM}(Z; \mathbf{U}, \mathbf{V}) = \sum_{i=1}^c \sum_{k=1}^N (\mu_{ik})^m D_{ikA}^2 \quad (2.20)$$

Où  $U = [\mu_{ik}]$  est la matrice de partition floue de dimension  $c \times N$ ,  $\mathbf{V} = [v_1, v_2, \dots, v_c]$  est un vecteur de prototypes de clusters qui doit être déterminé, avec  $v_i \in \mathfrak{R}^n$  centre de la  $i^{\text{ème}}$  classe,  $1 \leq i \leq c$ .

$$D_{ikA}^2 = \|z_k - v_i\|_A^2 = (z_k - v_i)^T \mathbf{A} (z_k - v_i), \quad 1 \leq i \leq c, \quad 1 \leq k \leq N \quad (2.21)$$

L'équation (2.31) représente la norme de distance quadratique dans l'espace considéré, qui définit la mesure de distance entre l'observation  $z_k$  et  $v_i$ , le centre au sens de la métrique induite par  $\mathbf{A}$  et  $m \in [1, \infty[$  est un facteur qui désigne le degré flou de la partition.

Le paramètre  $m$  influence directement la forme des clusters dans l'espace des données du système. Quand le facteur  $m$  s'approche de la valeur 1, la forme de la fonction d'appartenance pour chaque cluster est presque booléenne  $\mu_{ik} \in \{0,1\}$  et  $v_i$  sont les moyennes ordinaires des clusters. De l'autre côté, plus  $m$  est grand, plus la partition est floue. Quand  $m \rightarrow \infty$ , la partition devient floue au maximum ( $\mu_{ik} = 1/c$ ) et les moyennes des clusters sont toutes égales à la moyenne de  $Z$ . Ces propriétés limites sont indépendantes de la méthode d'optimisation sélectionnée [61]. Bien que le choix de  $m$  dépende des données [62], souvent ce paramètre est initialisé à une valeur entre 1,5 et 2,5.

La valeur de la fonction objectif  $J_{FCM}(Z, U, V)$  peut être vue comme une mesure de la variance totale de  $z_k$  par rapport aux prototypes  $v_i$ . La minimisation de  $J_{FCM}(Z, U, V)$  représente un problème d'optimisation non linéaire qui peut être résolu par différentes méthodes, parmi lesquelles la plus utilisée est basée sur l'itération de Picard à travers les conditions du premier ordre pour les points stationnaires. Cette méthode est connue comme l'algorithme FCM. Les points stationnaires de la fonction objectif (2.20) peuvent être trouvés en ajoutant la contrainte de normalisation (2.16) au critère  $J_{FCM}$  au moyen des multiplicateurs de Lagrange :

$$\bar{J}_{FCM}(Z; \mathbf{U}, \mathbf{V}, \boldsymbol{\lambda}) = \sum_{i=1}^c \sum_{k=1}^N (\mu_{ik})^m D_{ikA}^2 + \sum_{k=1}^N \lambda_k \left[ \sum_{i=1}^c \mu_{ik} - 1 \right] \quad (2.22)$$

où les termes  $\lambda_k$  sont les coefficients multiplicateurs de Lagrange. La minimisation du critère  $J_{FCM}$  s'obtient alors en annulant le gradient de  $\bar{J}_{FCM}$  par rapport  $\mathbf{U}$ ,  $\mathbf{V}$  et  $\boldsymbol{\lambda}$ . Si  $D_{ikA}^2 > 0 \quad \forall i, k$  et  $m > 1$  alors on obtient les relations de mise à jour Pour les coefficients  $\mu_{ik}$  de la matrice de partition floue est comme suit :

$$\mu_{ik} = \frac{1}{\sum_{j=1}^c (D_{i,k,A}^2 / D_{j,k,A}^2)^{1/(m-1)}}, \quad 1 \leq i \leq c, \quad 1 \leq k \leq N. \quad (2.23)$$

Et pour les (centres)  $v_i$  des clusters nous aurons :

$$v_i = \frac{\sum_{k=1}^N \mu_{ik} \cdot z_k}{\sum_{k=1}^N (\mu_{ik})^m}, \quad 1 \leq i \leq c. \quad (2.24)$$

Cette solution satisfait aussi les contraintes (2.16) et (2.17). Il faut remarquer que l'expression (2.24) exprime  $v_i$  comme la moyenne pondérée des données qui appartiennent à un cluster, où le facteur de pondération est associé aux degrés d'appartenance floue.

Le processus itératif s'arrête lorsque la partition devient stable, c'est-à-dire lorsqu'elle n'évolue plus significativement, entre deux itérations successives. Ceci s'exprime de manière générale par la vérification de l'expression (2.25) où le terme de gauche traduit une norme matricielle et le coefficient  $\varepsilon$  définit le seuil de convergence :

$$\|\mathbf{U}^{(t)} - \mathbf{U}^{(t-1)}\| < \varepsilon \quad (2.25)$$

L'expression  $\mathbf{U}(t)$  représente la matrice de partition floue à l'itération (t). L'algorithme général de résolution *FCM* peut se formuler ainsi :

**Algorithme FCM**

Etant donné un ensemble de vecteurs de données  $Z$  :

Définir la matrice  $A$  de métrique de la distance  $D_{i,k,A}^2$

Fixer le nombre de clusters  $1 < c < N$

**Fixer** le degré de flou  $m > 1$  et la tolérance de fin d'algorithme  $\varepsilon > 0$

**Initialiser** la matrice  $U = [\mu_{ik}]$  de partition floue

**Répéter**

Pas 1. Calculer les prototypes (centres) des clusters (Eq. (2.24))

Pas 2. Calculer les distances pour chaque cluster (Eq. (2.22))

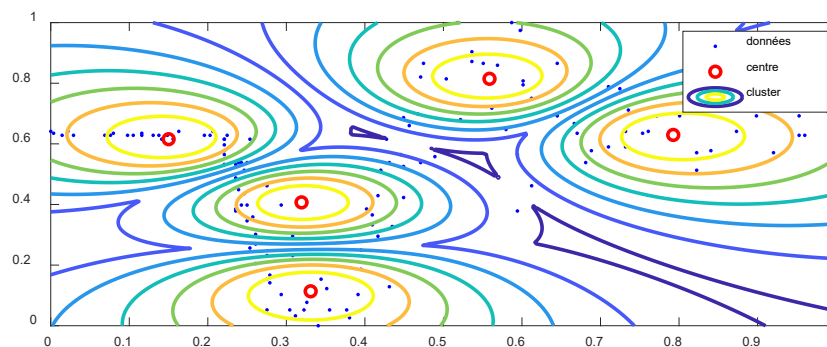
Pas 3. Mettre à jour la matrice  $U$  de partition floue (Eq. (2.23))

**Jusqu'à** obtenir la stabilité de la partition (Eq. (2.25))

**Remarque :** La valeur usuelle pour le seuil de terminaison est  $\varepsilon = 0.001$ , bien qu'une valeur de  $\varepsilon = 0.01$  convienne dans la plupart des cas [42]. La matrice d'induction de la norme  $A$  peut être adaptée en utilisant des estimations de la covariance des données, et peut être utilisée pour estimer la dépendance statistique des données dans chaque cluster.

**Exemple**

Nous illustrons les idées ci-dessus à l'aide d'un ensemble de données d'une expérience sur l'efficacité de casques de protection des motos [63]. Cet ensemble, tel que présenté dans la Fig. 2.3, se compose de 133 points prises à travers un accéléromètre. Pour diverses raisons, les points temporels ne sont pas régulièrement espacés, et il y a plusieurs observations à certains points de temps. Notre intérêt ici est de discerner la courbe d'accélération utilisant un modèle TS. Considérant les paramètres du clustering suivantes :  $c = 4$ , et  $m = 2$ . Les courbes dans la (fig. 2.2) représentent les valeurs d'appartenance aux clusters et qui sont inversement proportionnelles aux distances.



**Fig. 2.2** Résultats de l'algorithme FCM par l'ensemble de données de motorcycle.

**2.5.2 Clustering GK**

L'algorithme proposée par Gustafson et Kessel [59] n'est d'autre qu'une extension de l'algorithme FCM qui a l'avantage de tenir compte de la forme propre de chacune des classes, en particulier, de permettre la détection des classes hyper-ellipsoïdales, typiquement mieux adaptées à la géométrie des observations. Pour cela, pour chaque cluster, la mesure de distance est définie à l'aide d'une métrique  $A_i$  :

$$D_{i,k,A_i}^2 = (\mathbf{z}_k - \mathbf{v}_i)^T A_i (\mathbf{z}_k - \mathbf{v}_i) \quad (2.26)$$

où la matrice de norme induite  $A_i$  est propre à chaque cluster, avec  $1 \leq i \leq c$ . Les matrices  $A_i$  sont utilisées comme variables d'optimisation dans la fonctionnelle C-Means, permettant ainsi à chaque cluster d'adapter la norme de distance à la structure des données.

La fonction objectif de l'algorithme GK est définie par :

$$J_{GK}(Z; \mathbf{U}, \mathbf{V}, \mathbf{A}) = \sum_{i=1}^c \sum_{k=1}^N (\mu_{ik})^m D_{ik,A_i}^2 \quad (2.27)$$

où  $\mathbf{A}$  qui représente un  $c$ -uplet de matrices de norme induite définies positives  $\mathbf{A} = (A_1, A_2, \dots, A_c)$ . et  $\mathbf{U} \in \mathfrak{R}^{c \times N}$ ,  $\mathbf{V} \in \mathfrak{R}^{n \times c}$  et  $\mathbf{A} \in \mathfrak{R}^{n \times n}$  correspondent aux solutions de la minimisation (points stationnaires) de la fonctionnelle  $J_{GK}$ .

Pour  $\mathbf{A}$  fixé, les conditions (2.16) et (2.17) peuvent être appliquées. Néanmoins, la fonction objectif (2.37) ne peut pas être minimisée directement par rapport à  $A_i$  puisqu'elle est linéaire en  $A_i$ . Pour obtenir une solution *réalisable*,  $A_i$  doit être contrainte d'une certaine façon, pour cela, il suffit d'imposer une contrainte sur le déterminant de  $[A_i]$ . En permettant à la matrice  $A_i$  de varier alors que son déterminant reste fixe, cela correspond à optimiser la forme du cluster tandis que son volume reste constant. C'est pourquoi on considère l'algorithme GK comme une méthode de clustering avec une mesure adaptative de la distance. La contrainte sur le volume se traduit par l'expression :

$$|A_i| = \psi_i \quad (2.28)$$

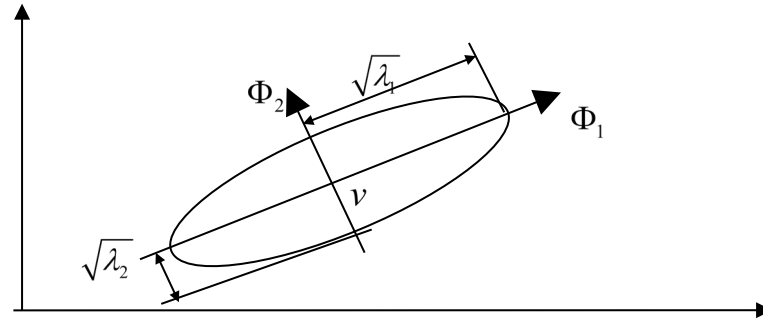
Nous avons la possibilité d'affecter des volumes différents pour chaque cluster par l'intermédiaire de  $\psi_i$ , initialisé par défaut à la valeur 1. En utilisant la méthode des multiplicateurs de Lagrange [59], la minimisation du critère d'optimisation conduit à l'expression de  $A_i$  suivante :

$$\mathbf{A}_i = (\psi_i \det(\mathbf{F}_i))^{1/n} \mathbf{F}_i^{-1} \quad (2.29)$$

où  $\mathbf{F}_i$  est la matrice de covariance floue du  $i^{\text{ème}}$  cluster et est définie par l'expression :

$$\mathbf{F}_i = \frac{\sum_{k=1}^N (\mu_{ik})^m (\mathbf{z}_k - \mathbf{v}_i)(\mathbf{z}_k - \mathbf{v}_i)^T}{\sum_{k=1}^N (\mu_{ik})^m} \quad (3.30)$$

La structure propre de la matrice de covariance floue  $\mathbf{F}_i \in \mathfrak{R}^{n \times n}$  fournit une information précieuse sur la forme et l'orientation des clusters hyper ellipsoïdaux. En effet, la racine carrée de chacune de ses valeurs propres  $\lambda_j, j=1, \dots, n$  représente les longueurs des axes dans l'hyperespace. Les directions des axes sont données par les vecteurs propres  $\Phi_j$ . Cela est illustré sur la figure Fig 2.3 pour un espace de dimension  $n = 2$ .



**Fig. 2.3** Interprétation géométrique d'une matrice de covariance d'un cluster

La substitution des équations (2.29) et (2.30) dans l'expression (2.26) donne une norme quadratique de distance Mahalonobis généralisée entre le vecteur  $\mathbf{z}_k$  et le centre du cluster  $\mathbf{v}_i$ , où la covariance est pondérée par les degrés d'appartenance en  $U$ . La mesure de distance pour l'algorithme GK sera définie comme suit :

$$D_{ik_i}^2 = (\mathbf{z}_k - \mathbf{v}_i)^T (\psi_i \det(\mathbf{F}_i))^{1/n} \mathbf{F}_i^{-1} (\mathbf{z}_k - \mathbf{v}_i) \quad (2.31)$$

L'algorithme de clustering de Gustafson-Kessel peut se formuler ainsi :

**Algorithme GK**

Etant donné un ensemble de vecteurs de données  $Z$  :  
 Fixer le nombre de clusters  $1 < c < N$   
 Fixer le degré de flou  $m > 1$ , le volume  $\psi_i$  des clusters et la tolérance de fin d'algorithme  $\varepsilon > 0$

Initialiser la matrice  $\mathbf{U} = [\mu]_{ik}$  de partition floue

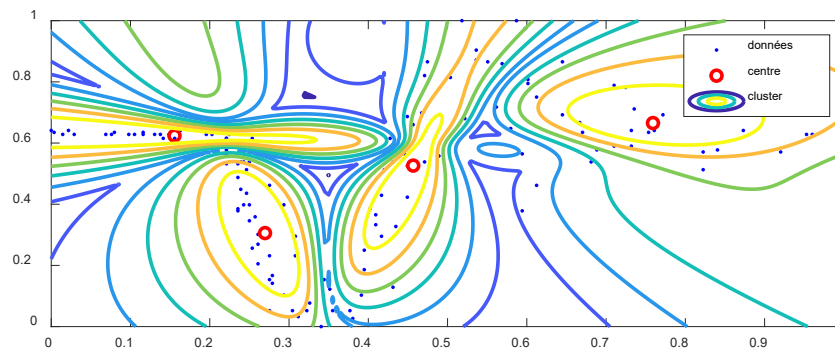
**Répéter**

- Pas 1. Calculer les prototypes (centres) des clusters (Eq. (2.24))
- Pas 2. Calculer les matrices de covariance de cluster (Eq. (2.30))
- Pas 3. Calculer les distances pour chaque cluster (Eq. (2.31))
- Pas 4. Mettre à jour la matrice U de partition floue (Eq. (2.24))

**Jusqu'à** obtenir la stabilité de la partition (Eq. (2.25))

L'algorithme FCM utilise une norme de distance fixe et préfère ainsi fortement les clusters d'une forme géométrique induite par cette norme. L'algorithme GK remédie à cet inconvénient en adaptant localement la norme de distance. Ainsi, l'algorithme GK présente un avantage sur le FCM qui consiste en la détection des clusters de formes et d'orientations différentes pour un ensemble des données dans l'espace de représentation. Cependant, il est plus lourd en temps de calcul.

Reprenant le même exemple de l'ensemble des données de moto, la figure Fig. 2.4 montre les clusters et les centres obtenus avec un pré choix de nombre de cluster = 4 et m = 2.



**Fig. 2.4** Résultats de l'algorithme de GK par l'ensemble de données synthétiques.

### 2.5.3 Algorithme de Clustering de Gath–Geva

L'algorithme de clustering des estimations de maximum de vraisemblance floues utilise une norme de distance basée sur les estimations de maximum de vraisemblance floues, proposées par Bezdek et Dunn [64] :

$$D_{i,k}(\mathbf{x}_k, \mathbf{v}_i) = \frac{(2\pi)^{\left(\frac{n}{2}\right)} \sqrt{\det(\mathbf{F}_i)}}{\alpha_i} \exp\left(\frac{1}{2}(\mathbf{x}_k - \mathbf{v}_i)^T \mathbf{F}_i^{-1}(\mathbf{x}_k - \mathbf{v}_i)\right) \quad (2.32)$$

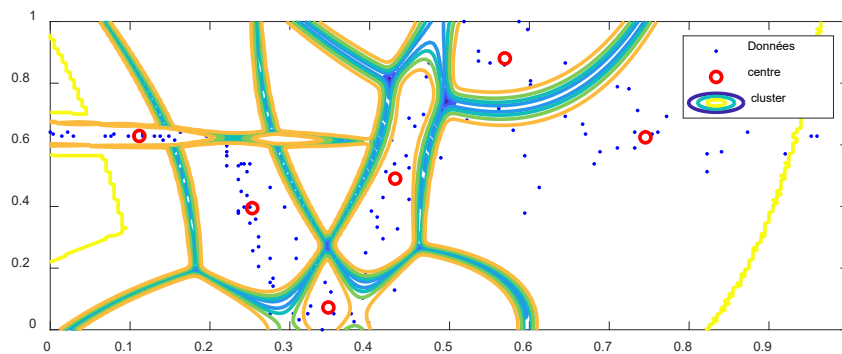
Contrairement à l'algorithme GK, cette norme de distance implique un terme exponentiel et diminue donc plus rapidement que la norme du produit interne.  $F_i$  désigne la matrice de covariance floue du  $i^{\text{ème}}$  cluster, de manière similaire à l'algorithme GK, donnée par (2.24). Le  $\alpha_i$  est la probabilité a priori de sélectionner le cluster  $i$ , donnée par

$$\alpha_i = \frac{1}{N} \sum_{k=1}^N \mu_{i,k}. \quad (2.33)$$

Les degrés d'appartenance  $\mu_{ik}$  sont interprétés comme les probabilités postérieures de sélectionner le  $i^{\text{ème}}$  cluster étant donné le point de données  $x_k$ . Gath et Geva [60] ont signalé que l'algorithme de regroupement d'estimations de probabilité maximale floue est capable de détecter des grappes de formes, de tailles et de densités variables.

En effet, la matrice de covariance des clusters est utilisée conjointement avec une distance « exponentielle », et les clusters ne sont pas contraints en volume. Cependant, cet algorithme est moins robuste dans le sens qu'il a besoin d'une bonne initialisation, car en raison de la norme de distance exponentielle, il converge vers un optimum local proche.

Avec le même exemple de l'ensemble des données de moto, la figure (Fig. 2.5) montre les clusters et les centres obtenus avec nombre de cluster pré sélectionné  $c=4$  et  $m=2$ .



**Fig. 2.5** Résultats de l'algorithme de Gath–Geva par l'ensemble de données synthétiques.

## 2.6 Méthodologie de Construction de modèles TS

Dans le cadre de la modélisation floue des systèmes, notre intérêt porte sur l'obtention des modèles TS qui permettent une décomposition d'un système non linéaire dans un ensemble de régions linéaires. Pour cela, nous appliquons des techniques de clustering flou dans l'espace produit d'entrée-sortie des données. Par conséquent, à partir des algorithmes présentés précédemment qui ont tous des caractéristiques particulières peuvent être réunies dans une seule vue d'ensemble. Nous abordons ainsi, une description d'une méthodologie générale pour la construction des modèles flous du type TS, en mettant l'accent sur les besoins communs qui sont : La génération des fonctions d'appartenance des antécédents, L'obtention des paramètres des conséquents et la validation numérique du modèle final. Ces aspects seront présentés dans les paragraphes qui suivent. La procédure est itérative, dans la mesure où, dans une même

session de modélisation, on peut répéter certains pas afin de tester différents choix pour plusieurs paramètres.

### 2.6.1 Conception et acquisition des données

C'est l'étape initiale de n'importe quelle méthode d'identification, puisqu'elle détermine le contenu d'information des données d'identification. Bien que le choix du signal d'excitation puisse être dépendant du problème et système, le signal d'entrée devrait de préférence exciter le système dans toute la gamme de fonctionnement, tant en amplitude qu'en fréquence, pour les variables considérées. Des choix typiques sont des signaux multi-sinusoïdaux ou bien des signaux multi-échelon avec amplitude et largeur aléatoires. Un bruit blanc de petite amplitude est souvent ajouté à ces signaux pour garantir l'excitation appropriée de la dynamique du procédé. Le choix d'une période d'échantillonnage ainsi que la durée des expériences sont aussi des aspects importants [42].

### 2.6.2 Sélection de la structure entrée-sortie

Le but de cette étape est de déterminer les variables d'entrée et de sortie par rapport à l'objectif de la modélisation. La structure peut être sélectionnée de manière automatisée en comparant différentes structures candidates en termes de certaines mesures de performance. Quand il s'agit de l'identification des systèmes dynamiques, la structure et l'ordre de la dynamique du modèle doivent être choisis. Ainsi, la sélection de structure permet de traduire l'identification d'un système dynamique en un problème de régression qui peut être résolu de manière statique ou quasi-statique. Dans la plupart des cas, un choix raisonnable peut être fait par l'utilisateur, basé sur la connaissance préalable du processus.

### 2.6.3 Clustering flou

La sélection de la structure conduit à un problème de régression non linéaire statique, qui est alors approximé par une collection de sous modèles linéaires locaux. La localisation et les paramètres des sous modèles sont établis en partitionnant les données disponibles en clusters. Chacun des clusters définit une région floue (par l'intermédiaire des antécédents) dans laquelle le système peut être approximé localement par un sous modèle linéaire (au moyen des conséquents).

Afin de déterminer le nombre de clusters approprié, il est conseillé d'appliquer des mesures de validation du nombre de clusters et/ou de fusion de clusters. Plusieurs mesures de validité ont été introduites dans la littérature [42] [58].

### 2.6.4 Obtention du modèle flou TS

Afin d'obtenir un modèle approprié pour des propos de prédiction ou de synthèse d'un contrôleur, un modèle flou (de structure définie) basé sur des règles est extrait de la matrice de partition floue disponible ainsi que des prototypes des clusters. Les règles, les fonctions d'appartenance et d'autres paramètres qui constituent le modèle flou sont obtenus de façon automatique. Des méthodes de réduction peuvent être employées afin de réduire la base de règles initiale ou bien pour améliorer l'interprétation linguistique des fonctions d'appartenance [42].

La procédure d'obtention du modèle flou TS correspondant passe par la génération des fonctions d'appartenance des antécédents (§ 2.6.8) et par l'obtention des paramètres des conséquents (§ 2.7).

### 2.6.5 Validation du modèle flou TS

Au moyen de la validation, le modèle final est accepté soit comme approprié pour l'objectif donné, soit rejeté. Dans ce dernier cas, certaines étapes de la boucle d'identification peuvent être répétées avec un cadre différent, comme c'est également le cas dans d'autres approches d'identification de systèmes linéaires et non linéaires. En plus de la validation numérique par simulation, l'interprétation de modèles flous joue un rôle important dans l'étape de validation. La couverture de l'espace d'entrée par les règles peut être analysée et, pour une base de règles incomplète, des règles supplémentaires peuvent être fournies sur la base de connaissances préalable.

Pour terminer cette vue d'ensemble de la construction des modèles flous de type TS, nous abordons les aspects concernant l'obtention de la base de règles (i.e., la génération des fonctions d'appartenance des antécédents et l'obtention des paramètres des conséquents) et la validation finale du modèle.

### 2.6.6 Obtention de la base de règles du modèle flou TS

En général, pour les algorithmes de coalescence floue présentés dans la section précédente excepté l'algorithme FCM, chaque cluster obtenu dans l'espace produit d'entrée-sortie des données peut être vu comme une approximation locale linéaire de l'hypersurface de régression. Le modèle flou global peut être représenté convenablement comme une collection de règles du type TS affines (§. 2.1.2), de la forme donnée par l'expression suivante :

$$R_i : \text{Si } x \text{ est } A_i \text{ Alors } y_i = a_i^T x + d_i, \quad i = 1, \dots, r \quad 2.34$$

où le nombre de règles correspond au nombre de clusters issu du processus de clustering,  $x \in \mathfrak{R}^p$  est le vecteur d'entrée correspondant à une observation et  $y \in \mathfrak{R}$  est la variable de sortie.  $A_i$  est le sous-ensemble flou de l'antécédent de l' $i^{\text{ème}}$  règle, définie, dans ce cas, par une fonction d'appartenance multidimensionnelle. Pour sa part, le conséquent de chacune des règles est formé par le vecteur des paramètres  $a_i \in \mathfrak{R}^p$  et le scalaire  $d_i$ .

Les conclusions des règles dans ce modèle sont alors des hyperplans dans l'espace  $\mathfrak{R}_{p+1}$ . Les ensembles flous  $A_i$  peuvent être décrits dans leur forme naturelle multidimensionnelle, ou bien dans leur forme décomposée, en projetant la partition floue sur les variables de l'antécédent. Les paramètres  $a_i$  et  $d_i$  des conséquents sont estimés à partir des données en utilisant des méthodes des moindres carrés ou bien le filtre de Kalman.

### 2.6.7 Construction des fonctions d'appartenance des antécédents

Les fonctions d'appartenance des antécédents peuvent être obtenues en calculant les degrés d'appartenance directement dans l'espace produit des variables de l'antécédent. Elles peuvent être aussi extraites à partir de la matrice de partition floue  $U$  en appliquant le mécanisme de projection sur ces variables.

#### Génération des fonctions d'appartenance par projection

Le principe de cette méthode est de projeter pour chaque règle, les ensembles flous multidimensionnels définis point par point dans la matrice de partition floue  $U$  sur les variables individuelles des antécédents. Ainsi, la matrice de partition floue est projetée sur chacun des axes des variables (i.e., sur les répresseurs) de l'antécédent  $x_j$ , avec  $1 \leq j \leq p$ . Les règles  $R_i$ ,  $i = 1, \dots, r$  correspondantes au modèle flou TS sont alors exprimées sous la forme suivante :

$$R_i : \text{Si } x_1 \text{ est } A_{i1} \text{ et } \dots \text{ et } x_p \text{ est } A_{ip} \text{ Alors } y_i = \mathbf{a}_i^T \mathbf{x} + d_i \quad (2.35)$$

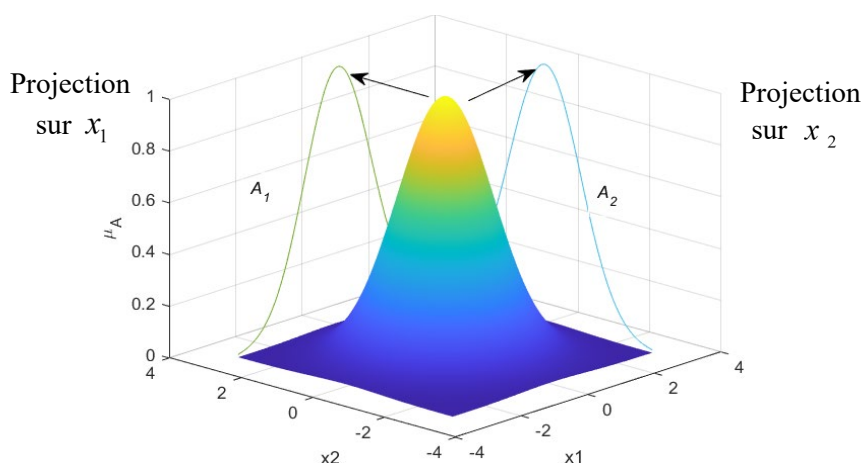
Pour obtenir les fonctions d'appartenance des sous-ensembles flous  $A_{ij}$  de l'antécédent, l'ensemble flou multidimensionnel défini point par point par l' $i^{\text{ème}}$  ligne de la matrice de partition est projeté sur les répresseurs  $x_j$ . Cette opération de projection est illustré dans la Figure (fig. 2.6), et représentée par l'expression suivante

$$\mu_{A_{ij}}(x_{jk}) = \text{proj}_j(\mu_{ik}) \quad 2.36$$

**Définition (projection) :** Soit un sous-ensemble flou  $A$  défini sur un univers  $X_1 \times X_2$ , produit cartésien de deux univers  $X_1$  et  $X_2$ . On définit la projection de  $A$  sur  $X_1$  comme le sous-ensemble flou de  $X_1$ , noté  $proj_{X_1}(A)$ , de fonction d'appartenance :

$$\mu_{proj_{X_1}(A)}(x_1) = \sup_{x_2 \in X_2} \mu_A(x_1, x_2), \forall x_1 \in X_1 \quad 2.36$$

On définit de manière analogue la projection de  $A$  sur  $X_2$ . Pour des raisons de simplicité on a défini cette projection sur deux univers, mais de la même façon on peut le faire sur  $p$  univers.



**Fig. 2.6** Illustration du mécanisme de projection

Comme il est déjà mentionné, le degré d'accomplissement  $\beta_i(x) = \mu_{A_i}(x)$  pour l' $i^{\text{ème}}$  règle doit être calculé comme une combinaison des degrés d'appartenance de chaque proposition individuelle en utilisant des opérateurs logiques flous. On reconstruit donc le cluster dans l'espace des entrées en appliquant l'opérateur intersection dans l'espace cartésien des variables de l'antécédent, de la façon suivante :

$$\beta_i = \mu_{A_1}(x_1) \wedge \mu_{A_2}(x_2) \wedge \dots \wedge \mu_{A_p}(x_p) \quad (2.38)$$

### Paramétrisation des fonctions d'appartenance

Une définition point par point de l'ensemble flou  $A_{ij}$  est obtenue en projetant l' $i^{\text{ème}}$  ligne  $\mu_i$  de la matrice de partition floue  $U$  sur la variable  $x_j$  de l'antécédent. Afin d'obtenir un modèle dans un but de prédiction ou de commande, les fonctions d'appartenance de l'antécédent doivent être exprimées sous une forme qui permet le calcul des degrés d'appartenance, même pour des données d'entrée non contenues dans l'ensemble des données  $Z$ . Cela est réalisé en approximant la fonction d'appartenance définie point par point par une fonction paramétrique appropriée. La fonction d'appartenance est adaptée à l'enveloppe des données projetées en optimisant numériquement ses paramètres.

L'avantage de cette méthode par rapport aux fonctions d'appartenance multidimensionnelles réside dans le fait que les fonctions d'appartenance projetées peuvent être toujours approximées par des ensembles flous convexes. De plus, des fonctions d'appartenance asymétriques peuvent être utilisées pour refléter la partition réelle du problème de régression non linéaire considéré.

## 2.7 Paramètres des conséquents

Les paramètres  $a_i \in \mathcal{R}^p$  et  $d_i$  du modèle TS donné par l'expression (2.34) peuvent être déterminés à partir de la structure géométrique des clusters dans le cas de l'algorithme GK, en général, par des techniques d'estimation de type moindres carré ou bien par la technique de Kalman.

### 2.7.1 Estimation des paramètres par la méthode des moindres carrés

Les paramètres des conséquents peuvent être établis par la technique des moindres carrés, en utilisant comme facteurs de pondération des données, les degrés d'appartenance de la matrice de partition floue  $U$  issus du processus de clustering. Cette approche conduit à une formulation de  $c$  problèmes indépendants de type moindres carrés pondérés, dans laquelle les degrés d'appartenance expriment l'importance de la paire de données  $(x_k, y_k)$  par rapport à chaque  $i^{\text{ème}}$  sous-modèle linéaire local, avec  $1 \leq i \leq c$ .

Les données d'identification entrée-sortie  $z_k = [x_k^T, y_k]^T$ , avec  $1 \leq k \leq N$ , et les degrés d'appartenance  $\mu_{ik}$  de la matrice de partition floue sont regroupés dans les matrices suivantes :

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}_1^T \\ \mathbf{x}_2^T \\ \vdots \\ \mathbf{x}_N^T \end{bmatrix}, \quad \mathbf{y} = \begin{bmatrix} \mathbf{y}_1^T \\ \mathbf{y}_2^T \\ \vdots \\ \mathbf{y}_N^T \end{bmatrix}, \quad \mathbf{w} = \begin{bmatrix} \mu_{i1} & 0 & \dots & 0 \\ 0 & \mu_{i2} & \dots & 0 \\ & & \vdots & \\ 0 & 0 & \dots & \mu_{iN} \end{bmatrix}, \quad (2.39)$$

Les paramètres des conséquents  $a_i$  et  $d_i$  appartenant à la règle correspondant à l' $i^{\text{ème}}$  cluster sont déterminés dans un seul vecteur de paramètres  $\theta_i$ , donné par :

$$\theta_i = [a_i^T, d_i]^T \quad (2.40)$$

Afin de faciliter le calcul, la matrice de régression  $X$  est augmentée en ajoutant un vecteur-colonne unitaire, selon l'expression :

$$X_\mu^e = [X, 1] \quad (2.41)$$

Si les colonnes de  $X_\mu^e$  sont linéairement indépendantes et  $i_k > 0$  pour  $1 \leq k \leq N$ , alors la solution des moindres carrés de  $y = X_e \theta + \varepsilon$  où la  $k$ -ième paire de données  $(x_k, y_k)$  est pondérée par  $\mu_{ik}$ , est donnée finalement par l'expression :

$$\theta_i = (X_e^T \mathbf{W}_i X_e)^{-1} X_e^T \mathbf{W}_i y \quad (2.42)$$

Les paramètres  $a_i$  et  $d_i$  sont donnés respectivement par :

$$a_i = [\theta_1, \theta_2, \dots, \theta_p], \quad d_i = \theta_{p+1} \quad (2.43)$$

### 2.7.2. Estimation des paramètres par la méthode de Kalman

Le filtre de Kalman (Kalman Filter KF) est l'un des développements mathématiques les plus intéressants dans la théorie de l'estimation linéaire [65] [66]. En effet, ce filtre est un ensemble d'équations mathématiques permettant d'effectuer des calculs récursifs et quadratiques efficaces pour résoudre quelques problèmes. Il décrit une solution récursive du problème de filtrage linéaire des données discrètes. De ce fait, le KF s'avère très puissant dans les aspects d'estimation des événements (états) passés, présents et futures. Il représente un estimateur d'état optimal des états d'un système linéaire stochastique avec l'hypothèse que les données de mesures soient bruitées.

L'opération d'estimation est déterminée selon la minimisation d'un certain critère ou d'une certaine fonction de coût reflétant une mesure quantitative de la qualité de l'estimation, il sera raisonnable de considérer des fonctions de coût de l'erreur non négatives, c'est-à-dire quadratiques. Dans ce cas, le modèle de régression linéaire liant une réponse aléatoire  $Y$  (appelée régressant) à une variable de régression  $x$  prend la forme suivante [67] :

$$Y_k = a_k x_k + d_k + v_k \quad (2.44)$$

où  $a_k$  et  $d_k$  sont les paramètres de régression inconnus, et  $v_k$  est un terme d'erreur aléatoire (bruit de mesure). Nous devons noter que  $Y$  et  $x$  sont des variables observables.

L'estimation des paramètres de régression est obtenue en utilisant  $n$  paires d'observations  $(x_k, y_k), k = 1, 2, \dots, n$ , appelées ensemble d'apprentissage. L'équation (2.43) peut être réécrite comme suit :

$$Y_k = a_k x_k + b_k + v_k = [x_k \quad 1] \begin{bmatrix} a_k \\ b_k \end{bmatrix} + v_k = \mathbf{C}_k \boldsymbol{\theta}_k + v_k \quad (2.45)$$

où  $\mathbf{C}_k = [x_k \ 1]$  sera considéré comme un vecteur d'observation et  $\boldsymbol{\theta}_k = [a_k \ b_k]^T$  sera le vecteur paramètre à déterminer. De l'équation (2.54),  $\boldsymbol{\theta}_k$  sera considéré comme une variable d'état au moment  $t = t_k$ , par conséquent nous pouvons proposer l'équation d'état suivante :

$$\boldsymbol{\theta}_k = \mathbf{A}\boldsymbol{\theta}_{k-1} + w_{k-1} \quad (2.46)$$

où  $\mathbf{A}$  est une matrice de transition d'ordre  $2 \times 2$ , et  $w$  est un bruit d'état. Le bruit d'état et le bruit de mesure doivent être statistiquement indépendants à valeurs moyennes nulles et blancs. Donc, leurs corrélations seront données par :

$$E[w_i w_j^T] = \begin{cases} \mathbf{Q} & i = j \\ \mathbf{0} & i \neq j \end{cases} \quad E[v_i v_j] = \begin{cases} r & i = j \\ 0 & i \neq j \end{cases} \quad E[w_i v_j] = 0 \quad \forall i, j$$

Soit  $\hat{\boldsymbol{\theta}}_{k/k-1}$  le meilleur estimé de  $\boldsymbol{\theta}$  au moment  $t = t_k$  basé sur les observations obtenues jusqu'au moment  $t = t_{k-1}$  (l'estimation d'état a priori de  $\boldsymbol{\theta}_k$ ), et  $\hat{\boldsymbol{\theta}}_{k/k}$  le meilleur estimé de  $\boldsymbol{\theta}$  au moment  $t = t_k$  basé sur les observations obtenues jusqu'au moment  $t = t_k$  (l'estimation d'état a posteriori de  $\boldsymbol{\theta}_k$ ). En procédant récursivement et en supposant que l'estimation  $\hat{\boldsymbol{\theta}}_{k-1/k-1}$  est connue, on peut alors donner une équation d'état prédictive et une équation de mesure prédictive comme le montre les équations (2.56) et (2.57) :

$$\hat{\boldsymbol{\theta}}_{k/k-1} = \mathbf{A} \hat{\boldsymbol{\theta}}_{k-1/k-1} \quad (2.47)$$

$$\hat{y}_k = \mathbf{C}_k \hat{\boldsymbol{\theta}}_{k/k-1} \quad (2.48)$$

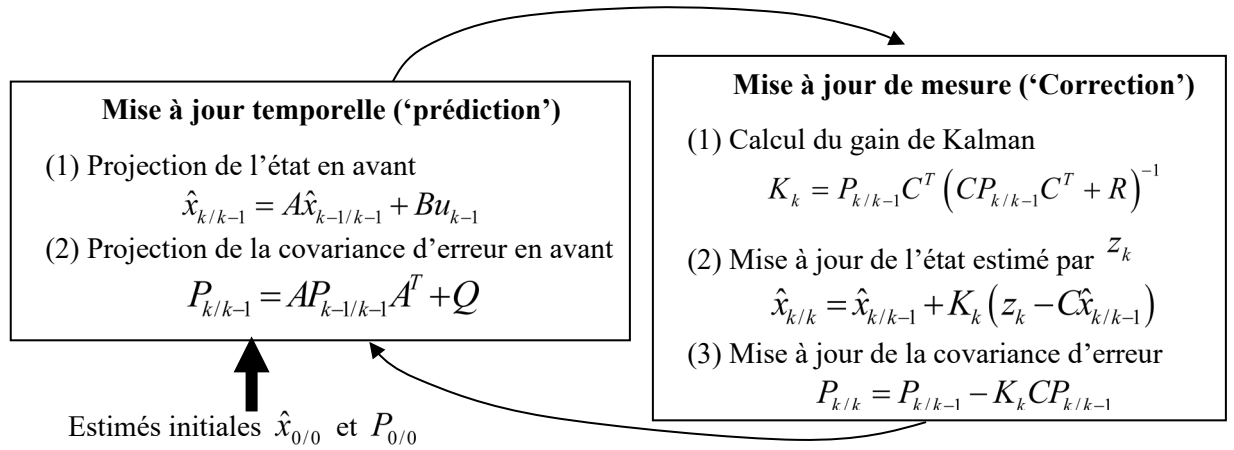
Comme la représentation du modèle du vecteur paramètre est complète, alors l'adaptation des paramètres par le KF peut être réalisée. L'ajustement des paramètres sera assuré par la récurrence suivante :

$$\hat{\boldsymbol{\theta}}_{k/k} = \hat{\boldsymbol{\theta}}_{k/k-1} + \mathbf{K}_k (y_k - \mathbf{C}_k \hat{\boldsymbol{\theta}}_{k/k-1}) \quad (2.49)$$

où  $\mathbf{K}_k$  est un gain de Kalman sera effectué selon les équations notées (2.50):

$$\begin{aligned} \mathbf{K}_k &= \mathbf{P}_{k/k-1} \mathbf{C}_k^T (\mathbf{C}_k \mathbf{P}_{k/k-1} \mathbf{C}_k^T + r)^{-1} & \mathbf{P}_{k/k} &= E[(\boldsymbol{\theta}_k - \hat{\boldsymbol{\theta}}_{k/k})(\boldsymbol{\theta}_k - \hat{\boldsymbol{\theta}}_{k/k})^T] \\ \mathbf{P}_{k/k-1} &= \mathbf{A} \mathbf{P}_{k-1/k-1} \mathbf{A}^T + \mathbf{Q} & \mathbf{P}_{k/k-1} &= E[(\boldsymbol{\theta}_k - \hat{\boldsymbol{\theta}}_{k/k-1})(\boldsymbol{\theta}_k - \hat{\boldsymbol{\theta}}_{k/k-1})^T] \\ \mathbf{P}_{k/k} &= \mathbf{P}_{k/k-1} - \mathbf{K}_k \mathbf{C}_k \mathbf{P}_{k/k-1} \end{aligned} \quad (2.50)$$

où  $P_{k/k-1}$  et  $P_{k/k}$  sont les matrices de covariance des erreurs d'estimation respectivement a priori et a posteriori,;  $Q = E[w_i w_i^T]$  et  $r = E[v_i v_i^T]$  sont respectivement des variances assignées aux bruit d'état et de mesure [68].



**Fig. 2.7** Image complète du KF [68]

## 2.8 Validation numérique du modèle flou

Pour évaluer la qualité de l'approximation obtenue par les modèles flous TS, nous utilisons le critère de la Racine de l'Erreur Quadratique Moyenne (RMSE) qui est une mesure globale sur le nombre total de points de l'écart par rapport à la valeur attendue. Sa valeur optimale est zéro; ce critère est défini par l'expression (2.51) :

$$RMSE = \sqrt{\sum_{k=1}^N \frac{1}{N} (y_k - \hat{y}_k)^2} \quad (2.51)$$

où  $1 \leq k \leq N$  est le nombre de points considérés pour la modélisation,  $y$  est la sortie mesurée et  $\hat{y}$  est la sortie du modèle.

## 2.9 CONCLUSION

Ce chapitre nous a permis de faire le tour d'horizon des principes théoriques nécessaires à la compréhension de la modélisation et de l'identification floues des systèmes en utilisant l'approche par clustering à partir des données entrée-sortie.

Après avoir introduit des concepts de base concernant la structure générale et les différents types de modèles flous, nous avons étudié plus particulièrement le modèle de type

TS dans lesquels le conséquent des règles utilise des variables numériques sous la forme des fonctions TS.

Ensuite, nous avons abordé la méthodologie générale d'identification de modèles TS à partir des données, en décrivant les différentes étapes depuis la conception de l'expérience et l'acquisition des données jusqu'à la validation du modèle final. En particulier nous avons mis l'accent sur quelques aspects généraux et communs aux algorithmes traités, afin de passer de l'application des techniques de clustering à l'obtention du modèle flou suivant les aspects suivants :

- Génération des fonctions d'appartenance des antécédents ce qui permet, l'obtention des règles en utilisant le mécanisme de projection sur les axes d'entrée.
- Obtention des paramètres des conséquents, en utilisant des techniques de type moindres carrés et filtre de Kalman.
- Validation numérique du modèle flou.

# Identification des modèles flous type-2

### 3.1 Introduction

Nous avons vu à partir de l'étude bibliographique que les techniques de clustering flou sont des outils puissants pour la reconnaissance des formes à partir des données. Dans le cadre de la modélisation floue des systèmes, notre intérêt porte sur l'obtention des modèles Takagi-Sugeno. Nous avons vu ainsi, une description d'une méthodologie générale pour la construction des modèles flous de type TS, en mettant l'accent sur les besoins communs telle que la génération des fonctions d'appartenance des antécédents. Dans ce qui suit, on s'intéresse à la construction de ces dernières par une méthode qui les enveloppe et les généralise par projection (les fonctions des valeurs intervalles).

Dans les sections suivantes une présentation générale sur les systèmes flous de type-2, ainsi que la méthode de construire les fonctions à partir la projection des clusters des données seront étudiées. En conséquence, la fiabilité de la méthode proposée est validée par des exemples et des simulations, à la fin de ce chapitre le modèle obtenu pour contrôler un système discret est utilisé.

### 3.2 Logique floue type-2

Le concept des ensembles flous type-2 a été introduit par Zadeh [69], comme extension du concept de l'ensemble flou ordinaire appelé ensemble flou type-1. Un ensemble flou type-2 est caractérisé par une fonction d'appartenance floue où le degré d'appartenance de chaque élément de l'ensemble est un ensemble flou dans  $[0, 1]$ . De tels ensembles peuvent être utilisés dans les situations où l'existence de l'incertitude sur les valeurs d'appartenance elles-mêmes. L'incertitude peut être soit dans la forme de la fonction d'appartenance ou dans l'un de ses paramètres [70], Les ensembles flous type-1 peuvent être considérés comme une approximation du premier ordre de l'incertitude, alors que les ensembles flous type-2 sont considérés comme une approximation du deuxième ordre [71][72].

La structure d'un système flou type-2 est représentée par la (fig.3.1). Dans cette section les fonctions d'appartenance des prémisses et des conséquences sont supposées de type-2.

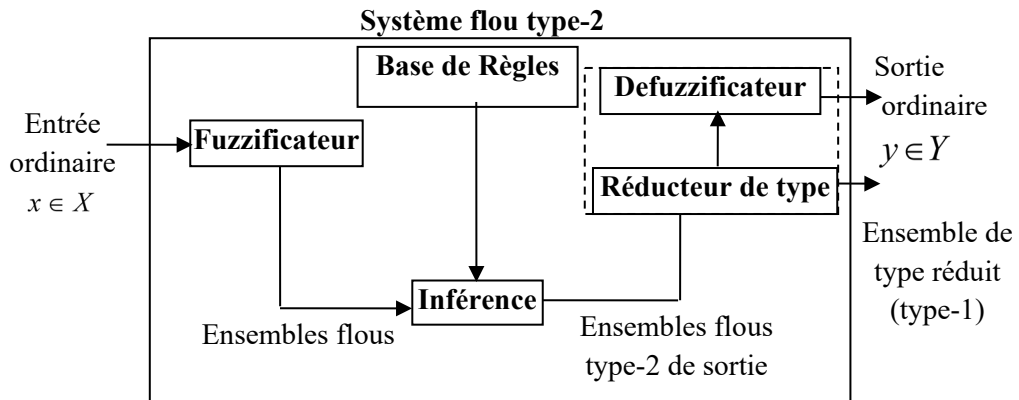


Fig. 3.1 Structure d'un système flou type-2 [72],

### 3.2.1 Fuzzification

La fonction d'appartenance type-2 donne plusieurs degrés d'appartenance pour chaque entrée dont l'incertitude sera mieux représentée. Pour illustrer cet aspect, nous allons considérer une fonction gaussienne avec une incertitude au niveau de la variance et une incertitude au niveau du centre, figure (fig.3.2. a et b).

Afin de faciliter le calcul, nous ne prenons que deux degrés ; le plus grand et le plus petit. Pour une entrée  $x$  nous aurons  $\mu_{\tilde{A}}(x) = [\underline{\mu}_{\tilde{A}}(x) \ \overline{\mu}_{\tilde{A}}(x)]$  où  $\underline{\mu}_{\tilde{A}}(x)$  et  $\overline{\mu}_{\tilde{A}}(x)$  représentent respectivement la valeur minimale et maximale de l'intervalle d'activation correspondant à l'entrée  $x$ .

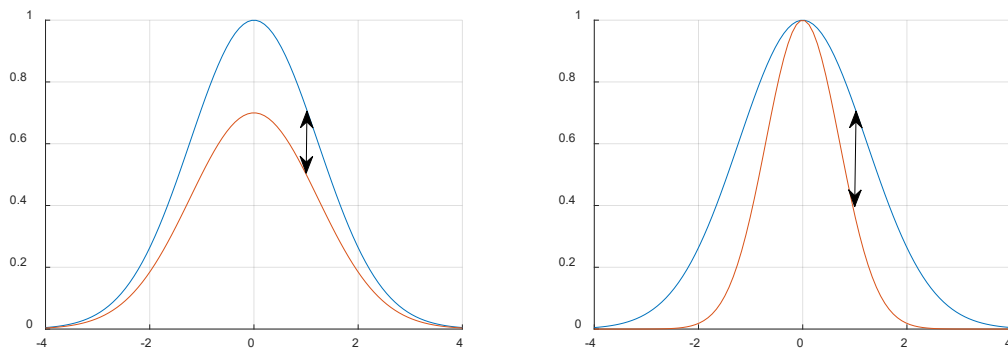


Fig. 3.2 Ensemble flou type-2 a) Incertitude de variance, si  $x = 1$  alors  $\mu_{\tilde{A}}(1) = [0.5 \ 0.7]$   
 b) incertitude de valeur moyenne,  $\mu_{\tilde{A}}(1) = [0.4 \ 0.69]$

### 3.2.2 Règles d'inférence

Si on désire faire une comparaison entre les règles d'un système de type-1 et celles de type-2, la différence résidera seulement dans la nature des fonctions d'appartenance. Ainsi, la structure des règles dans le cas du type-2 va rester exactement la même que celle du type-1. La seule différence étant que quelques (ou toutes) les fonctions d'appartenance seront de type-2, alors, la  $j^{\text{ème}}$  règle R d'un système flou type-2 aura la forme [70] et [71] :

$$R_j: \text{ Si } x_1 \text{ est } \tilde{F}_1^j \text{ et } x_2 \text{ est } \tilde{F}_2^j \dots \text{ et } x_p \text{ est } \tilde{F}_p^j \text{ Alors } y \text{ est } \tilde{G}_i^j \quad (3.1)$$

Où  $x_1 \in X_1, x_2 \in X_2, \dots, x_p \in X_p$  sont les entrées, les  $\tilde{F}_i^j$  sont les ensembles des prémisses tel que pour  $i = 1, 2, \dots, p$ ,  $y \in Y$  est la sortie et les  $\tilde{G}_i^j$  sont les ensembles des conséquences.

Notons qu'il n'est pas nécessaire que toutes les fonctions d'appartenance des prémisses et des conséquences soient de type-2. Il suffit qu'une seule fonction d'appartenance dans une prémisse ou dans une conséquence le soit pour que tout le système soit de type-2.

### 3.2.3. Inférence d'un système flou type- 2

L'inférence dans un système flou type-2 utilise la base de règles floues (3.1) pour effectuer une relation entre le vecteur d'entrée  $(x_1, x_2, \dots, x_p)$  et la sortie  $y$  [71]. La première étape dans l'opération d'inférence floue est le calcul de l'intervalle d'activation associé au  $j^{\text{ème}}$  ensemble flou de la sortie :

$$F(\underline{x}) = \prod_{i=1}^p \mu_{\tilde{F}_i^j}(x_i) \quad (3.2)$$

Où  $\mu_{\tilde{F}_i^j}(x_i)$  est l'intervalle d'activation associé à la variable  $x_i$ .

Si on note l'ensemble flou de sortie correspondant à la  $j^{\text{ème}}$  règle  $R^j$  par  $\tilde{B}$ , avec une entrée  $\underline{x}'$  appliquée et la fuzzification est de type singleton ; l'ensemble  $\tilde{X}'$  auquel appartient  $\underline{x}'$  possède un degré d'appartenance unitaire à  $\underline{x} = \underline{x}'$  et zéro ailleurs. Par conséquent, cet ensemble de sortie correspondant à la  $j^{\text{ème}}$  règle est calculé comme suit, en utilisant l'implication minimum ou produit (l'opération *meet* avec T-norm minimum ou produit dans le cas du type-2):

$$\mu_{\tilde{B}^j}(y) = \mu_{\tilde{G}^j}(y) \cap \prod_{i=1}^p \mu_{\tilde{F}_i^j}(x_i)$$

avec  $\cap$  dénote l'opération *meet* et l'intervalle d'activation associé à la  $j^{\text{ème}}$  règle est donné par:

$$F^j(\underline{x}) = [\underline{f}^j(\underline{x}), \bar{f}^j(\underline{x})] \quad (3.3)$$

$$\begin{aligned} \underline{f}^j(\underline{x}) &= \underline{\mu}_{\tilde{F}_1^j}(x_1) * \underline{\mu}_{\tilde{F}_2^j}(x_2) * \dots * \underline{\mu}_{\tilde{F}_p^j}(x_p) = \prod_{i=1}^p \underline{\mu}_{\tilde{F}_i^j}(x_i) \\ \bar{f}^j(\underline{x}) &= \bar{\mu}_{\tilde{F}_1^j}(x_1) * \bar{\mu}_{\tilde{F}_2^j}(x_2) * \dots * \bar{\mu}_{\tilde{F}_p^j}(x_p) = \prod_{i=1}^p \bar{\mu}_{\tilde{F}_i^j}(x_i) \end{aligned} \quad (3.4)$$

Les termes  $\underline{\mu}_{\tilde{F}_i^j}(x_i)$  et  $\bar{\mu}_{\tilde{F}_i^j}(x_i)$  sont respectivement les valeurs inférieure et supérieure de l'intervalle d'activation correspondant à  $\mu_{\tilde{F}_i^j}(x_i)$ .

### 3.2.4. Réduction de Type

Etant donné que la sortie du système d'inférence est un ensemble flou type-2, elle doit être alors réduite avant qu'elle soit défuzzifiée. Pour transformer la sortie floue type-2 en un ensemble flou type-1, la méthode des centres des ensembles (center of sets « cos ») est utilisée [71] [73]. L'expression de l'ensemble flou de type réduit par cette dernière est donnée par :

$$Y_{\text{cos}}(Y^1, \dots, Y^k, F^1, \dots, F^k) = \int_{y^1} \dots \int_{y^k} \int_{f^1} \dots \int_{f^k} 1 \left/ \frac{\sum_{j=1}^k y^j f^j}{\sum_{j=1}^k f^j} \right. \quad (3.5)$$

Etant donné que chaque ensemble dans l'équation (3.5) est un ensemble d'intervalle type-1, alors  $Y_{\text{cos}}(Y^1, \dots, Y^k, F^1, \dots, F^k)$  est aussi un ensemble d'intervalle type-1 dont le domaine est situé sur l'axe des réels tel que :

$$Y_{\text{cos}}(Y^1, \dots, Y^k, F^1, \dots, F^k) = [y_l, y_r] \quad (3.6)$$

Où  $y_l$  et  $y_r$  sont deux points gauche et droite caractérisant l'ensemble réduit  $Y_{\text{cos}}$ .

$f^j(\underline{x})$  est le  $j^{\text{ème}}$  élément de l'intervalle d'activation de (3.6) et  $y^j$  est un élément de l'intervalle type-2 avec  $Y^j = [y_l^j, y_r^j]$ .

Afin de calculer les points extrêmes  $y_l$  et  $y_r$ , Karnik et Mendel [73][74] ont développé un algorithme itératif qui permet d'obtenir un intervalle d'incertitude pour le centre de gravité d'un système flou dans un intervalle de type-2 donné par la figure (Fig.3.3).

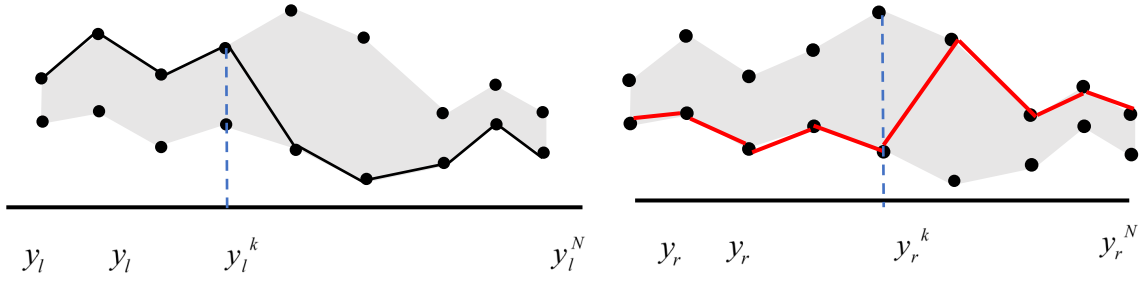


Fig. 3.3: Illustration des points de commutation pour le calcul de  $y_l$  et  $y_r$  [71]

Les points candidats sont obtenus sont donnée par les équations (3.8) et (3.9)

$$y_l(k) = \frac{\sum_{j=1}^k \underline{y}^j \bar{f}^j + \sum_{j=k+1}^K \bar{y}^j \underline{f}^j}{\sum_{j=1}^k \underline{f}^j + \sum_{j=k+1}^K \underline{f}^j} \quad (3.8)$$

$$y_r(k) = \frac{\sum_{j=1}^k \bar{y}^j \underline{f}^j + \sum_{j=k+1}^K \underline{y}^j \bar{f}^j}{\sum_{j=1}^k \underline{f}^j + \sum_{j=k+1}^K \bar{f}^j} \quad (3.9)$$

où  $k$  est un entier dans  $[1, K - 1]$ ,  $K$  est le nombre de points de discrétisation.

Ainsi, les limites optimales de l'intervalle peuvent être obtenues par  $y_l$  et  $y_r$ , données par l'équation (3.10) :

$$y_l(k) = \min_{k \in [1, M-1]} y_l(k) \equiv y_l(L) \equiv \frac{\sum_{j=1}^L \underline{y}^j \bar{f}^j + \sum_{j=L+1}^K \bar{y}^j \underline{f}^j}{\sum_{j=1}^L \underline{f}^j + \sum_{j=L+1}^K \underline{f}^j} \quad (3.10)$$

$$y_r(k) = \max_{k \in [1, M-1]} y_r(k) \equiv y_r(R) \equiv \frac{\sum_{j=1}^R \bar{y}^j \underline{f}^j + \sum_{j=R+1}^K \underline{y}^j \bar{f}^j}{\sum_{j=1}^R \underline{f}^j + \sum_{j=R+1}^K \bar{f}^j}$$

avec  $L$  et  $R$  sont des points de commutation satisfaisant les hypothèses suivantes :

$$y^L \leq y_l \leq y^{(L+1)} \quad y^R \leq y_r \leq y^{(R+1)} \quad (3.11)$$

### 3.2.5. Défuzzification

Le type réduit sera déterminé par ses deux points extrêmes de droite et de gauche respectivement  $y_l$  et  $y_r$ . En appliquant le centre de gravité au type réduit, la sortie (3.11), [75][76] [71]:

$$Y = \frac{y_r + y_l}{2} \quad (3.12)$$

$$\text{Ou } y_l = \frac{\sum_{j=1}^k y_l^j f_l^j}{\sum_{j=1}^k f_l^j} = \sum_{j=1}^k y_l^j \xi_l^j = y_l^T \xi_l(x) \quad y_r = \frac{\sum_{j=1}^k y_r^j f_r^j}{\sum_{j=1}^k f_r^j} = \sum_{j=1}^k y_r^j \xi_r^j = y_r^T \xi_r(x)$$

ou  $f_l^j$  et  $f_r^j$  design le degré d'activation ( soit  $\underline{f}^j$  ou  $\bar{f}^j$ ) contribuant au point extrême gauche et droite  $y_l$  et  $y_r$ , Ou  $\left[ \begin{matrix} \xi_l^j(x) \\ \xi_r^j(x) \end{matrix} \right] = \left[ \frac{f_l^j}{\sum_{j=1}^k f_l^j} \quad \frac{f_r^j}{\sum_{j=1}^k f_r^j} \right]$  sont les composant du premier vecteur de

$\xi_l^T = [\xi_l^1, \dots, \xi_l^k]$ , et  $\xi_r^T = [\xi_r^1, \dots, \xi_r^k]$ ,  $y_l^T = [y_l^1, \dots, y_l^k]$  et  $y_r^T = [y_r^1, \dots, y_r^k]$  sont la conclusion du système flou type-2. Finalement, l'équation après substitution de et peut être donnée par :

$$Y = \frac{y_l^T \xi_l + y_r^T \xi_r}{2} \quad (3.13)$$

### 3.3. Systèmes flous TS de type-2

Les fonctions conséquentes sont des polynômes avec des paramètres de nombre précis défini comme suit :

$$R_i : \text{Si } x_1 \text{ est } \tilde{F}_{i1} \text{ et...et } x_j \text{ est } \tilde{F}_{ij}, \text{ Alors } y_i = C_{i1} x_1 + \dots + C_{ij} x_j(k) \quad (3.14)$$

où  $R_i$  représente la  $i^{\text{ème}}$  règle floue,  $\tilde{F}_{ij}$  sont des ensembles flou de type-2,  $C_{ij}$  présente les paramètres polynomiaux conséquents,  $i = [1, \dots, M]$  où  $M$  est le nombre de règles floues,  $j = [1, \dots, N]$  avec  $N$  représentant le nombre d'antécédents et  $y_i$  est la sortie de la règle.

Comme le principe de réduction de type est déjà présenté dans le paragraphe § 3.2.4, Les équations de réduction de type TS s'écrivent comme suit :

$$\begin{cases} y_l = \sum_{j=1}^L \bar{\xi}^j y^j + \sum_{j=L+1}^K \underline{\xi}^j y^j \\ y_r = \sum_{j=1}^L \underline{\xi}^j y^j + \sum_{j=L+1}^K \bar{\xi}^j y^j \end{cases} \quad \dots \quad (3.15)$$

$$\text{ou : } \frac{\bar{f}^j}{\sum_{j=1}^L f^j} = \bar{\xi}^j \text{ et } \frac{f^j}{\sum_{j=L+1}^K f^j} = \underline{\xi}^j$$

On pose  $(\mathbf{C}^j)^T = [C_1^j \quad C_2^j \quad \dots \quad C_i^j]$  et  $\mathbf{x} = [x_1, \dots, x_i]$  l'équation (3.15) de  $y_l$  devient alors

$$y_l = \underbrace{\left[ \bar{\xi}^1 [x_1, \dots, x_i] \quad \dots \quad \bar{\xi}^L [x_1, \dots, x_i], \underline{\xi}^{L+1} [x_1, \dots, x_i] \quad \dots \quad \underline{\xi}^K [x_1, \dots, x_i] \right]}_{\phi} \underbrace{\begin{bmatrix} \mathbf{C}^1 \\ \vdots \\ \mathbf{C}^K \end{bmatrix}}_{\theta} = \phi_l(x)^T \theta$$

Les mêmes étapes à suivre par rapport à  $y_r$  et on obtient :

$$[y_l, y_r] = \theta^T [\phi_l(x) \quad \phi_r(x)] \quad (3.16)$$

Cette écriture nous aide à faire séparer les variables de l'antécédent de celles de permises, afin de les faire adapter dans le cas d'un système de commande adaptative, où  $\theta$  est le paramètre d'adaptation :

$$Y(x, \dot{x}) = \frac{y_r + y_l}{2} = \theta^T \left[ \frac{\phi_l(x, \dot{x}) + \phi_r(x, \dot{x})}{2} \right] \quad (3.17)$$

### 3.4. Approche proposée

#### 3.4.1. Conception des fonctions d'appartenance de type-2

Après la présentation des notions des systèmes flous et des techniques de modélisation floues utilisant les clusters. La technique suivie pour construire les fonctions d'appartenance et le modèle flou TS repose sur les étapes suivantes :

- 1 : Génération de clusters de données floues en utilisant le clustering flou dans l'espace produit cartésien des entrées et des sorties [42].
- 2 : Estimation de la fonction d'appartenance (MF) des antécédents des valeurs d'intervalle utilisant un nouvel Algorithme de Détection d'Enveloppe proposé (EDA) [77].
- 3 : Calcul des paramètres des fonctions conséquentes, en utilisant les moindres carrés pour les entrées et le Filtre Kalman pour les conséquences [42] [77][75].

Par hypothèse, nous avons les fonctions d'appartenance des sous-ensembles flous  $A_{ij}$  de l'antécédent et l'ensemble flou multidimensionnel défini point par point par la  $i^{\text{ème}}$  ligne de la matrice de partition par la projection sur les entrées  $x_j$ . avec  $\mu_{A_j}(x_{jk}) = \text{proj}_j(\mu_{ik})$

Dans le cas des données bruitées ou mal acquises (fig. 3.4); la projection donne une distribution qu'on peut approximer par une fonction paramétrique, faite par une régression.

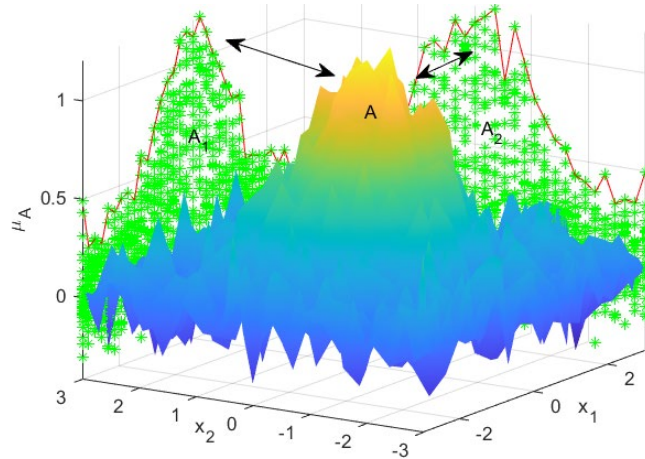


Fig. 3.4 Approximation des données bruitées projetées par une fonction d'appartenance

Notons que la fonction générée par la régression de l'ensemble de nuage des points peut donner plusieurs fonctions de type-1 selon le degré de précision, la marge de fiabilité et même la fiabilité de la méthode de conception des fonctions d'antécédents. En conséquence, l'ensemble de ces fonctions peut être vu comme une Fonction d'appartenance des Valeurs d'Intervalle (Interval Valued Function).

Chaque partitionnement projeté dans l'espace des entrées (régresseurs)  $X$  donne directement une région floue dans laquelle les données peuvent approximer raisonnablement les sous-systèmes linéaires définis par le groupement comme présenté dans la figure (fig. 3.5).

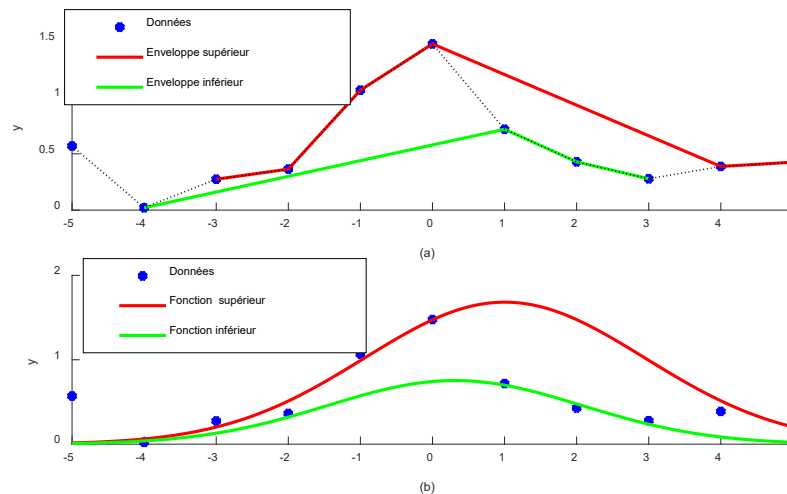


Fig. 3.5 (a) Détection d'enveloppes (b) Régression gaussienne

Comme nous avons vu, les données projetées peuvent être interpréter facilement par l'être humain. L'algorithme de clustering de Gustafson-Kessel est utilisé pour l'estimation des fonctions d'appartenance des permises. Une fois les clusters sont obtenus, on les projette dans

l'espace régresseur. Ce dernier permet d'obtenir les fonctions d'appartenance en forme de nuage [93].

Ainsi, on propose de considérer ce nuage comme une incertitude ou plus précisément l'empreinte d'incertitude (Footprint Of Uncertainty : FOU). Donc, on peut l'extraire d'une fonction d'appartenance de type-2. Ce nuage des points forme une région bornée qu'on peut l'envelopper par deux fonctions, une fonction supérieure (upper fuzzy Membership Function) et une fonction inférieure (lower fuzzy Membership Function). Par la suite, nous proposons nous proposons deux méthodes pour la détection et la séparation des enveloppes

### **3.4.2 Méthodes de détection et de séparation des enveloppes :**

#### **a) Méthode EDA :**

Pour la détection des enveloppes, on propose la méthode EDA basée sur le concept de dérivée. L'EDA comprend deux algorithmes, l'algorithme (EDA1) pour la détection des fonctions d'appartenance inférieure et supérieure et l'algorithme (EDA2) pour séparer les fonctions d'appartenance de la partie prémisses détectées.

##### a) Présentation de l'algorithme EDA1

1. Interpolation des points linéairement (la courbe pointée dans la (fig. 3.5. (a)) entre tous les segments  $[x(i), x(i+1)]$  où  $i$  représente les données d'indices.

2. calcul de la pente entre tous les points successifs  $[y(i), y(i+1)]$  tel que :

$$d(i+1) = \frac{y(i+1) - y(i)}{x(i+1) - x(i)} \quad (3.18)$$

On a deux cas :

(i) si la pente  $d$  dans l'intervalle  $[x(i), x(i+1)]$  est positive, donc le point  $y(i+1)$  est considéré comme un point appartenant à la fonction d'appartenance supérieure  $y_h$  et après le mémoriser avec l'indice  $I_h$  où  $I_h = i+1$ , i.e.  $y_h(I_h) = y(i+1)$

(ii) sinon la pente  $d$  dans l'intervalle  $[x(i), x(i+1)]$  est négative, donc le point  $y(i+1)$  est considéré comme un point de la fonction d'appartenance inférieure  $y_L$  et le mémoriser avec l'indice  $I_L$  où  $I_L = i+1$ , i.e.  $y_L(I_L) = y(i+1)$

L'application de l'algorithme EDA1 sur un ensemble arbitraire est présentée par la Figure (fig 3.6.a.)

La forme des enveloppes inférieure et supérieure obtenues représentent les fonctions d'appartenance de type-2.

Cet algorithme peut mener à un problème d'intersection des fonctions détectées supérieure et inférieure, puisque les points ne sont pas distribués régulièrement, i.e. quelques points paraissent dans le mauvais côté (points singuliers). Dans ce cas, ou on tire la connexion linéaire entre chaque point de l'enveloppe, il y a possibilité que les enveloppes supérieure  $f_U$  et inférieure  $f_L$  se croisent comme indiqué dans la figure 3.6. Pour contourner ce problème on propose dans la suite un deuxième algorithme pour réajuster la classification de ces points singuliers.

a) Présentation de l'algorithme EDA 2

Pour mieux comprendre le mécanisme de réajustement de l'ensemble des points de l'enveloppe supérieure vers l'enveloppe inférieure. Notons qu'entre deux points inférieurs successifs à indices  $[k, k + 1]$ , on peut trouver plusieurs d'autres points supérieurs à indices  $[I_L(k)+1, I_L(k+1)]$ . Dans la prochaine étape, le point à indice  $I_L(k)$  est considérée comme point de référence pour l'ensemble des pentes calculé entre  $I_L(k)$  et tous les points dans l'intervalle  $[I_L(k)+1, I_L(k+1)]$ .

Cet algorithme est donné comme suit :

1. Calcule de la pente de deux points inférieurs successifs avec les indices  $[k, k + 1]$ :

$$d_L(k) = \frac{y_L(I_L(k+1)) - y_L(I_L(k))}{I_L(k+1) - I_L(k)} \quad (3.19)$$

avec  $k = 1, \dots, N_L$ , où  $N_L$  est le nombre total des points inférieurs.

2. Calcule de la pente entre les points fixés  $I_L(k)$  et le point avec indice  $s(k) = I_L(k) + 1$  comme suit :

$$d_k(s) = \frac{y(s) - y(I_L(k))}{s - I_L(k)} \quad (3.20)$$

3. Comparaison entre  $d_k(s)$  et  $d_L(k)$  : si  $d_L(k) > d_k(s)$ , donc  $y_L(k) = y(s)$  et  $I_L(k) = s$ , et après retirer  $y(s)$  de l'ensemble de points supérieurs.
4. Poser  $s(k) = s(k) + 1$  et retour vers l'étape 2 où  $s(k) = I_L(k + 1)$ .

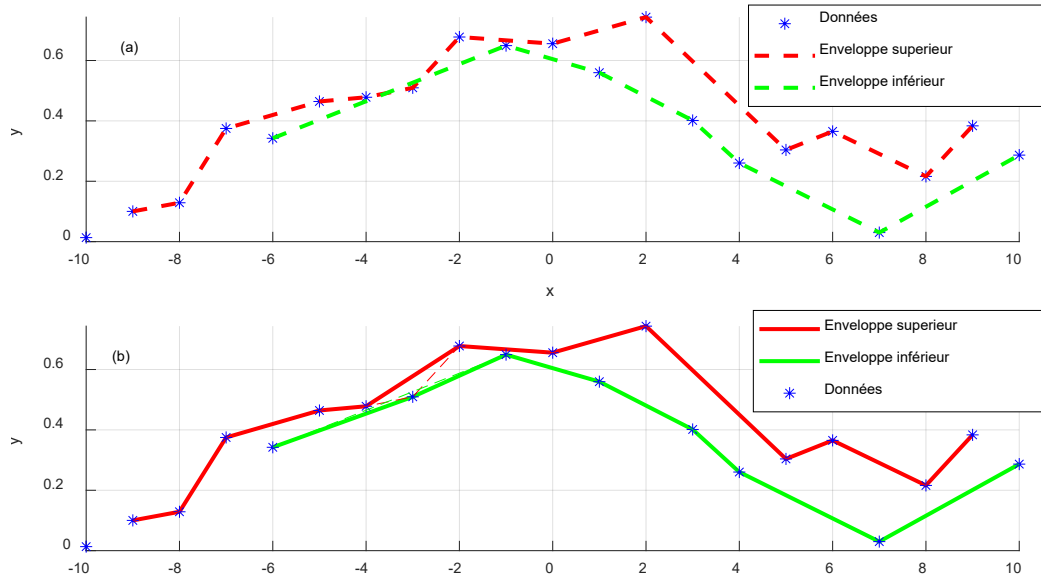


Fig. 3.6. Algorithme EDA2 (a) avant réajustement (b) après réajustement.

Le même algorithme sera appliqué à l'ensemble des points d'enveloppes supérieures pour réajuster les points singuliers.

### b) Méthode EDA1<sup>n</sup>

Une autre technique pour la détection et les séparations des enveloppes par l'application de l'algorithme EDA1 (n) fois. En utilisant l'algorithme EDA1 sur les données initiales, cette étape donne deux ensemble  $\underline{f} = (x_L, y_L)$  et  $\bar{f} = (x_H, y_H)$ . Ensuite, le même algorithme EDA1 est utilisé sur les deux ensembles  $(\underline{f}, \bar{f})$  pour obtenir quatre ensembles comme indiqué dans le diagramme sur la figure (Fig. 3.7) où  $i$  et  $j$  représente le nombre de répétition pour chaque enveloppe.

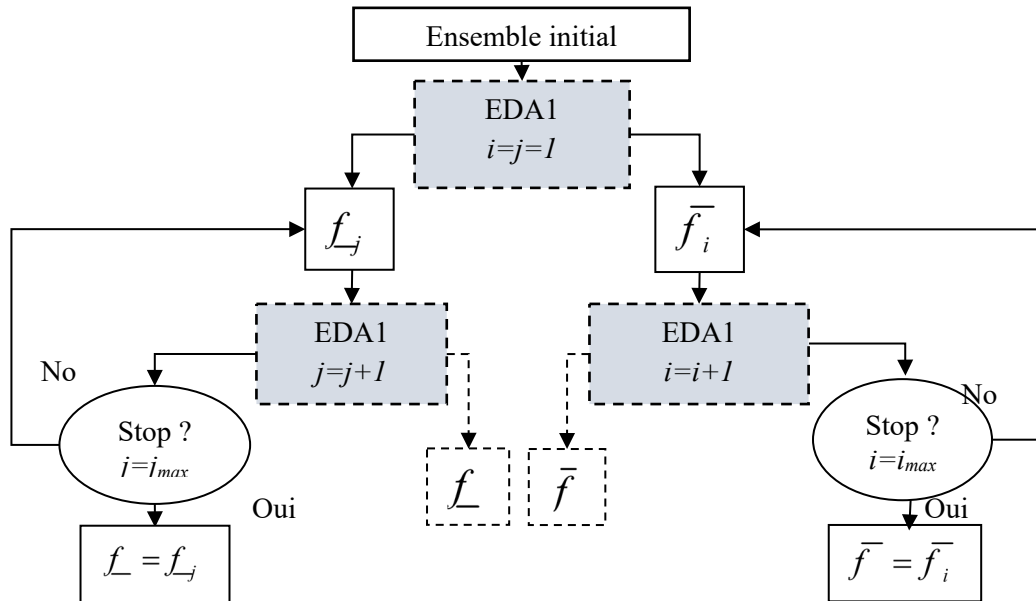


Fig. 3.7 Algorithme proposé par la méthode 2

Cette méthode est plus simple, mais nécessite des points d'échantillonnage plus importants que la première méthode. Les figures (fig. 3.8) montrent les résultats obtenus moyennant l'application de l'EDA pour différent  $i$  et  $j$ .

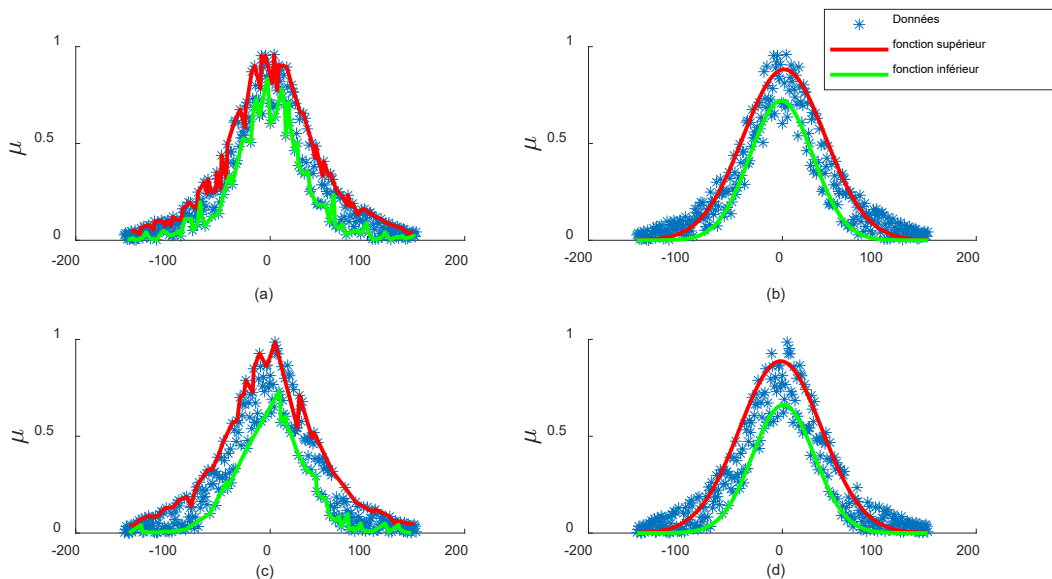


Fig. 3.8 Résultats obtenu après l'application de EDA (a)  $i=2$ , (c)  $i=3$ , (b) et (d) représentent les fonctions gaussiennes paramétrisées

### 3.5. Modélisation floue TS type-2

A cette étape, nous avons deux ensembles de données de points : l'ensemble de données 1 représentant les données de  $f_L$  et l'ensemble de données 2 représentant les données de  $f_U$ . L'utilisation de fonctions d'appartenance triangulaires, gaussiennes ou autres dépendent de nombreux aspects. Par exemple, la différentiabilité favorise l'utilisation des fonctions

d'appartenance gaussienne et polynomiale puisqu'elles présentent des dérivées continues facilitant l'analyse de sensibilité sur le système d'inférence floue obtenu.

Dans ce travail, la fonction d'appartenance gaussienne sera utilisée pour la partie prémisse, par conséquent les ensembles de données respectivement 1 et 2 seront adaptés aux modèles gaussiens qui donneront par la suite les fonctions inférieure et supérieure  $\bar{f}$  et  $\underline{f}$ .

Généralement, une fonction d'appartenance à valeur d'intervalle est caractérisée par les contraintes suivantes :

$$\left. \begin{array}{l} \max \bar{f}_i(x) = 1 \\ \min \underline{f}_i(x) > \varepsilon \end{array} \right\} \text{ Avec } i = 1, 2, \dots, c$$

où  $\varepsilon$  est un nombre positif prédéfini permettant que les fonctions obtenus inférieure et supérieure ne se croisent pas (avec  $\varepsilon = 0,05$  dans ce cas) et  $c$  représente le nombre de clusters

L'ajustement du modèle gaussien représente un problème de régression non linéaire donné par :

$$\hat{y}_i(x, a_i, b_i, c_i) = a_i e^{\frac{-(x - c_i) b_i}{2}} = f_i(x) \text{ Avec } i = 1, 2, \dots, c$$

Avec  $y_i$  la sortie du  $i^{\text{ème}}$  modèle gaussien non linéaire,  $\varphi_i = [a_i \ b_i \ c_i]$  est le vecteur de paramètres à optimiser (hauteur, largeur et centre) et  $f$  est le modèle gaussien obtenu qui peut être  $\bar{f}_i$  ou  $\underline{f}_i$ .

Les vecteurs de paramètres  $i$  seront estimés par les moindres carrés non linéaires afin d'ajuster les ensembles de données supérieure et inférieure  $f_L$  et  $f_U$  respectivement aux modèles gaussiens  $\bar{f}_i$  ou  $\underline{f}_i$

Le modèle global peut être commodément représenté par un ensemble de règles  $R_i$  de type TS comme suit :

$$R_i: \text{ Si } \mathbf{x} \text{ est } \tilde{\mathbf{A}}_i \text{ alors } y_i = \mathbf{C}_i \mathbf{x} = C_{i1} x_1 + \dots + C_{ij} x_j$$

où  $x \in R^p$  est l'entrée, et  $y \in R$  est la sortie,  $\tilde{\mathbf{A}} = [\bar{f}_i, \underline{f}_i]$  la  $i^{\text{ème}}$  fonction d'appartenance floue des valeurs d'intervalle de la partie prémisse obtenue.

les vecteurs des paramètres sont estimés en utilisant la méthode du filtre de kalman dont le principe de cette méthode a été déjà présenté dans (§2.7.2).

### 3.6. Simulation

On considère quelques exemples d'approximation et d'identifications et de commande pour tester et valider la méthode proposée. Quelques résultats de l'approximation des fonctions statiques non linéaires, des systèmes dynamiques sont présentés. Dans la suite on note IVFC (Interval value fuzzy control) et IVFM (Interval value fuzzy Model) pour la commande basé sur le system flou d'intervalle et le systeme flou type-2, d'autre part on note T1FC et T1FM pour les commandes et les modèles de type-1.

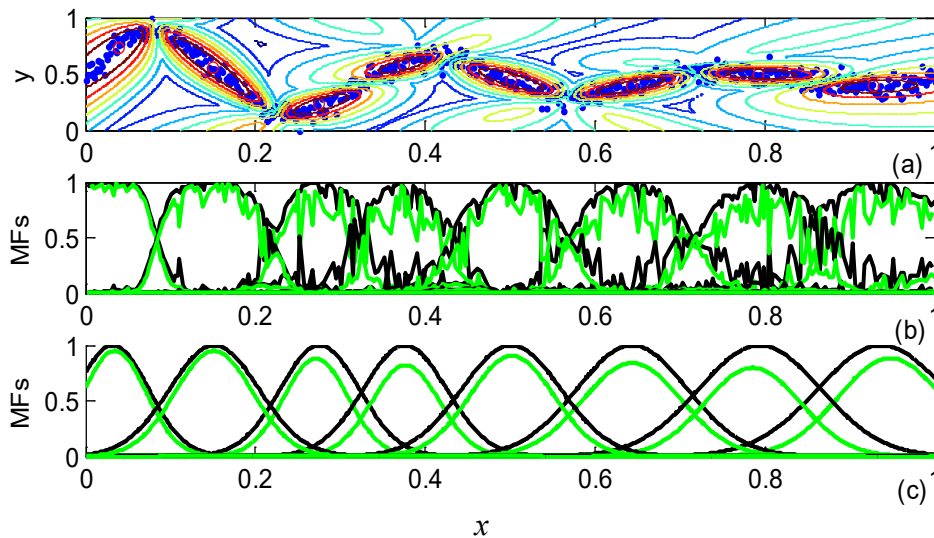
### 3.6.1 Approximation de fonctions non linéaires

**1<sup>er</sup> exemple :** Considérant le problème d'approximation de la fonction non linéaire suivante :

$$y = 10e^{-0.05x} \sin(0.1x) + v \quad \text{où } x \in [0, 30] \quad (3.21)$$

Où  $v$  est un bruit blanc Gaussien avec une variance  $\text{Var}(v) = 0,1$  sur un intervalle de  $[0, 1]$  avec 301 points générés pour l'apprentissage.

Le résultat obtenu par la Fig. 3.9 (a), montre les huit (08) clusters détectés et à partir de l'application de l'algorithme EDA,  $f_L$  et  $f_U$  sont identifiées dans la figure (fig. 3.9 b et c).



**Fig. 3.9** (a) GK clustering (b) EDA1-EDA2 détection des enveloppes  
 (c) Estimation des prémisses de type-2.

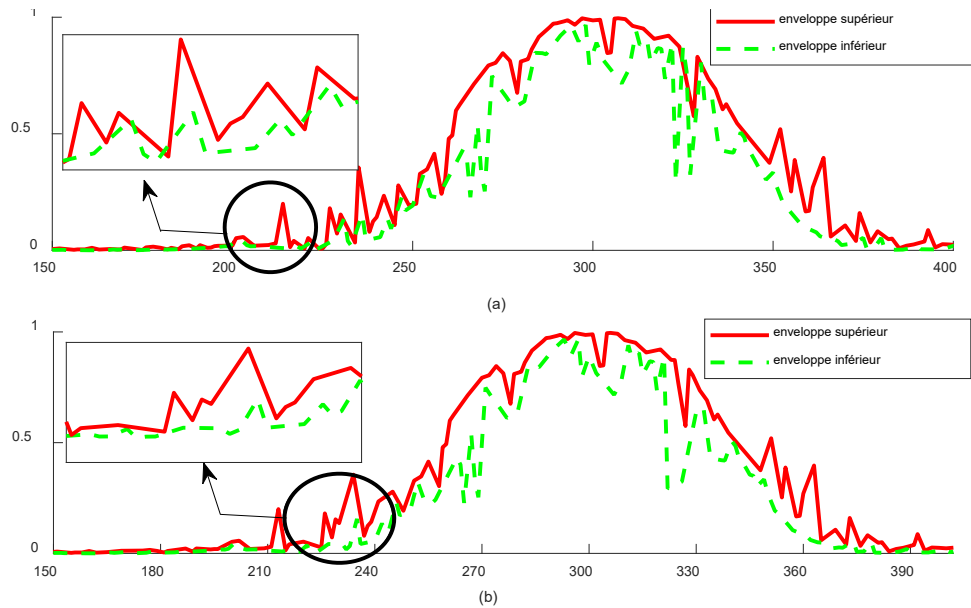


Fig. 3.10 EDA1 détection des enveloppes (b) EDA2 corrections des intersections

### Interprétation des résultats

Dans cet exemple les enveloppes sont croisées, ce problème est corrigé par EDA2, le résultat est présenté clairement dans la figure (fig. 3.10).

Les conséquences sont estimés en utilisant FK et les résultats sont présentés dans la figure (fig. 3.11). Les règles du modèle TS flou sont indiquées dans la Table 3.2. La figure (fig. 3.12) montre la comparaison entre les sorties de type-2 et de type-1. On voit clairement que la fonction approximée par notre approche est plus proche de la fonction réelle que la fonction obtenue par la technique de type-1, ce qui montre l'efficacité et reflète la résolution des modèles flous de type-2.

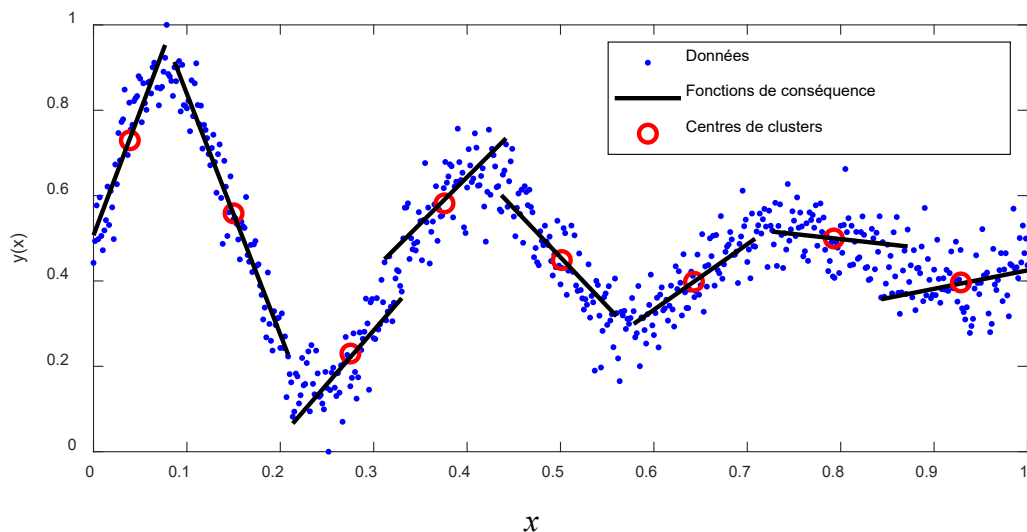


Fig. 3.11 Fonctions de conséquences

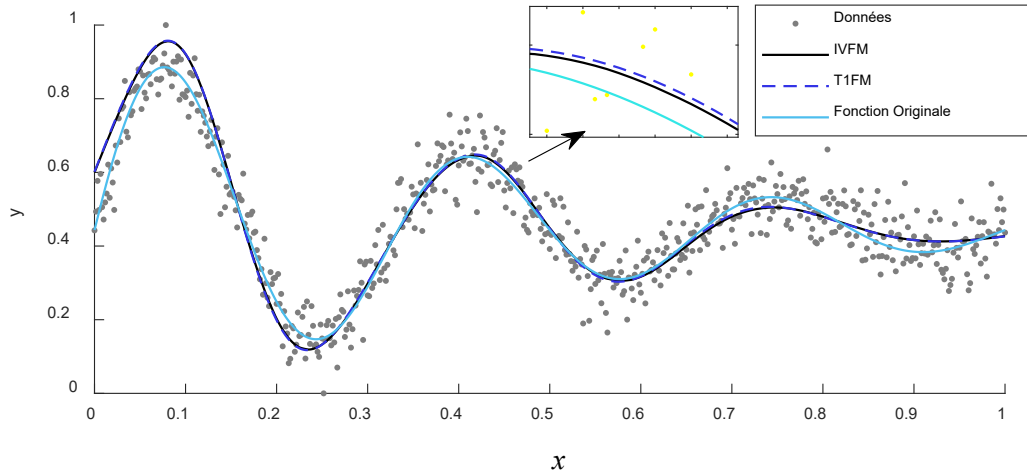


Fig. 3.12 Modélisation et comparaison entre T1FM et IVFM

L'utilisation du critère MSE pour la comparaison quantitative pour les deux cas de simulation, montre la domination du modèle flou type-2 par rapport au modèle flou type-1 pour une identification de 8 et 9 règles.

Fonctions d'appartenance	Type	MSE
8 MF	Type-1 TS	0.7076
	Type-2 TS	0.6708
9 MF	Type-1 TS	0.6763
	Type-2 TS	0.6263

Table. 3.2 Comparaison entre l'approche proposée et le modèle flou Type-1

$R_i$	Si $x \in \tilde{A}_i = [f_i, \bar{f}_i]$ Alors	$y_i =$
1	$[\exp(\frac{x-0.0321}{0.0598})^2, 0.95\exp(\frac{x-0.0345}{0.0523})^2]$	$9.594x + 0.477$
2	$[\exp(\frac{x-0.1517}{0.0744})^2, 0.95\exp(\frac{x-0.1513}{0.0650})^2]$	$1.612 - 10.49x$
3	$[\exp(\frac{x-0.2758}{0.0658})^2, 0.879\exp(\frac{x-0.2715}{0.0536})^2]$	$5.026x - 0.8377$
4	$[\exp(\frac{x-0.3762}{0.0694})^2, 0.82\exp(\frac{x-0.3771}{0.0563})^2]$	$1.131 - 2.125x$
5	$[\exp(\frac{x-0.5015}{0.0824})^2, 0.90\exp(\frac{x-0.5024}{0.0699})^2]$	$0.9742x - 0.1043$
6	$[\exp(\frac{x-0.6439}{0.0922})^2, 0.84\exp(\frac{x-0.6440}{0.0821})^2]$	$0.4615 - 0.1357x$
7	$[\exp(\frac{x-0.7943}{0.1017})^2, 0.797\exp(\frac{x-0.7861}{0.0796})^2]$	$0.559 - 0.314x$
8	$[\exp(\frac{x-0.9378}{0.1059})^2, 0.88\exp(\frac{x-0.9482}{0.0863})^2]$	$0.4786 - 0.1395x$

Table. 3.3 Règles du modèle flou TS de type-2

**2<sup>ème</sup> exemple :** Dans cet exemple, on va approximer une fonction sinc(x). Cette dernière a été présentée dans plusieurs travaux d'identification [11-29]. L'ensemble de données est de 121 points qui sont uniformément échantillonnées entre [-10,10].

$$y(x) = \text{sinc}(x) + v \quad (3.22)$$

$v$  est un signal aléatoire uniformément distribuée dans  $[-0.1, 0.1]$ .

Le modèle choisi est construit selon 8 règles correspondant à 8 clusters, avec des fonctions d'appartenance gaussiennes. Les résultats de simulation sont présentés dans la figure (fig. 3.13) où le critère obtenu MSE est d'environ 0.0047. La méthode est comparée avec d'autres méthodes, voir table (3.4) et à partir de laquelle on constate que notre approche présente une erreur mineure.

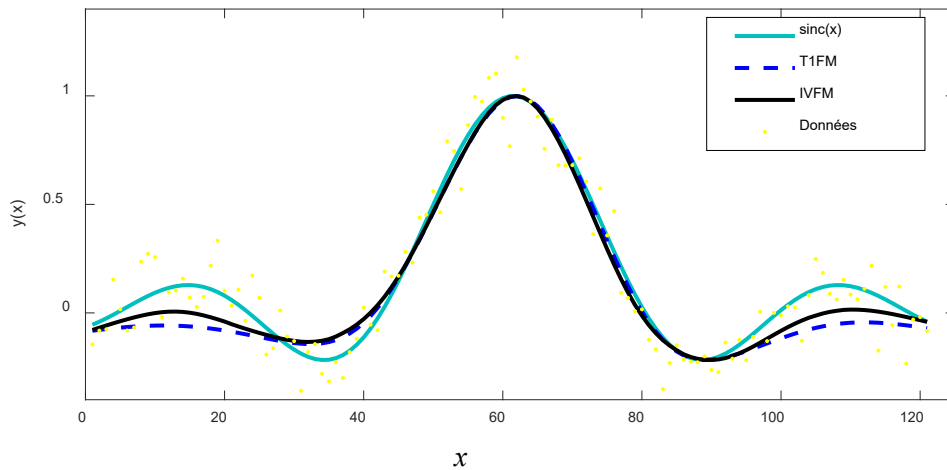


Fig. 3.13 Approximation de la fonction sinc(x)

Méthode	Nombre de règles	MSE
SCM [78]	2	0.0447
EUM [78]	2	0.0450
EFCM [78]	2	2 0.0089
Fazel Zarandi [79]	4	0.0238
T1FM [77]	8	0.0085
IVFM [77]	8	0.0047

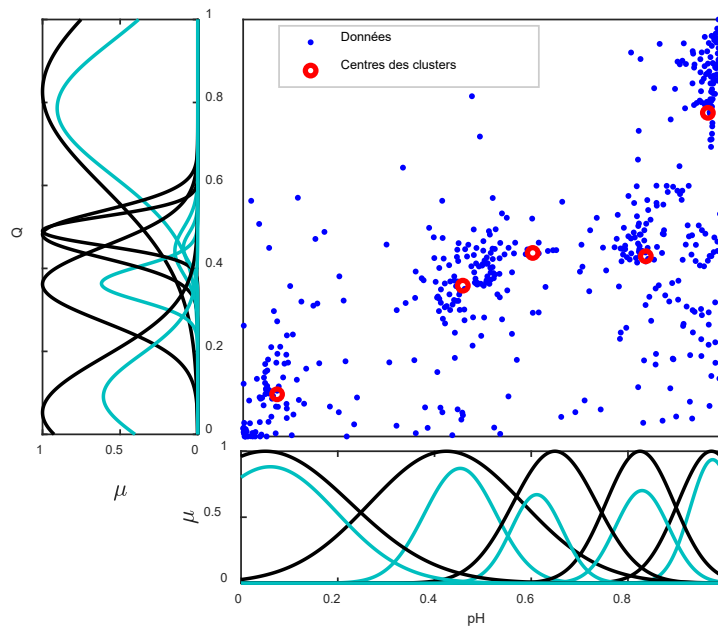
Table 3.4: Comparaison des différents modèles pour la fonction sinc(x) [77]

**3.6.2. Modélisation de procédés de neutralisation du pH :** La commande des procédés de régulation de pH joue un rôle très important dans l'industrie des procédés. La neutralisation d'une solution est un cas particulier, mais important de la régulation de pH. Le contrôle de pH est connu pour être un problème difficile ; en effet, obtenir une commande robuste et performante est souvent difficile à cause des variations non linéaires des caractéristiques du procédé. D'une manière générale, pour la neutralisation du pH, l'identification parfait de procédé nous permet de comprendre et de gouverner le système d'une manière plus fiable.

Le système est un bac de neutralisation avec trois flux influents (acide, buffer et base) et un flux effluent. L'ensemble des données est obtenu par simulation de modèle de Hall and Seborg [96] pour le changement aléatoire de débit d'influent de base  $Q$ . le flux d'influent buffer et le flux d'influent d'acide sont constants. La sortie est le pH du mélange dans le bac. Les données d'identification sont de  $N = 480$  échantillons avec un temps d'échantillonnage égale à 15 s. Le processus est approximé par un modèle discret de premier ordre :

$$pH(k+1) = f(pH(k), Q(k)) \quad (3.23)$$

Où  $k$  dénote l'instant d'échantillonnage, et  $f$  est une fonction inconnue qu'on veut l'approximer avec IVFM. La Figure (fig. 3.14) donne une image sur les 5 clusters détectés dans  $\mathbb{R}^2$ . Le partitionnement des données est présenté dans la matrice de partitionnement  $U = (\mu_{ij})$  leurs éléments représentent les degrés d'appartenance de vecteur de données dans les clusters flou  $A_i$  avec les centres  $c_1, c_2, c_3, c_4$  et  $c_5$ . Les fonctions d'appartenance d'antécédent sont donc extraites par la projection des clusters vers les espaces des variables  $Q$  et pH comme indiqués dans la figure (fig. 3.14).



**Fig. 3.14:** Clusters et projection pour les données de processus de Neutralisation du pH

Le modèle est construit selon les 5 règles suivant :

Si  $Q(k)$  est  $\tilde{A}_{1,1}$  et  $pH(k)$  est  $\tilde{A}_{1,2}$  Alors  $pH(k+1) = -0.0023 pH(k) + 0.5278Q(k) + 0.2456$

Si  $Q(k)$  est  $\tilde{A}_{2,1}$  et  $pH(k)$  est  $\tilde{A}_{2,2}$  Alors  $pH(k+1) = 0.0258 pH(k) + 0.6972Q(k) + 0.3284$

Si  $Q(k)$  est  $\tilde{A}_{3,1}$  et  $pH(k)$  est  $\tilde{A}_{3,2}$  Alors  $pH(k+1) = -0.0321 pH(k) + 0.9283Q(k) + 0.2068$

Si  $Q(k)$  est  $\tilde{A}_{4,1}$  et  $pH(k)$  est  $\tilde{A}_{4,2}$  Alors  $pH(k+1) = 0.0650 pH(k) + 0.3932Q(k) + 0.8312$

Si  $Q(k)$  est  $\tilde{A}_{5,1}$  et  $pH(k)$  est  $\tilde{A}_{5,2}$  Alors  $pH(k+1) = 0.2189 pH(k) + 0.7468Q(k) + 0.0361$

Les résultats de simulation sont indiqués dans la figure (fig. 3.15) où le MSE obtenu est d'environ 0.3446. La table comparative (table. 3.5) montre la réduction d'erreur de notre méthode.

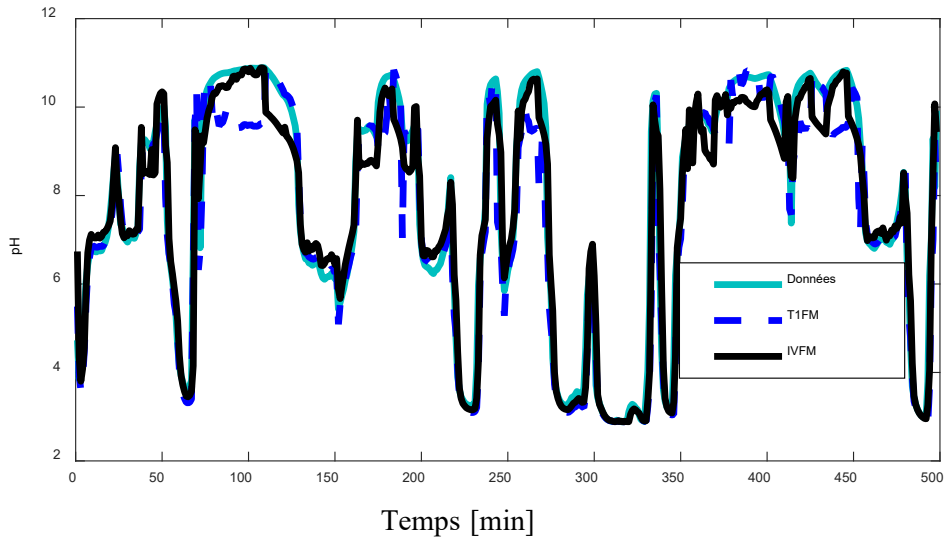


Fig. 3.15 Identification T1FM / IVFM du processus de pH Neutralisation (13).

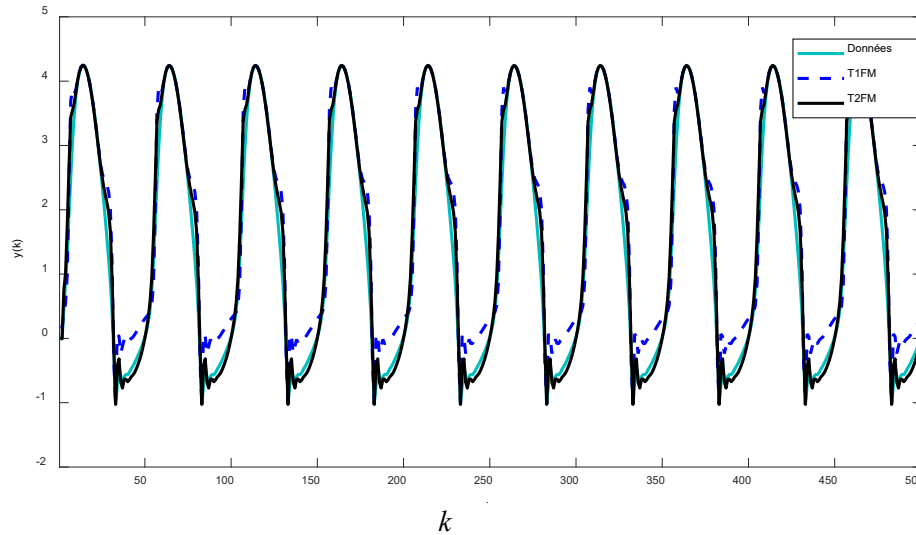
Modèle	Nombre de règles	MSE
IVFM	5	0.3446
T1FM	5	0.5027
IVFM	4	0.655
T1FM	4	0.4290
Neuro-flou [97]	3	0.6724

Table 3.5 Comparaison des différents modèles pour le processus de pH Neutralisation

**3.6.3 Modélisation d'un système dynamique non linéaire :** Testons maintenant l'efficacité de la méthode sur un système dynamique non linéaire du second ordre de la forme  $y(k) = f(y(k-1), y(k-2)) + u(k)$  où :

$$f(y(k-1), y(k-2)) = \frac{y(k-1)y(k-2)(y(k-1)-0.5)}{(1+y(k-1))^2 + y(k-2)^2} \quad (3.24)$$

Nous approchons la composante non linéaire  $f$  avec un modèle flou, avec deux entrées  $y(k-1)$  et  $y(k-2)$ . Les données sont générées à partir du modèle dynamique en appliquant un signal d'entrée sinusoïdale :  $u(k) = \sin(2k/25) + v(k)$  Où  $v(k)$  est un signal aléatoire uniformément distribué sur  $[-1,5 ; 1.5]$ . Le modèle choisi est construit selon un ensemble de 4 règles avec des fonctions d'appartenance gaussiennes. Le résultat de la simulation présenté sur la figure (fig. 3.16) où le MSE obtenu est d'environ 0,0297.



**Fig. 3.16** Identification du modèle dynamique type-1/type-2

La méthode utilisée est comparée à d'autres méthodes et le tableau (table 3.6) montre la réduction d'erreur de la méthode proposée.

Modèle	Nombre de règles	MSE
Sugeno and Yasukawa [98]	12	0.5072
Farag et al.[99]	75	0.0374
Wang and Lee [100]	8	0.6184
Evsuko_ et al. [101]	100	0.1577
Bagis [102]	4	0.0341
modele Type-1	4	0.2107
IVFM [93]	4	0.0297

**Table 3.6:** Comparaison des différents modèles dynamiques non linéaires dynamiques

### 3.6.4 Identification et commande d'un système dynamique

On considère le système non linéaire [103] décrit par l'équation suivante :

$$y(k+1) = g(y(k), y(k-1)) + u(k) \quad \text{où} \quad g(k) = \frac{y(k)y(k-1)[y(k)+2.5]}{1+y^2(k)+y^2(k-1)} \quad (3.25)$$

Où la fonction non linéaire  $g$  est supposée inconnue. Notre objectif est d'obtenir une commande  $u(k)$  pour que la sortie  $y(k)$  du système en boucle fermée suive le signal du modèle de référence  $y_m(k)$ .

$$y_m(k) = 0.6y_m(k-1) + 0.2y_m(k-2) + r(k) \quad (3.26)$$

Si la fonction  $g(y(k), y(k-1))$  est connue, nous pouvons construire la commande comme suit :

$$u(k) = -g(y(k), y(k-1)) + 0.6y(k-1) + 0.2y(k-2) + r(k) \quad (3.27)$$

Le système dynamique (3.25) devient :

$$y(k) = 0.6y(k-1) + 0.2y(k-2) + r(k) \quad (3.28)$$

En combinant (3.26) et (3.28), on obtient :

$$e(k+1) = 0.6e(k-1) + 0.2e(k-2)$$

Où  $e(k) = (y(k) - y_m(k))$  est l'erreur de suivi.

Ainsi,  $\lim_{k \rightarrow \infty} e(k) = \lim_{k \rightarrow \infty} (y(k) - y_m(k)) \rightarrow 0$  ce qui montre la stabilité du système.

Cependant, comme  $g(y(k), y(k-1))$  est inconnue, la commande (3.27) ne peut pas être implémentée. Pour résoudre ce problème, nous remplaçons la fonction  $g$  par son approximation ; le système flou  $\hat{g}(y, \theta_g)$  et la commande deviennent alors :

$$u(k) = -\hat{g}(y, \theta_g) + 0.6y(k) + 0.2y(k-2) + r(k) \quad (3.29)$$

où  $\hat{g}(y, \theta_g)$  est de forme (3.13) et (3.14).

$$\hat{g}(x, \theta_g) = \theta_g^T \phi_g(x)$$

Cependant, pour le système flou de type-1,  $\phi_g(k)$  est calculé à partir de l'équation (2.16).

La sortie de (3.25) devienne alors :

$$y(k+1) = (g - \hat{g}) + 0.6y(k) + 0.2y(k-1) + r(k) = \tilde{g}(k) + 0.6y(k) + 0.2y(k-1) + r(k)$$

Où  $\tilde{g}$  présente l'erreur d'approximation.

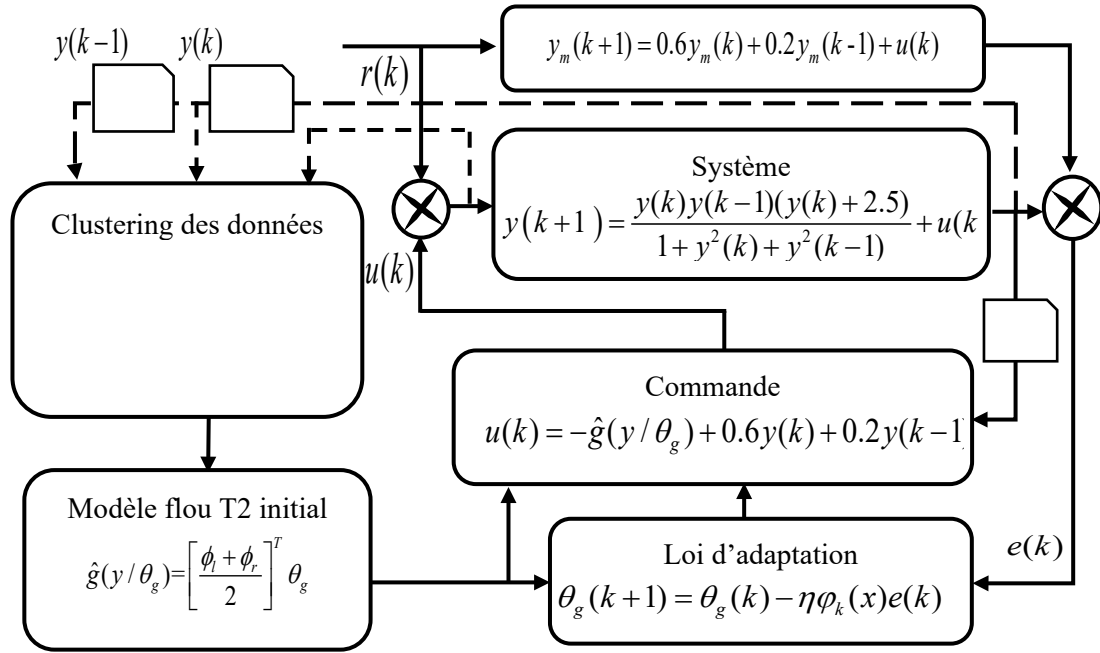


Fig. 3.17 Commande floue indirecte proposée

### Identification hors ligne

Les fonctions inconnues qui sont approximées initialement peuvent-être obtenues soit par un algorithme en ligne ou bien hors ligne. Dans cette étude, on les a obtenus à partir des données hors ligne ensuite on les ajusté en fonction de l'erreur.

Pour la collecte des données d'entrée et de sortie pour l'identification, on applique un signal d'entrée  $u(k)$  telle que :

$$u(k) = \begin{cases} n(k) + 5\sin(2k\pi/350) + 0.5\sin(2k\pi/250) & \text{pour } k < 800 \\ n(k)/5 + \sin(2k\pi/150) & \text{pour } k > 800 \end{cases}$$

Avec  $n(k)$  est un bruit blanc défini par  $\mathcal{N}(0,4)$

La figure (fig. 3.18) présente les données entrées et sorties obtenues. Les variables de permises de système TS flou sont présélectionnées comme :  $x_1(k) = y(k-1)$  et  $x_2(k) = y(k)$ , le nombre de cluster est 04, avec un temps d'échantillonnage  $T=0.01s$ .

La technique de clustering et d'identification des paramètres proposée est appliquée (fig. 3.19). Alors, les paramètres obtenus qui décrivent les fonctions d'appartenance sont présentées par (fig. 3.20). Ainsi, le modèle TS flou est construit avec les règles suivantes :

$$\begin{aligned} \text{Si } u_1 \text{ est } A_{11} \ \& \ u_2 \text{ est } A_{12} \ \text{Alors } y(k) &= 1.3 \cdot 10^{-1} x_1 + 7.3 \cdot 10^{-2} x_2 + 6.8 \cdot 10^{-1} \\ \text{Si } u_1 \text{ est } A_{21} \ \& \ u_2 \text{ est } A_{22} \ \text{Alors } y(k) &= 1.0 \cdot 10^1 x_1 - 1.6 \cdot 10^{-1} x_2 + 4.7 \cdot 10^{-2} \\ \text{Si } u_1 \text{ est } A_{31} \ \& \ u_2 \text{ est } A_{32} \ \text{Alors } y(k) &= 1.5 \cdot 10^1 x_1 - 8.8 \cdot 10^{-1} x_2 - 2.5 \cdot 10^{-2} \\ \text{Si } u_1 \text{ est } A_{41} \ \& \ u_2 \text{ est } A_{42} \ \text{Alors } y(k) &= 3.5 \cdot 10^{-1} x_1 + 5.0 \cdot 10^{-2} x_2 + 1.7 \cdot 10^{+1} \end{aligned} \quad (3.30)$$

La comparaison entre le modèle et le système alimenté par  $u(t) = 3\sin(1.5t)$  est montré par (fig. 3.21).

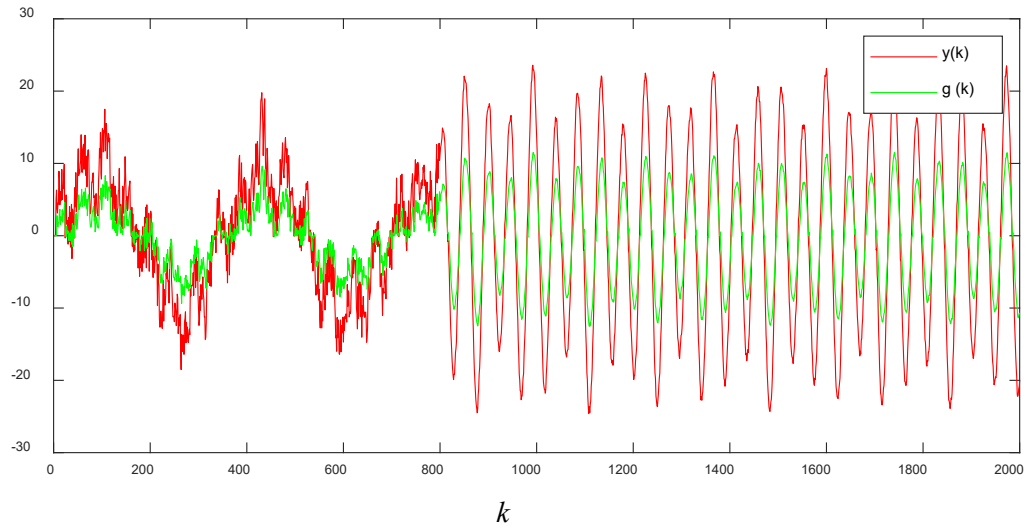


Fig. 3.18 Présentation de données utilisées pour la construction du modèle

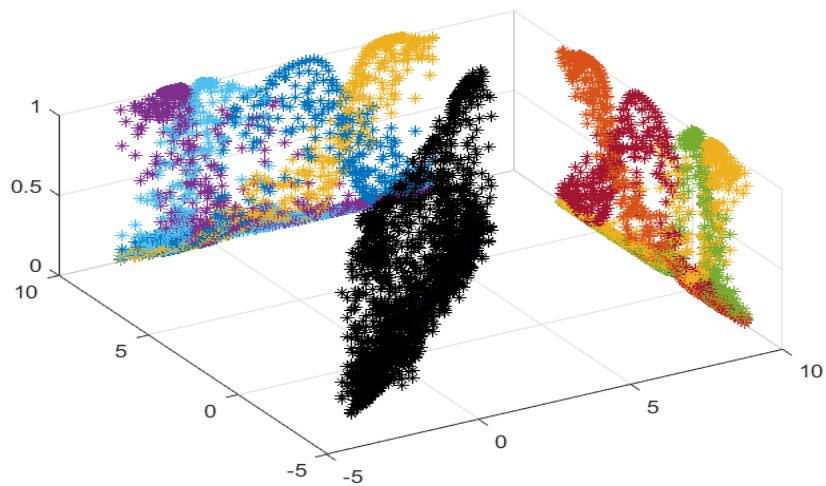


Fig. 3.19 Clusters et projections  $\mu_{ij}$  de données

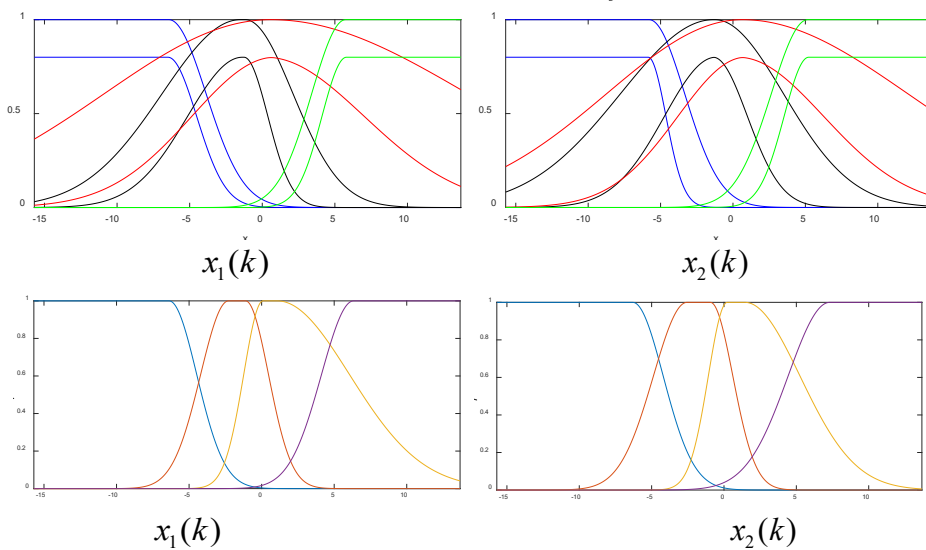


Fig. 3.20 Fonctions d'appartenance de système de type-1 et de type-2

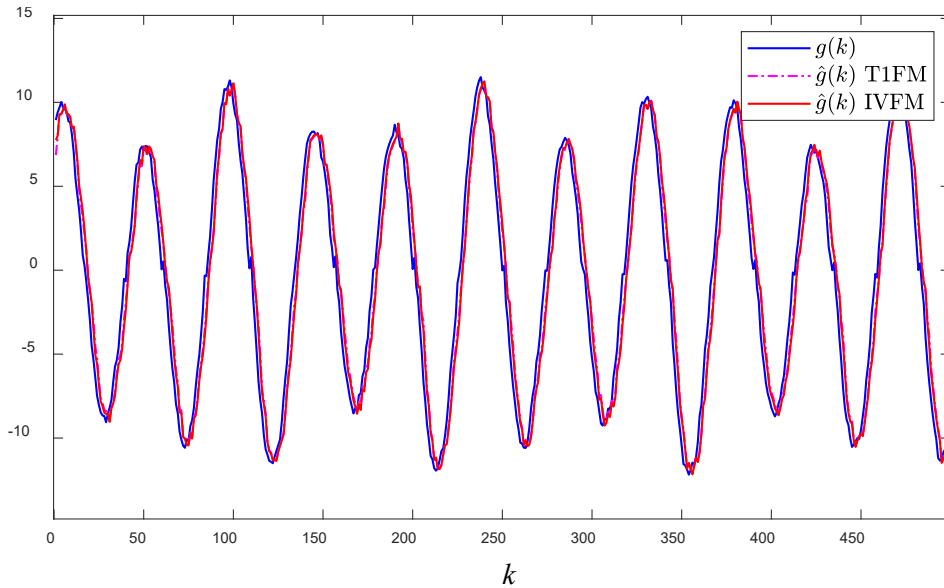


Fig. 3.21 Résultats de la sortie  $g(k)$  identifiée

Les valeurs obtenues de RMSE de modèle de type-1, noté  $RMSE_1 = 0.2349$  et RMSE de modèle de IVFM type-2 noté  $RMSE_2 = 0.1278$  montrent les capacités de l'approche proposée par rapport à l'approche classique type-1. Le modèle IVFM obtenu peut être considéré comme une bonne approximation de modèle.

### Adaptation des paramètres en boucle fermée

L'ajustement du modèle TS nous permet d'avoir une amélioration des réponses de la commande. Dans la boucle fermée, la loi d'adaptation utilisée est donnée par l'équation (3.31) :

$$\theta_g(k+1) = \theta_g(k) + \eta e(k) \phi_g(k) \quad (3.31)$$

où  $\eta = 0.1$  est le pas d'adaptation .

la performance du régulateur est testée en utilisant un signal  $r(t)$  de référence suivant:

$$\begin{cases} r(k) = 0 & \text{if } k < 200 \text{ et } r(k) = 5 & \text{if } 400 > k > 200 \\ r(k) = 5 \sin(2k\pi / 300) + n(k) & \text{if } k > 400 \end{cases}$$

L'ajustement des paramètres du modèle en fonction du temps est présenté par la figure 3.22 et la figure 3.23 montre l'efficacité de l'adaptation dans la boucle fermée.

Les résultats des commandes adaptatives de type-2 sont présentés par la figure (fig. 3.24) où le RMSE obtenu entre le système flou type 1 et le modèle de référence est d'environ 2.5339 et le

RMSE entre le système flou type 2 et le modèle de référence est d'environ 1.7105. Cela montre l'efficacité (robustesse) de la méthode adaptative de type IVFM par rapport au type-1.

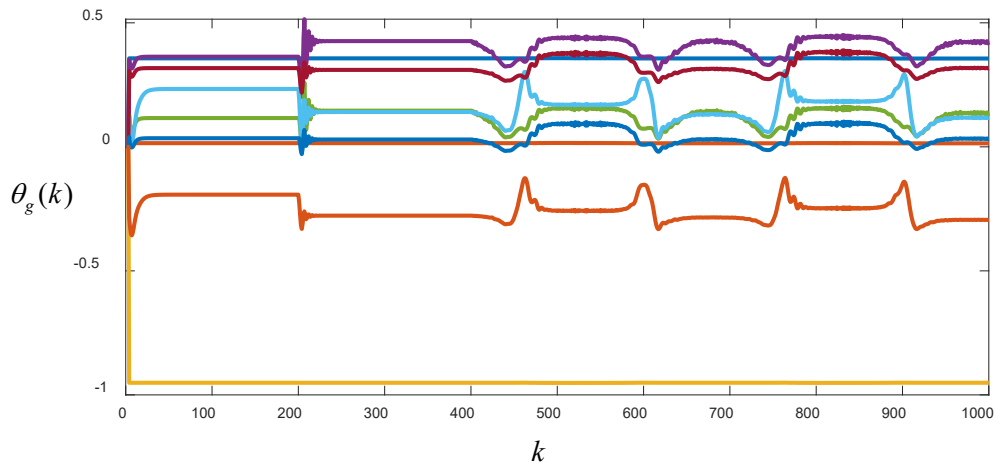


Fig. 3.22 Adaptation des paramètres de contrôleur adaptative IVFC

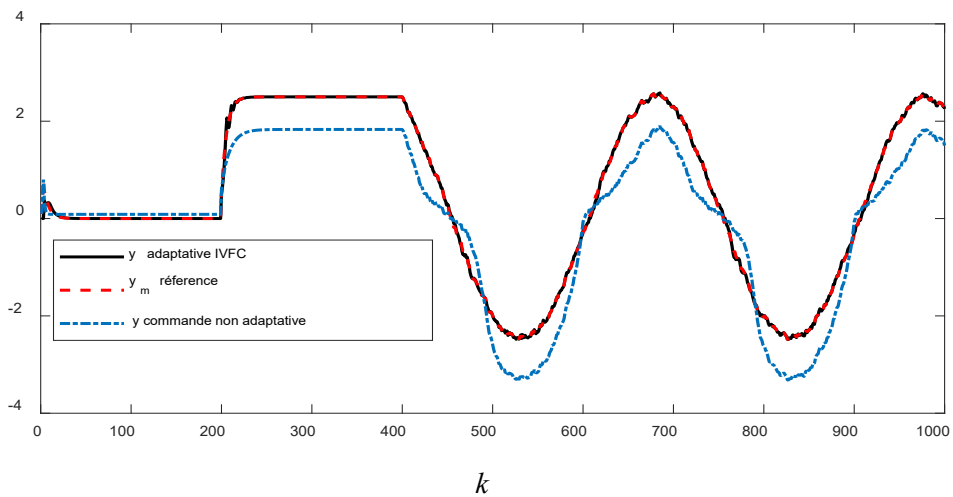


Fig. 3.23 Estimation obtenue avec et sans adaptation

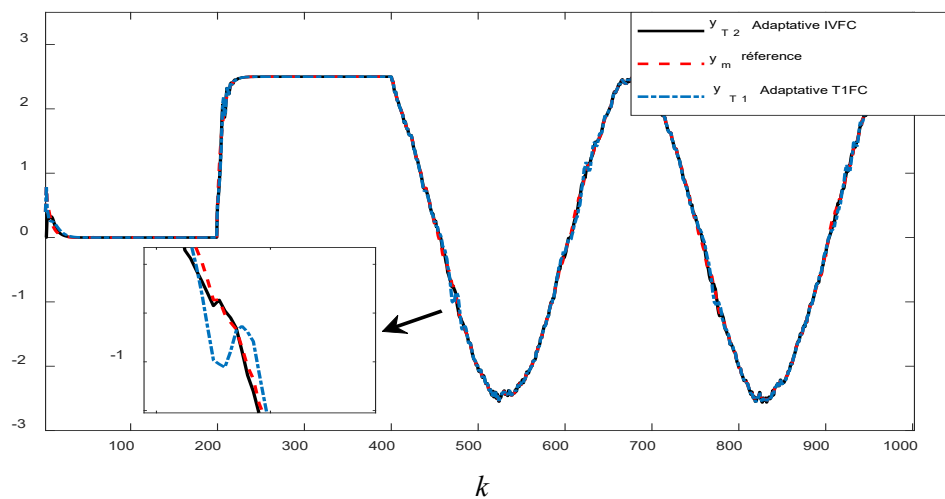


Fig. 3.24 Comparaison entre la commande adaptative type-1 et type-2

### 3.7 Conclusion

Dans ce chapitre, un nouvel algorithme de modélisation floue à valeurs d'intervalle est proposé et sa validité est vérifiée par quelques simulations. A partir des résultats obtenus, la méthode proposée a une excellente capacité à décrire une grande classe de systèmes même en présence d'incertitudes. L'objectif de cette étude était la détermination et l'estimation des fonctions d'appartenance floues à valeurs d'intervalle utilisant des techniques de clustering floues ordinaires.

L'algorithme proposé était composé de trois étapes : (1) clustering flou, (2) estimation des fonctions d'appartenance à valeurs d'intervalle type-2 (3) estimation des paramètres TS. Dans un premier temps, l'algorithme GK a été utilisé afin de détecter des clusters de formes différentes. Dans un second temps, nous avons proposé de nouveaux algorithmes EDA1 et EDA2 qui ont été utilisés pour estimer les formes des fonctions d'appartenance à intervalle inférieur et supérieur des locaux. Dans la troisième étape, le filtre de Kalman a été utilisé pour estimer les valeurs des paramètres de la partie conséquente du modèle identifié.

Les performances de la technique de modélisation proposée ont été démontrées avec des simulations sous Matlab et comparées à certaines données du monde réel (neutralisation du pH) ainsi que la commande basée sur modèle flou d'un système discret, où les résultats numériques montrent la robustesse et le pouvoir de réduction des erreurs de la méthode proposée par une comparaison des résultats des algorithmes proposés avec plusieurs travaux dans la littérature.

# Chapitre 4

## Commande Adaptative Floue d'un Quadrirotor

### 4.1. Introduction

La commande non linéaire a connu une expansion ainsi qu'une diversification importante depuis les années 50, dues à la multiplication des procédés industriels et des applications robotiques. L'étude du contrôle non linéaire est d'un grand intérêt, puisque la majorité des systèmes réels sont essentiellement non linéaires. Les méthodes linéaires conventionnelles sont satisfaisantes mais pour des plages de fonctionnement restreintes. Dès que le système sort de son domaine de fonctionnement, le contrôleur linéaire n'est plus valable et ne garantit plus la stabilité du système, d'où l'intérêt d'étudier plus profondément les méthodes de commande non linéaire.

à partir de la disponibilité d'un modèle flou, qui peut être utilisé pour la synthèse d'un contrôleur par n'importe quelle technique. Parmi celle ci les techniques les plus fréquemment utilisées la linéarisation entrée-sortie [88], la commande prédictive [89], la commande optimale [91], les techniques basées sur l'inversion du modèle [92][93] et le mode glissant [90].

Dans ce chapitre, nous considérons la possibilité qu'offrent les modèles flous de type TS pour la commande des drones en utilisant la technique adaptative du type flou. Par conséquent une commande adaptative spécifiquement par mode glissant basée sur un modèle affine TS sera développée.

### 4.2. Commande par mode glissant

La commande par Mode Glissants (CMG) a été largement utilisée dans la littérature [94][95]. Ce succès est dû à sa simplicité de mise en œuvre et à sa robustesse vis-à-vis des variations paramétriques et des perturbations externes.

Considérant d'abord une surface dite de glissement qui représente la dynamique désirée du système, puis synthétiser une loi de commande qui doit agir pour le forcer à rejoindre cette surface et d'assurer son maintien et son glissement pour atteindre l'origine du plan de phase.

On considère le système non linéaire incertain et perturbé donné par le système d'équations suivant (4.1):

$$\begin{cases} \dot{x}^{(d)} = f(t, x) - g(t, x)u(t) + m \\ y = x \end{cases} \quad (4.1)$$

La perturbation externe  $m$  est considérée inconnue mais bornée telle que  $m < W$  ( $W$  est une constante positive).

soit  $\sigma(x, t): X \times R^+ \rightarrow R$  une fonction suffisamment différentiable et considérée telle que son annulation permettra de satisfaire l'objectif de commande. La fonction  $e_s(x, t)$  est appelée variable de glissement ou de commutation.

$$S = \{x \in X \mid e_s(x, t) = 0\} \quad (4.2)$$

L'ensemble  $S$  représente alors une sous-variété de  $X$  de dimension  $(n - 1)$  appelée surface de glissement ou contrainte de commutation, avec :

$$e_s(t) = e^{(d-1)}(t) + \dots + k_1 \dot{e}(t) + k_0 e(t) \quad (4.3)$$

où  $e(t) = y_r(t) - y(t)$  représente l'erreur de suivi et  $K = [k_0, k_1, \dots, k_{d-2}, 1]$  un vecteur de paramètres de conception, ainsi,

$$L(s) = s_{d-1} + k_{d-2}s^{d-2} + \dots + k_1 s + k_0 \quad (4.4)$$

Supposons maintenant que les paramètres de conception dans  $K$  sont choisis de telle sorte que  $L(s)$  ait ses racines dans le demi-plan gauche (ouvert). Par conséquent  $\lim_{t \rightarrow \infty} e_s(t) \rightarrow 0$ .

Par exemple, pour le cas où  $d = 2$ ,  $K = [k_0, 1]$  et  $e_s(t) = \dot{e}(t) + k_0 e(t)$  et il faut avoir  $k_0 > 0$ .

$$e_s(t) = \dot{e}(t) + k_0 e(t) = 0 \quad (4.5)$$

Alors,  $\dot{e}(t) = -k_0 e(t) \Rightarrow \lim_{t \rightarrow \infty} e(t) \rightarrow 0$  et  $\lim_{t \rightarrow \infty} y(t) \rightarrow y_r(t)$

L'objectif est d'assurer la poursuite de la trajectoire de référence  $y_r(t)$  de la sortie  $y(t)$  du système, telle que  $\lim_{t \rightarrow \infty} e = \lim_{t \rightarrow \infty} (y - y_r) \rightarrow 0$ . Pour  $f(x)$  et  $g(x)$  parfaitement connues avec  $g(x) \neq 0$ , la CMG est donnée par :

$$u = u_{eq} + u_s \quad (4.6)$$

avec 
$$\begin{cases} u_{eq} = g^{-1}(t, x) [-f(t, x) + \ddot{y}_d + k\dot{e}] \\ u_s = g^{-1}(t, x) k_s \operatorname{sgn}(e_s) = \eta(x) \operatorname{sgn}(e_s) \end{cases}, \quad \operatorname{sgn}(e_s) = \begin{cases} 1 & e_s > 0 \\ -1 & e_s < 0 \end{cases}$$

où  $u_{eq}$  est la commande équivalente permettant le maintien et le glissement le long de la surface  $e_s$ ,  $u_s$  représente le signal de commutation assurant la convergence du système vers la surface. La loi de commande (4.6) est robuste vis-à-vis des perturbations paramétriques et externes mais présente quelques inconvénients majeurs tels que : le phénomène de broutement provoqué par l'utilisation du terme  $\operatorname{sgn}(e_s)$ , la convergence de l'algorithme dépend du choix des valeurs initiales ce qui rend son implémentation complexe, et la nécessité de disposer d'une connaissance même partielle de la dynamique du système [43][88]

Par conséquent, pour assurer de bonnes performances de poursuite, un algorithme d'adaptation sera développé et utilisé pour améliorer l'approximation du modèle flou type-2 initialement identifié à partir du clustering des données. Cette approche résout à la fois le problème de choix des valeurs initiales ainsi que le problème de dynamique inconnue ou dissimulée.

### 4.3. Commande adaptative floue par mode glissant

Dans cette section, pour contourner les problèmes cités, on propose une commande adaptative robuste en présence de perturbations paramétriques ou externes. Nous considérons le cas où nous disposons d'une connaissance partielle du système où la partie inconnue sera approximée par un système flou type-2.

Dans (4.1), on suppose que les fonctions dynamiques  $f(x)$  et  $g(x)$  sont inconnues. La loi de commande (4.6) ne peut être utilisée sous cette forme.

De ce fait, pour synthétiser la loi de commande proposée, la fonction inconnue  $f(x)$  est approximée par un système flou type-2 et l'objectif de la commande est de concevoir un contrôleur flou adaptatif pour le système (4.1) tels que :

- tous les signaux en boucle fermée du système soient bornés,
- la sortie du système suive le signal de référence désiré  $y_r(k)$ .

Dans ce travail on suppose que  $g(x) > 0$ . Cependant, l'analyse peut être facilement adaptée aux systèmes avec  $g(x) < 0$ .

Il existe donc une constante positive inconnue telle que  $g(x) \geq g_0 > 0$ . Il est à noter que ce paramètre est requis uniquement pour l'analyse mathématique, mais sa valeur réelle n'est pas nécessaire, ni pour la mise en œuvre, ni pour l'implémentation de la loi de commande.

Nous supposons également l'existence des dérivées de  $y_r$  jusqu'à l'ordre de deux et que  $y_r^{(i)}, i = 0, 1, 2$ , sont bornées.

### 4.3.1 Approximation floue Type-2 et initialisation du modèle flou par Clustering

Comme il a été prouvé que les systèmes TS flous type-2 peuvent approximer uniformément, n'importe quelle fonction non linéaire avec une précision arbitraire donnée sur un ensemble compact, Pour Les systèmes TS prennent la forme suivante (3.16) :

$$Y(x) = \frac{y_r + y_l}{2} = \frac{1}{2} \theta^T [\phi_l(x) + \phi_r(x)] = \theta^T \varphi(x) \quad (4.7)$$

où  $\varphi^T(x) = \frac{1}{2} [\phi_l(x) + \phi_r(x)]^T$  et

$$y_l = \underbrace{\left[ \xi^1[x_1 \dots x_i] \quad \dots \quad \xi^L[x_1 \dots x_i], \xi^{L+1}[x_1 \dots x_i] \quad \dots \quad \xi^K[x_1 \dots x_i] \right]}_{\phi_l} \underbrace{\begin{bmatrix} C^1 \\ \vdots \\ C^K \end{bmatrix}}_{\theta} = \phi_l(x)^T \theta \quad (4.8)$$

Cette structure simplifiée nous permettra d'approximer les dynamiques inconnues et d'exploiter aisément les différentes techniques de commande adaptative.

L'initialisation de la partie précédente dans l'équation (4.7) joue un rôle important dans la structure du système. Étant donné que sa dimension appropriée est à peine connue au début de la phase de conception, les algorithmes de clustering sont souvent utilisés comme

procédure d'initialisation de la partie antécédente ainsi que la base de règles des systèmes flous.

#### 4.4. Conception de la loi de commande adaptative floue par mode glissant

Les fonctions approximées  $f(x)$  et  $g(x)$  par l'approximateur universelle flous, prennent la forme suivante :

$$\begin{cases} \hat{f}(x, \theta) = \theta_f^T \varphi_f(x) \\ \hat{g}(x, \theta) = \theta_g^T \varphi_g(x) \end{cases} \quad (4.9)$$

Avec  $\varphi_f$  et  $\varphi_g$  sont des vecteurs de fonctions floues de base, supposés convenablement fixés par la technique d'identification présentée dans §3.5, et  $\theta_f$  et  $\theta_g$  sont les vecteurs de paramètres ajustables supposés définis respectivement dans le domaine  $\Omega_f$  et  $\Omega_g$ . On définit le sous-ensemble  $S_x \subseteq \mathfrak{R}^n$  comme un espace dans lequel les trajectoires des états  $x$  restent à l'intérieur en boucle fermée.

Les fonctions  $f(x)$  et  $g(x)$  sont exprimées comme suit :

$$\begin{cases} \hat{f}(x, \theta_f) = f^*(x, \theta_f^*) + w_f(x) \\ \hat{g}(x, \theta_g) = g^*(x, \theta_g^*) + w_g(x) \end{cases} \quad (4.10)$$

Avec  $w_f$  et  $w_g$  représentant les erreurs d'approximation floue  $f^*(x, \theta)$  et  $g^*(x, \theta)$  sont respectivement les paramètres optimaux de  $\hat{f}(x)$  et  $\hat{g}(x)$ .

Les valeurs des paramètres  $\theta_f$  et  $\theta_g$  minimisant respectivement les erreurs d'approximation  $w_f(x)$  et  $w_g(x)$ . Ces paramètres optimaux satisfont :

$$\begin{aligned} \theta_f^* &= \arg \min_{\theta_f \in \Omega_f} \left( \sup_{x \in S_x} \left| \theta_f^T \varphi_f(x) - f(x) \right| \right) \\ \theta_g^* &= \arg \min_{\theta_g \in \Omega_g} \left( \sup_{x \in S_x} \left| \theta_g^T \varphi_g(x) - g(x) \right| \right) \end{aligned} \quad (4.11)$$

Sachant que les paramètres optimaux  $f^*(x, \theta)$  et  $g^*(x, \theta)$  sont introduites uniquement pour faire l'étude théorique de la stabilité de la commande.

On suppose que  $|w_f(x)| \leq W_f$  et  $|w_g(x)| \leq W_g$  où  $W_f(x)$  et  $W_g(x)$  sont les bornes connues de l'erreur de l'approximation. Les deux fonctions  $w_f(x)$  et  $w_g(x)$  peuvent être choisis

arbitrairement petites. Il est important de noter que les  $w_f(x)$  et  $w_g(x)$  représentent l'amplitude de l'erreur entre les fonctions décrivant la dynamique du système et l'approximateur.

Prenant comme approximateurs de  $f(x)$  et  $g(x)$  le système d'équations suivant :

$$\begin{cases} \hat{f}(x, \theta) = \theta_f^T \phi_f \\ \hat{g}(x, \theta) = \theta_g^T \phi_g \end{cases} \quad (4.12)$$

où  $\theta_f$  et  $\theta_g$  sont les vecteurs adaptés en ligne.

Les paramètres d'erreurs sont écrits par :

$$\begin{cases} \tilde{\theta}_f(t) = \theta_f(t) - \theta_f^* \\ \tilde{\theta}_g(t) = \theta_g(t) - \theta_g^* \end{cases} \quad (4.13)$$

Considérons la loi de commande adaptative floue indirecte proposé de forme (4.4)

$$u = u_{eq} + u_s$$

Les deux termes  $u_{eq}$  et  $u_s$  vont être calculé en détails dans les sections qui suivent. Dans la suite on note IVFC (Interval value fuzzy control) et IVFM (Interval value fuzzy Model) pour la commande basée sur les systèmes flous type-2, d'autre part on note TIFC et TIFM pour les systèmes de type-1.

#### 4.4.1. Terme de contrôle d'équivalence $u_{eq}$

Pour pouvoir calculer le terme  $u_{eq}$  et pour plus de commodité, prenons  $\bar{e}_s(t) = k_{d-2}e^{(d-1)}(t) + \dots + k_0\dot{e}(t)$  de sorte que  $\bar{e}_s(t) = \dot{e}_s(t) - e^{(d)}(t)$

De ce fait, le terme de contrôle **d'équivalence** devient comme :

$$u_{eq} = -\frac{1}{\hat{g}(x)}(\hat{f}(x) + v(t)) \quad (4.14)$$

où

$$\begin{aligned} v(t) &= y^{(d)} + \gamma e_s(t) + \dot{e}_s(t) \\ &= y_m^{(d)} + \gamma e_s + \bar{e}_s \end{aligned}$$

Avec  $\gamma > 0$  un paramètre de conception.

Nous utiliserons la projection pour nous assurer que ( $\hat{g} > 0$ ) et afin que  $u_{ce}$  soit bien définie.

La d<sup>ème</sup> dérivé de l'erreur de sortie sera  $e^{(d)} = y_m^{(d)} - y^{(d)}$  donc :

$$e^{(d)} = y_m^{(d)} - f(x) - g(x)u(t) \quad (4.15)$$

Comme  $u = u_{eq} + u_s$  alors :

$$e^{(d)} = y_m^{(d)} - f(x) - g(x)u_{eq}(t) - g(x)u_s(t)$$

$$e^{(d)} = y_m^{(d)} - f(x) - \frac{g(x)}{\hat{g}(x)}(-\hat{f}(x) + v(t)) - (g(x))u_s \quad (4.16)$$

Notons que les deux premiers termes de l'équation (4.16) :

$$y_m^{(n)} - f(x) = y_m^{(n)} - \hat{f}(x) - f(x) + \hat{f}(x) = (-\hat{f} + v) - f + \hat{f} - \gamma e_s - \bar{e}_s \quad (4.17)$$

En substituant (4.17) à l'équation (4.16), on trouve

$$e^{(n)} = \left(1 - \frac{g(x)}{\hat{g}(x)}\right) \left(-\hat{f}(x) + v(t)\right) - f(x) + \hat{f}(x) - \gamma e_s - \bar{e}_s - g(x)u_s$$

$$= (\hat{f}(x) - f(x)) + (\hat{g}(x) - g(x))u_{eq} - \gamma e_s - \bar{e}_s - g(x)u_s$$

et comme  $\bar{e}_s = \dot{e}_s - e^{(n)}$  on aura

$$\dot{e}_s + \gamma e_s = (\hat{f}(x) - f(x)) + (\hat{g}(x) - g(x))u_{eq} - (g(x))u_s \quad (4.18)$$

#### 4.4.2. Lois d'adaptation des paramètres

Considérons la fonction candidate de Lyapunov suivante :

$$V_i = \frac{1}{2}e_s^2 + \frac{1}{2\eta_f}\tilde{\theta}_f^T\tilde{\theta}_f + \frac{1}{2\eta_g}\tilde{\theta}_g^T\tilde{\theta}_g \quad (4.19)$$

où  $\eta_f > 0$  and  $\eta_g > 0$  sont les paramètres de conception (adaptation).

La fonction de Lyapunov (4.19) quantifie l'erreur de poursuite et l'erreur d'estimation des paramètres, sa dérivée donne :

$$\dot{V}_i = e_s\dot{e}_s + \frac{1}{\eta_f}\tilde{\theta}_f^T\dot{\tilde{\theta}}_f + \frac{1}{\eta_g}\tilde{\theta}_g^T\dot{\tilde{\theta}}_g \quad (4.20)$$

Remplaçant la dérivée de l'erreur  $\dot{e}_s$  de (4.18) dans l'équation (4.20) on aura :

$$\dot{V}_i = e_s \left( -\gamma e_s + (\hat{f}(x) - f(x)) + (\hat{g}(x) - g(x))u_{ce} - (g(x))u_{si} \right) + \frac{1}{\eta_f}\tilde{\theta}_f^T\dot{\tilde{\theta}}_f + \frac{1}{\eta_g}\tilde{\theta}_g^T\dot{\tilde{\theta}}_g$$

Ainsi,

$$\hat{f}(x) - f(x) = \theta_f^T\phi_f(x) - \theta_f^{*T}\phi_f(x) - w_f(x) = \tilde{\theta}_f^T\phi_f(x) - w_f(x) \quad (4.21)$$

La même chose pour  $\hat{g}(x) - g(x)$ , donc,

$$\dot{V}_i = -\gamma e_s^2 + \left( \tilde{\theta}_f^T \phi_f(x) - w_f(x) + \tilde{\theta}_g^T \phi_g(x) u_{eq} - w_g(x) u_{ce} - g(x) u_s \right) e_s + \frac{1}{\eta_f} \tilde{\theta}_f^T \dot{\tilde{\theta}}_f + \frac{1}{\eta_g} \tilde{\theta}_g^T \dot{\tilde{\theta}}_g$$

On prend les lois d'adaptation suivantes :

$$\dot{\tilde{\theta}}_f(t) = -\eta_f \varphi_f(x) e_s \quad (4.22)$$

$$\dot{\tilde{\theta}}_g(t) = -\eta_g \varphi_g(x) e_s u_{eq} \quad (4.23)$$

$$\text{où } \varphi_f(x) = \frac{1}{2} (\xi_l^f + \xi_r^f) \text{ et } \varphi_g(x) = \frac{1}{2} (\xi_l^g + \xi_r^g)$$

La rapidité d'adaptation dépend du choix des valeurs des gains d'adaptation  $\eta_f$  et  $\eta_g$

On suppose que les paramètres idéals sont constants  $\dot{\tilde{\theta}}_f = \dot{\theta}_f$  et  $\dot{\tilde{\theta}}_g = \dot{\theta}_g$ . Maintenant avec

cette condition on remarque que :  $\frac{1}{\eta_g} \tilde{\theta}_g^T \dot{\tilde{\theta}}_g = -\tilde{\theta}_g^T \varphi_g(x) e_s u_{eq}$  et  $\frac{1}{\eta_f} \tilde{\theta}_f^T \dot{\tilde{\theta}}_f = -\tilde{\theta}_f^T \varphi_f(x) e_s$

alors,

$$\begin{aligned} \dot{V}_i &= -\gamma e_s^2 + \left( \tilde{\theta}_f^T \phi_f(x) - w_f + \tilde{\theta}_g^T \phi_g(x) u_{eq} - w_g u_{eq} - g u_s \right) e_s - \tilde{\theta}_f^T \varphi_f(x) e_s - \tilde{\theta}_g^T \varphi_g(x) e_s u_{eq} \\ &= -\gamma e_s^2 + \left( -w_f(x) - w_g(x) u_{eq} - g(x) u_s \right) e_s \end{aligned} \quad (4.24)$$

#### 4.4.3. Modification des lois de commande par projection

Les lois d'adaptation dans les équations (4.22) et (4.23) ne garantissent pas  $\theta_f \in \Omega_f$  et  $\theta_g \in \Omega_g$

il faut utiliser, la projection pour cela (e.g. pour assurer que  $\hat{g}(x) \geq g_0$ ). L'interprétation

géométrique de cette projection est présentée dans la fig 4.1. Supposons en particulier que la

i<sup>ème</sup> composante de  $\theta_f^*$  et  $\theta_g^*$  est dans un intervalle connu :  $\theta_{f_i}^* \in [\theta_{f_i}^{\min}, \theta_{f_i}^{\max}]$  et  $\theta_{g_i}^* \in [\theta_{g_i}^{\min}, \theta_{g_i}^{\max}]$

et on veut que  $\theta_f \in [\theta_f^{\min}, \theta_f^{\max}]$  et  $\theta_g \in [\theta_g^{\min}, \theta_g^{\max}]$

Défini  $\theta_f^m \in [\theta_f^{\min}, \theta_f^{\max}]$  comme un point dans la plage acceptable.

Nous pouvons résumer la mise en œuvre en utilisant la règle suivante :

$$\begin{cases} \text{si } \theta_f(t) \notin (\theta_f^{\min}, \theta_f^{\max}) \text{ et } \theta_f^{ud}(\theta_f - \theta_f^m) \geq 0 \text{ alors } \dot{\theta}_f(t) = 0 \\ \text{sinon } \dot{\theta}_f(t) = -\eta_f \varphi_f(x) e_s u_{eq} \end{cases}$$

Dans l'analyse de stabilité, puisque  $\theta_f^*$  et  $\theta_g^*$  sont dans cette projection des lois d'adaptation

sera toujours entraînée par une erreur d'estimation des paramètres  $w_f(x)$  et  $w_g(x)$  qui décroît

la fonction  $V_i$  au moins autant que si la projection n'a pas été utilisée ; par conséquent, le terme

droit de l'équation (4.24) dépassera le  $\dot{V}_i$  qui en résulterait si la projection est utilisée. Pour cette raison, nous concluons que :

$$\dot{V}_i \leq -\gamma e_s^2 + (-w_f(x) - w_g(x)u_{ce} - g(x)u_s)e_s \quad (4.25)$$

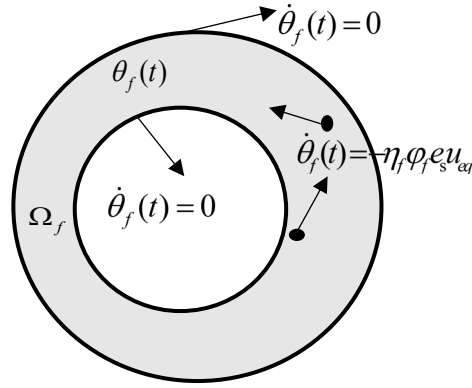


Fig. 4.1 Illustration des projections des lois d'adaptation

#### 4.4.4. Terme de contrôle du mode glissant $u_s$

Pour assurer que l'inégalité (4.25) est inférieure ou égale à zéro, on choisit :

$$u_s = \eta(x) \operatorname{sgn}(e_s) = \frac{(W_f(x) + W_g(x)|u_{eq}|)}{g_0} \operatorname{sgn}(e_s) \quad (4.26)$$

On voit que :  $-(w_f(x) + w_g(x)u_{ce})e_s \leq (|w_f(x)| + |w_g(x)u_{ce}|)|e_s|$  donc :

$$\dot{V}_s \leq -\gamma e_s^2 + (|w_f(x)| + |w_g(x)u_{eq}|)|e_s| - e_s \frac{g(x)}{g_0} (W_f(x) + W_g(x)|u_{eq}|) \operatorname{sgn}(e_s) \quad (4.27)$$

avec  $\frac{g(x)}{g_0} \geq 1$  on aura :

$$\dot{V}_i = -\gamma e_s^2 + |w_f(x)||e_s| + |w_g(x)u_{eq}||e_s| - e_s \operatorname{sgn}(e_s) W_f(x) - e_s \operatorname{sgn}(e_s) W_g(x)|u_{eq}|$$

Où  $|e_s| = e_s \operatorname{sgn}(e_s) \quad \forall e_s \neq 0 : |w_f(x)| \leq W_f(x)$  et  $|w_g(x)| \leq W_g(x)$  alors :

$$|w_f(x)||e_s| - e_s \operatorname{sgn}(e_s) W_f(x) = |e_s| (|w_f(x)| - W_f(x)) \leq 0$$

$$|w_g(x)u_{eq}||e_s| - e_s \operatorname{sgn}(e_s) W_g(x)|u_{eq}| = |e_s| (|w_g(x)u_{eq}| - W_g(x)|u_{eq}|) \leq 0$$

$$\dot{V}_i \leq -\gamma e_s^2 \quad (4.28)$$

Puisque  $\gamma e_s^2 \geq 0$ , cela montre que  $\dot{V}_i$  est une fonction non croissante dans le temps qui représente l'erreur de suivi et l'erreur d'estimation des paramètres, Noter que  $\gamma > 0$  a une influence sur la rapidité de convergence de  $V_i \rightarrow 0$ , et lorsque  $\gamma$  est plus grand, l'erreur de suivi converge plus rapidement.

#### 4.4.5. Stabilité asymptotique de l'erreur de suivi et borne des signaux

A partir des hypothèses considérées (les signaux de référence sont bornés,  $x$  est mesurable,  $d = n$ , et que la projection garantit que le terme  $u_{eq}$  est bien défini), ce qui suit est vrai :

1. la sortie de système est telle que  $y, \dot{y}, \dots, y^{(d-1)}$  soit bornée.
2. les signaux de commande  $u, u_{eq}$  et  $u_s$  sont bornés.
3. les paramètres  $\theta_f(t)$  et  $\theta_g(t)$  sont bornés.
4. Le suivi est asymptotique ( $\lim_{t \rightarrow \infty} e(t) = 0$ )

Pour prouver ça, premièrement notez que  $V_i$  une fonction positive et  $\dot{V}_i \leq -\gamma e_s^2$ .

On sait que  $e_s, \theta_f$  et  $\theta_g$  sont bornées. Puisque  $e_s$  est bornée et  $y_r$  et ses dérivées sont bornées et on sait que  $y, \dot{y}, \dots, y^{(d-1)}$  sont bornées. Donc,  $x$  est bornés.

Par conséquent,  $f(x), \hat{f}(x), g(x)$  et  $\hat{g}(x)$  sont bornées. Puisque  $x$  est borné et  $\hat{g}(x) \geq g_0$ , alors  $u_{eq}$  et  $u_s$  donc  $u$ , sont bornés. Ensuite, notez que :

$$\int_0^{\infty} \gamma e_s^2 dt \leq -\int V_i dt = V_i(0) - V_i(\infty) \quad (4.29)$$

Ceci établit que  $e_s \in L_2$  ou  $L_2 = \left\{ z(t) : \int_0^{\infty} z^2(t) dt < \infty \right\}$  puisque  $V_i(0)$  et  $V_i(\infty)$  sont bornés.

Notez que d'après l'équation (4.23),  $\dot{e}_s$  est borné. Par conséquent, puisque  $e_s$  et  $\dot{e}_s$  sont bornés et  $e_s \in L_2$ , et d'après le lemme de Barbalat [104] nous avons que  $\lim_{t \rightarrow \infty} e_s(t) = 0$ . Il devrait être clair alors, via la définition de  $e_s(t)$ , que  $\lim_{t \rightarrow \infty} e(t) = 0$

Il est possible de réduire les signaux de haute fréquence qui peuvent résulter du terme de commande en mode glissant, en utilisant une transition lisse de la fonction  $\text{sgn}(e_s)$  [94]. Dans ce cas, nous n'obtenons que la convergence vers un voisinage de  $e_s = 0$ . Pour monter la stabilité, on définit l'erreur  $e_e$  :

$$e_\varepsilon = e_s - \varepsilon \text{sat}(e_s / \varepsilon) \text{ ou } \text{sat}(x) = \begin{cases} 1 & \text{si } 1 \leq x \\ x & \text{si } -1 < x < 1 \\ -1 & \text{si } x \leq -1 \end{cases} \quad (4.30)$$

où  $e_\varepsilon$  représente la distance entre  $e_s$  et la bande, alors  $e_\varepsilon = 0$  lorsque  $e_s$  est à l'intérieur de la bande autour de la surface de glissement. La commande d'équivalence est définie par :

$$u_{eq} = -\frac{1}{\hat{g}(x)}(\hat{f}(x) + v_\varepsilon(t)) \text{ avec } v_\varepsilon(t) = y^{(n)}_d + \gamma e_\varepsilon + \bar{e}_s$$

On considère la fonction de Lyapunov suivante :

$$V_i = \frac{1}{2}e_\varepsilon^2 + \frac{1}{2\eta_f}\tilde{\theta}_f^T\tilde{\theta}_f + \frac{1}{2\eta_g}\tilde{\theta}_g^T\tilde{\theta}_g$$

Les lois d'adaptation deviennent alors :

$$\dot{\tilde{\theta}}_f(t) = -\eta_f \varphi_f(x) e_\varepsilon \quad (4.31)$$

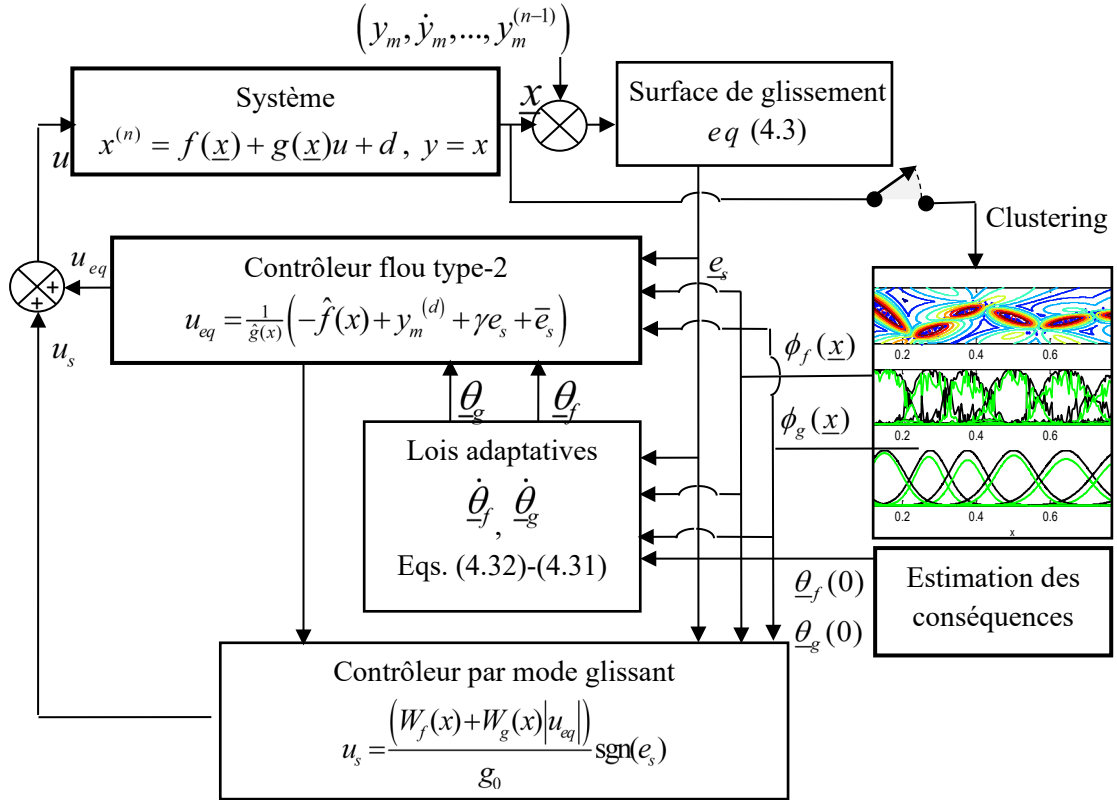
$$\dot{\tilde{\theta}}_g(t) = -\eta_g \varphi_g(x) e_\varepsilon u_{eq} \quad (4.32)$$

Avec ces lois, la dérivée de Lyapunov s'exprime comme :

$$\dot{V}_i \leq -\gamma e_\varepsilon^2 + (|w_f(x)| + |w_g(x)u_{ce}|)|e_\varepsilon| - g(x)u_s e_\varepsilon$$

Et le terme  $u_s$  devient une commande à une action lisse.

$$u_s = \frac{(W_f(x) + W_g(x)|u_{eq}|)}{g_0} \text{sat}(e_s / \varepsilon) \quad (4.33)$$



**Fig. 4.2** Commande floue indirecte proposée, phase de clustering à  $t < t_0$  et la phase de commande à  $t \geq t_0$

Avec  $e_\varepsilon \text{ sat}(e_s / \varepsilon) = |e_s|$ , on peut montrer que  $\dot{V}_i \leq -\gamma e_\varepsilon^2$ , s'implique que  $|e_s|$  est stable asymptotiquement, ainsi que  $e_s$  converge vers le  $\varepsilon$ -voisin de  $e_s = 0$ , et  $e$  vers un voisin de  $e = 0$ .

#### 4.5. Etude de cas : Application au quadrirotor

Dans cette étude de cas, nous appliquons les méthodes d'identification et la commande pour le quadrirotor en appliquant respectivement le clustering flou et la CMG adaptative. Une synthèse de la loi de commande la mieux adaptée au système afin de réguler l'attitude du système est utilisée.

Considérant le modèle du système (1.46) qui s'écrira comme une représentation d'états où  $\underline{x} = (\varphi, \dot{\varphi}, \theta, \dot{\theta}, \psi, \dot{\psi}, x, \dot{x}, y, \dot{y}, z, \dot{z})$  est le vecteur d'état et  $U = [U_1 U_2 U_3 U_4]$  est le vecteur de commande. La matrice de transformation entre les dérivés des angles d'orientation  $(\dot{\varphi}, \dot{\theta}, \dot{\psi})$  et la vitesse angulaire  $(p, q, r)$  est considérée comme une matrice unitaire si les perturbations sont petites  $((\dot{\varphi}, \dot{\theta}, \dot{\psi}) \approx (p, q, r)) [1]$ .

$$\dot{\underline{x}} = f(\underline{x}, U) = \begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \\ \dot{x}_5 \\ \dot{x}_6 \\ \dot{x}_7 \\ \dot{x}_8 \\ \dot{x}_9 \\ \dot{x}_{10} \\ \dot{x}_{11} \\ \dot{x}_{12} \end{pmatrix} = \begin{pmatrix} \dot{\phi} \\ \dot{\theta}\psi a_1 + \dot{\theta}a_2\Omega_r + b_1U_2 \\ \dot{\theta} \\ \dot{\phi}\psi a_3 + \dot{\phi}a_4\Omega_r + b_2U_3 \\ \dot{\psi} \\ \dot{\theta}\dot{\phi}a_5 + b_3U_4 \\ \dot{z} \\ g - (\cos\phi \cos\theta)\frac{1}{m}U_1 \\ \dot{x} \\ u_x \frac{1}{m}U_1 \\ \dot{y} \\ u_y \frac{1}{m}U_1 \end{pmatrix} = \begin{pmatrix} \dot{\phi} \\ f_1 + g_1U_2 \\ \dot{\theta} \\ f_2 + g_2U_3 \\ \dot{\psi} \\ f_3 + g_3U_4 \\ \dot{z} \\ f_4 - (\cos\phi \cos\theta)g_4U_1 \\ \dot{x} \\ u_x g_5U_1 \\ \dot{y} \\ u_y g_6U_1 \end{pmatrix} \quad (4.34)$$

$$\text{Où : } \begin{cases} a_1 = (I_{yy} - I_{zz}) / I_{xx} \\ a_2 = J_r / I_{xx} \\ a_3 = (I_{zz} - I_{xx}) / I_{yy} \\ a_4 = J_r / I_{yy} \\ a_5 = (I_{xx} - I_{yy}) / I_{zz} \end{cases} \left| \begin{cases} b_1 = l / I_{xx} \\ b_2 = l / I_{yy} \\ b_3 = 1 / I_{zz} \\ u_x = (c\phi s\theta c\psi - s\phi s\psi) \\ u_y = (c\phi s\theta s\psi - s\phi c\psi) \end{cases} \right.$$

Les paramètres de drone sont :

$$\begin{aligned} g &= 9.81 m.s^{-2}; & m &= 1.2 \text{ kg}; \\ I_{xx} &= 0.18 \text{ kg.m}^2; & d &= 1.1e-6 \text{ N.m.s}^2 \\ I_{yy} &= 0.18 \text{ kg.m}^2; & b &= 54.2e-6 \text{ N.m}^2; \\ I_{zz} &= 0.09 \text{ kg.m}^2; & J_r &= 1.32e-3 \text{ kg.m}^2; \\ l &= 0.23 \text{ m}; \end{aligned}$$

#### 4.5.1 Identification du système hors ligne

Le système quadrirotor (4.30) est décomposé en trois sous-systèmes :

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \begin{pmatrix} \dot{\phi} \\ f_1(\dot{\theta}, \dot{\psi}) + g_1U_2 \end{pmatrix} \quad \begin{pmatrix} \dot{x}_3 \\ \dot{x}_4 \end{pmatrix} = \begin{pmatrix} \dot{\psi} \\ f_2 + g_2U_3 \end{pmatrix} \quad \begin{pmatrix} \dot{x}_5 \\ \dot{x}_6 \end{pmatrix} = \begin{pmatrix} \dot{\theta} \\ f_3 + g_3U_3 \end{pmatrix}$$

Où  $f_1(\dot{\theta}, \dot{\psi}, \Omega_r) = \dot{\theta}\psi a_1 + \dot{\theta}a_2\Omega_r$ ,  $f_2(\dot{\theta}, \dot{\psi}, \Omega_r) = \dot{\theta}\psi a_3 + \dot{\theta}a_4\Omega_r$ ,  $f_3(\dot{\theta}, \dot{\psi}, \Omega_r) = \dot{\theta}\psi a_5$  et  $g_1 = b_1$ ,  $g_2 = b_2$ ,  $g_3 = b_3$ .

Les fonctions inconnues sont respectivement estimées comme suit : La première étape est destinée à estimer les fonctions  $f_i(\theta, \psi, \Omega_r)$  et  $g_i$  où  $\{i=1,2,3\}$  par clustering et  $\hat{f}_i$  s'écrit comme une approximation TS floue de type IVFM (4.33) :

$$\begin{aligned} f_1(x) &= \hat{f}_1(x_8, x_{12}, \Omega_r, \theta_{f_1}) + w_{f_1} = \theta_{f_1}^T \varphi_{f_1}(x) \\ f_2(x) &= \hat{f}_2(x_{10}, x_{12}, \Omega_r, \theta_{f_2}) + w_{f_2} = \theta_{f_2}^T \varphi_{f_2}(x) \\ f_3(x) &= \hat{f}_3(x_{10}, x_{12}, \theta_{f_3}) + w_{f_3} = \theta_{f_3}^T \varphi_{f_3}(x) \end{aligned} \quad (4.35)$$

où  $\theta_{f_i} = [w_i^T, \underline{w}_r^T]^T$  et  $\underline{\psi}_{f_i}(x, \dot{x}) = \frac{1}{2} [\underline{\xi}_i^T(x, \dot{x}), \underline{\xi}_r^T(x, \dot{x})]^T$ . et  $w_{f_1}, w_{f_2}, w_{f_3}$  sont les erreurs d'approximation

Dans cette étape nous avons généré deux jeux de données pour l'identification par clustering et la validation respectivement afin de tester la performance de prédiction du modèle TS flou obtenu.

Nous commençons par une séquence d'acquisition des données de 4000 échantillons (2000 pour l'identification et 2000 pour la validation). Le temps d'échantillonnage est de  $T=0.01s$  avec un bruit blanc ajouté à chaque sortie de 5% d'amplitude. Les nuages obtenus sont traités par la méthode EDA pour la construction des fonctions d'appartenances de type-2 (Fig. 4.3).

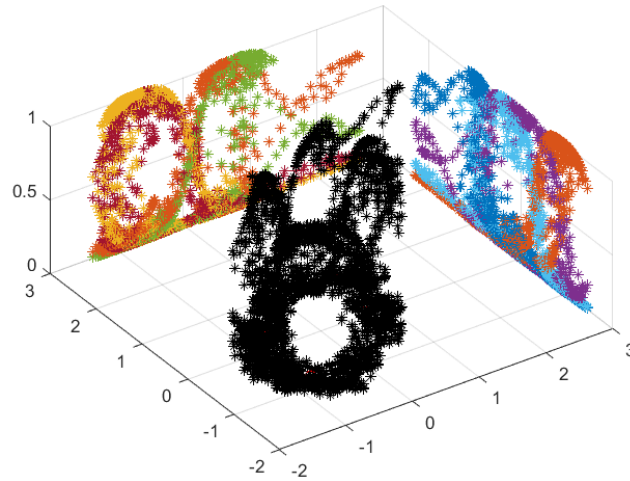


Fig. 4.3 Projection et construction des fonctions type-1 et type-2

Les figures (fig. 4.4, 4.5 et 4.6) représentent les fonctions d'appartenance de type-1 et 2 obtenues à partir des données de simulation des fonctions  $f_1(x, \Omega)$   $f_2(x, \Omega)$  et  $f_3(x, \Omega)$  respectivement.

La validation des modèles flous obtenus (le modèle flou de type-2 ; IFVS et le modèle flou de type-1 ; T1FS) comme indiquent les figures (fig.4.7- 4.9) et la table 4.1 montre que la

technique floue de type-2 proposée ( IVFM) est plus proche aux fonctions réelles du système par rapport au type-1.

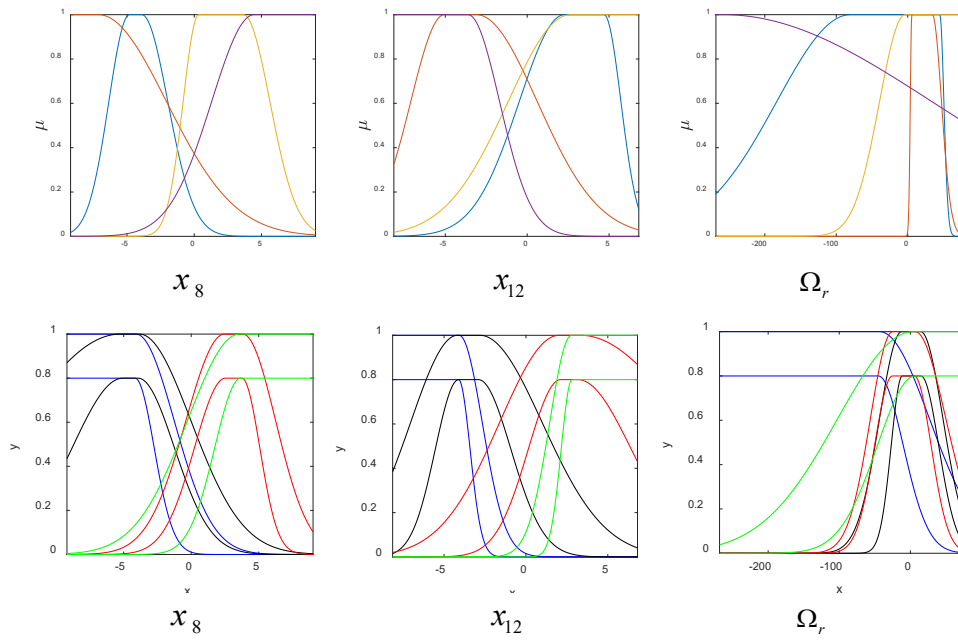


Fig. 4.4 Fonctions d'appartenance de type-2/type-1 correspondant au  $\hat{f}_1(\underline{x})$

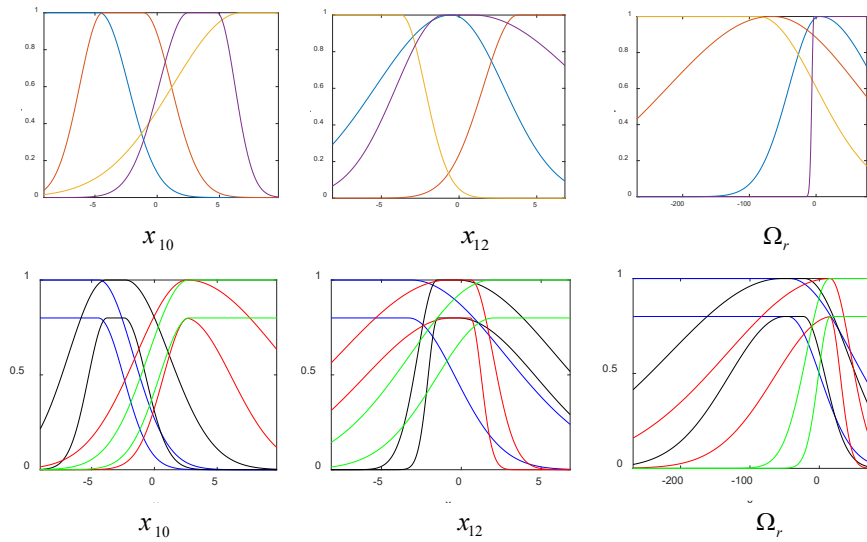
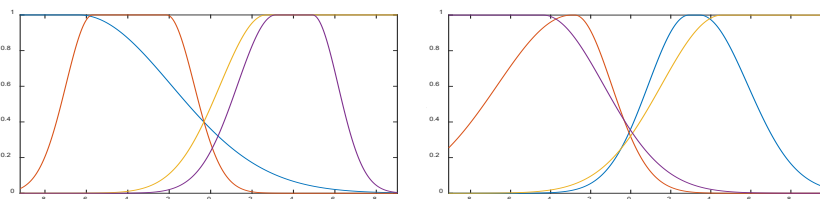


Fig. 4.5 Fonctions d'appartenance de type-2/type-1 correspondant au  $\hat{f}_2(\underline{x})$



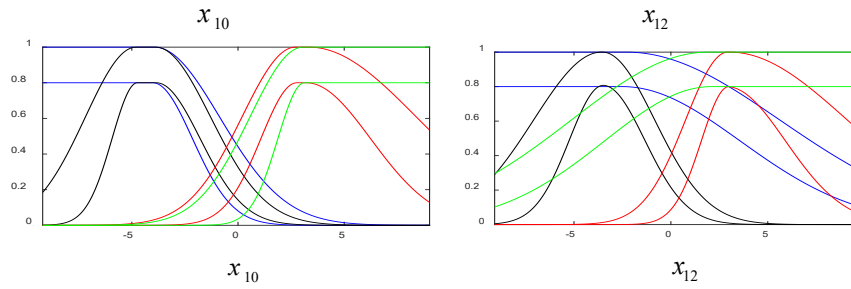


Fig. 4.6 Fonctions d'appartenance de type-2/type-1 correspondant au  $\hat{f}_3(\underline{x})$

Fonction	Fonction approximé	RMSE T1FM	RMSE IVFM
$f_1(x_8, x_{12}, \Omega)$	$\hat{f}_1(x_8, x_{12}, \theta_{f_1})$	1.5958	1.4016
$f_2(x_{10}, x_{12}, \Omega)$	$\hat{f}_2(x_{10}, x_{12}, \theta_{f_2})$	1.1717	1.1009
$f_3(x_{10}, x_{12}, \Omega)$	$\hat{f}_3(x_{10}, x_{12}, \theta_{f_3})$	0.2072	0.1930

Table. 4.1 Comparaison entre RMSE de T1FC et IVFM modèle

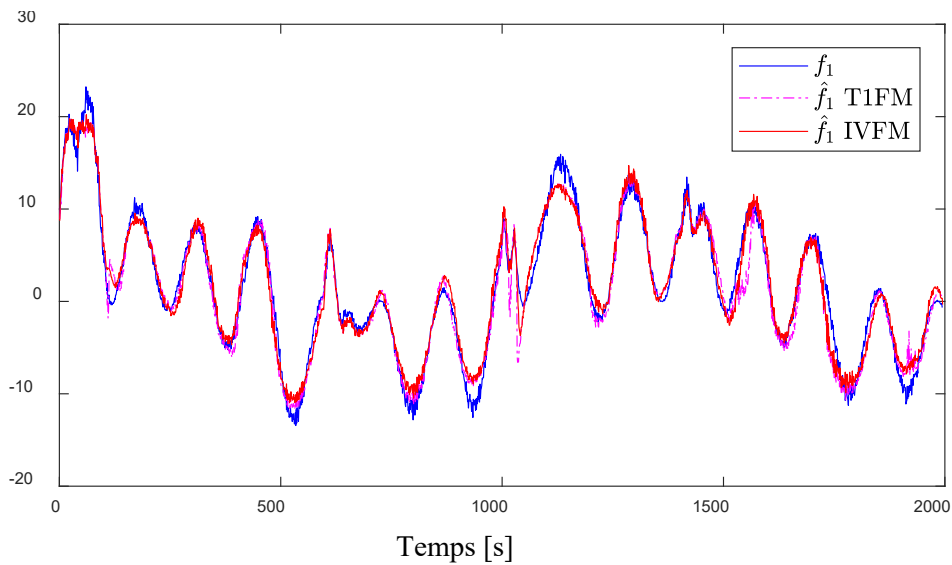


Fig. 4.7 comparaison entre  $\hat{f}_1(\underline{x})$  et  $f_1(\underline{x})$

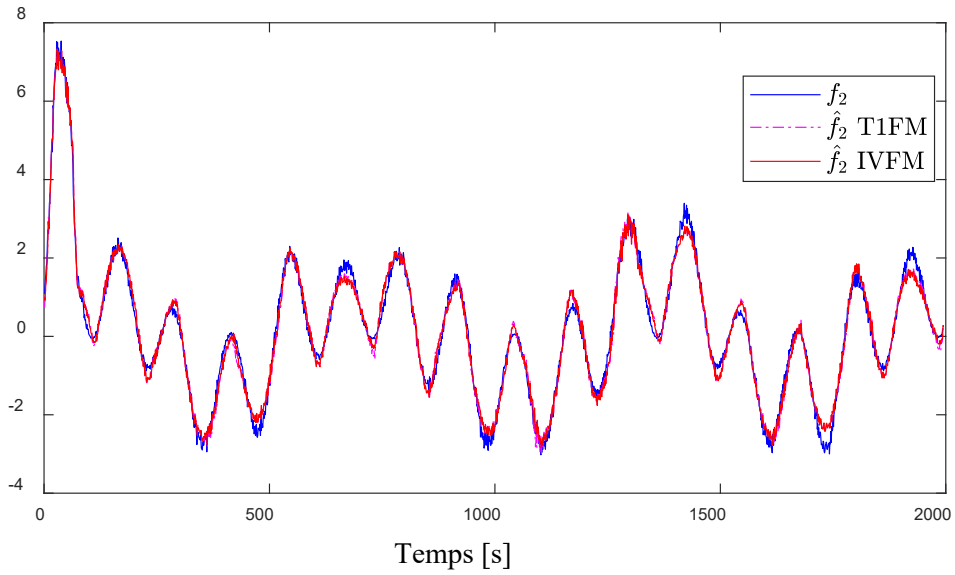


Fig. 4.8 comparaison entre  $\hat{f}_2(x)$  et  $f_2(x)$

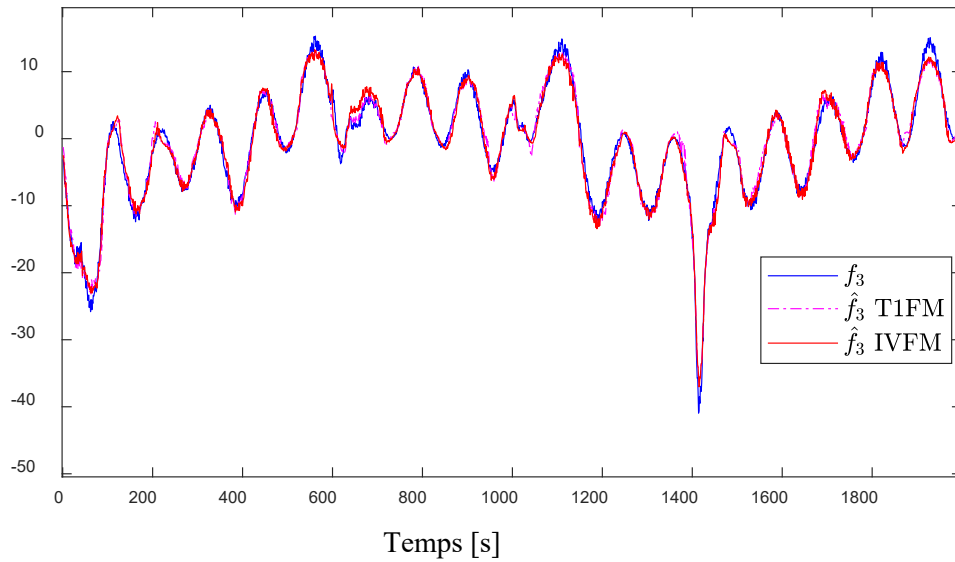


Fig. 4.9 comparaison entre  $\hat{f}_3(x)$  et  $f_3(x)$

#### 4.5.2 Commande proposée

Après l'obtention du modèle flou TS, on applique la technique de la commande CMG adaptative, en utilisant le modèle flou comme modèle initial et l'adaptation des paramètres sont fait par l'ajustement des paramètres de conséquences du modèle flou TS initial.

Soient l'erreur de suivi  $\underline{e}(t) = \underline{x}_m(t) - \underline{x}(t)$ ,  $K = [k_0, 1]$  un vecteur de paramètres de conception et l'équation de surface de glissement donnée par :

$$\underline{e}_s(t) = \underline{e} + k_0 \dot{\underline{e}} \quad (4.36)$$

Où  $\mathbf{e}_s(t) = (e_{s_x}, e_{s_y}, e_{s_z}, e_{s_\theta}, e_{s_\phi}, e_{s_\psi})$  représente la surface de glissement, dans le but de minimiser le chattering come on a deja vu (§ 4.4.5) on troduire  $\mathbf{e}_\varepsilon = (e_{\varepsilon_x}, e_{\varepsilon_y}, e_{\varepsilon_z}, e_{\varepsilon_\theta}, e_{\varepsilon_\phi}, e_{\varepsilon_\psi})$  où  $e_\varepsilon = e_s - \varepsilon \text{sat}(e_s / \varepsilon)$ .

**Proposition 1:** Pour la cconception du contrôleur d'attitude décrit par les sous-systèmes suivants:

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \begin{pmatrix} \dot{\phi} \\ f_1(\theta, \psi) + g_1 U_2 \end{pmatrix} \quad \begin{pmatrix} \dot{x}_3 \\ \dot{x}_4 \end{pmatrix} = \begin{pmatrix} \dot{\psi} \\ f_2 + g_2 U_4 \end{pmatrix} \quad \begin{pmatrix} \dot{x}_5 \\ \dot{x}_6 \end{pmatrix} = \begin{pmatrix} \dot{\theta} \\ f_3 + g_3 U_3 \end{pmatrix} \quad (4.37)$$

On propose la CMG de chaque un sous-systèmes (4.37) de la forme suivante:

$$U_i = u_{eq_i} + u_{s_i} \quad (4.38)$$

Où  $i = \{1, 2, 3\}$  represent le sous systeme de  $\phi, \psi$  et  $\theta$  respectivement, et :

$$u_{s_i} = \frac{(W_{f_i}(x) + W_{g_i}(x) |u_{eq_i}|)}{g_{0i}} \text{sat}(e_{s_i} / \varepsilon) \text{ et } u_{eq_i} = -\frac{1}{\hat{g}_i(x)} \left( \hat{f}_i(x, \theta_{f_i}) + y_{m_i}^{(2)} + \gamma e_{\varepsilon_i} + \bar{e}_{s_i} \right)$$

Ou  $\bar{e}_s = k_i e$ ,  $y_{d_i} = \{\phi_d, \varphi_d, \theta_d\}$ ,  $d = 2$  et  $W_{f_i}(x)$  et  $W_{g_i}(x)$  sont les bornes de l'erreur de l'approximation de  $f_i$  et  $g_i$  respectivement. Avec  $g_{0i}$  est la valeur minimum de  $b_i$ . La commande (4.38) assure la convergence asymptotique de  $\phi, \theta$  et  $\psi$  vers  $\phi_d, \theta_d$  et  $\psi_d$  respectivement.

**Preuve :** similairement aux sections (§4.4.1) et (§4.4.4), en prenant  $n = 2$ , les équations (4.16-4.26) deviennent :

$$\begin{aligned} U_2 &= \frac{1}{\hat{b}_1} \left( -\hat{f}_1(x, \theta_{f_1}) + \gamma e_{\varepsilon_\phi} + \bar{e}_{s_\phi} + \ddot{\phi}_d \right) + \eta_\phi \text{sat}(e_{s_\phi} / \varepsilon); \\ U_3 &= \frac{1}{\hat{b}_2} \left( -\hat{f}_2(x, \theta_{f_2}) + \gamma e_{\varepsilon_\theta} + \bar{e}_{s_\theta} + \ddot{\theta}_d \right) + \eta_\theta \text{sat}(e_{s_\theta} / \varepsilon); \\ U_4 &= \frac{1}{\hat{b}_3} \left( -\hat{f}_3(x, \theta_{f_3}) + \gamma e_{\varepsilon_\psi} + \bar{e}_{s_\psi} + \ddot{\psi}_d \right) + \eta_\psi \text{sat}(e_{s_\psi} / \varepsilon) \end{aligned} \quad (4.39)$$

D'après les équations (4.31) et (4.32), les lois d'adaptation peuvent être réécrite comme suit :

$$\begin{aligned} \dot{\theta}_{f_1} &= -\eta_{f_1} \varphi_{f_1}(x) e_{\varepsilon_\phi} \\ \dot{\theta}_{f_2} &= -\eta_{f_2} \varphi_{f_2}(x) e_{\varepsilon_\theta} \\ \dot{\theta}_{f_3} &= -\eta_{f_3} \varphi_{f_3}(x) e_{\varepsilon_\psi} \end{aligned} \quad (4.40)$$

**Proposition 2:** pour la conception du contrôleur de l'altitude  $z$ , le sous-système qui gouverne le mouvement  $z(t)$  est présenté comme suit:

$$\begin{pmatrix} \dot{x}_7 \\ \dot{x}_8 \end{pmatrix} = \begin{pmatrix} \dot{z} \\ f_4 - Ug_4U_1 \end{pmatrix}$$

avec  $U = (\cos \phi \cos \theta)$  et  $g_4 = \frac{1}{m}$  et  $f_4 = g$  représente la gravité.

La commande du mode glissant obtenue (4.40) assurant la convergence vers la consigne  $z_d$

$$U_1 = \frac{1}{Ug_4} (-\hat{f}_4(x) + \ddot{z}_d + \gamma e_{\varepsilon z} + \bar{e}_{s_z}) + \frac{(W_{f_4}(x)/(U) + W_{g_4}(x)|u_{eq}|)}{g_0} \text{sat}(e_{s_z} / \varepsilon) \quad (4.40)$$

où  $W_{f_4}(x)$  et  $W_{g_4}(x)$  sont les bornes connues de l'erreur de l'approximation de  $g_4$  et  $f_4$  respectivement.

**Preuve :** Suivant la même procédure précédente §4.4.1 et §4.4.4, considérant :

$$z^d = f_4(x) - Ug(x)U_1(t) \text{ où } U = \cos \phi \cos \theta$$

Le terme de contrôle **d'équivalence** est défini comme :

$$u_{eq_z} = -\frac{1}{Ug_4} (\hat{f}_4(x|\theta_{f_4}) + \ddot{z}_d + \gamma e_{\varepsilon z} + \bar{e}_{s_z}) \quad (4.41)$$

Où  $v(t) = \ddot{z}_d + \gamma e_{\varepsilon z} + \bar{e}_{s_z}$  et  $\gamma > 0$

après la 2<sup>ème</sup> dérivée de l'erreur de sortie et d'après (4.16) on trouve :

$$\ddot{e} = \ddot{z}_d - f_4(x) - \frac{g_4}{\hat{g}_4} (-\hat{f}_4(x) + v(t)) - (Ug_4)u_s \quad (4.42)$$

Prenons les deux premiers termes comme suit :

$$\ddot{z}_d - f_4(x) = (-\hat{f}_4(x) + v(t)) - f_4(x) + \hat{f}_4(x) - \gamma e_{\varepsilon z} - \bar{e}_{s_z}$$

En substituant les deux termes dans l'équation (4.42), et puisque  $\bar{e}_{s_z} = \dot{e}_s - e^{(d)}$  on trouve :

$$\dot{e}_s + \gamma e_s = (\hat{f} - f) + (\hat{g} - g)Uu_{eq} - (Ug)u_s \quad (4.43)$$

Selon la fonction candidate de Lyapunov  $V_i$  :

$$V_i = \frac{1}{2} e_{\varepsilon z}^2 + \frac{1}{2\eta_f} \tilde{\theta}_{f_4}^T \tilde{\theta}_{f_4} + \frac{1}{2\eta_g} \tilde{\theta}_{g_4}^T \tilde{\theta}_{g_4} \quad (4.44)$$

Avec  $\eta_f > 0$  et  $\eta_g > 0$  sont les gains d'adaptation. La dérivé de  $V_i$  nous donne:

$$\dot{V}_i = e_\varepsilon \dot{e}_\varepsilon + \frac{1}{\eta_f} \tilde{\theta}_f^T \dot{\tilde{\theta}}_f + \frac{1}{\eta_g} \tilde{\theta}_g^T \dot{\tilde{\theta}}_g$$

$$\dot{V}_i = -\gamma e_{\varepsilon_z}^2 + \left( \tilde{\theta}_f^T \phi_f(x) - w_f(x) + U \tilde{\theta}_g^T \phi_g(x) u_{eq} - U w_g(x) u_{eq} - U g(x) u_s \right) e_{\varepsilon_z} + \frac{1}{\eta_f} \tilde{\theta}_f^T \dot{\tilde{\theta}}_f + \frac{1}{\eta_g} \tilde{\theta}_g^T \dot{\tilde{\theta}}_g$$

(4.45)

Supposons que les paramètres idéals sont constants  $\dot{\tilde{\theta}}_f = \dot{\theta}_f$  et  $\dot{\tilde{\theta}}_g = \dot{\theta}_g$ , et les lois d'adaptation choisies :

$$\dot{\theta}_{f_4}(t) = -\eta_{f_4} \varphi_{f_4}(x) e_{\varepsilon_z} \quad (4.46)$$

$$\dot{\theta}_{g_4}(t) = -\eta_{g_4} \varphi_{g_4}(x) e_{\varepsilon_z} u_{eq_z} (\cos \phi \cos \theta) \quad (4.47)$$

On trouve que :

$$\frac{1}{\eta_g} \tilde{\theta}_g^T \dot{\tilde{\theta}}_g = -\tilde{\theta}_g^T \varphi_g(x) e_{\varepsilon_z} u_{eq} \text{ et } \frac{1}{\eta_f} \tilde{\theta}_f^T \dot{\tilde{\theta}}_f = -\tilde{\theta}_f^T \varphi_f(x) e_{\varepsilon_z}, \text{ donc l'équation (4.45) s'écrira}$$

comme :

$$\dot{V}_i = -\gamma e_{\varepsilon_z}^2 + \left( -w_f(x) - U w_g(x) u_{eq} - U(g(x)) u_s \right) e_{\varepsilon_z} \quad (4.48)$$

Pour l'analyse de stabilité, nous concluons que :

$$\dot{V}_i \leq -\gamma e_{\varepsilon_z}^2 + \left( -w_f(x) - U w_g(x) u_{eq} - U(g(x)) u_s \right) e_{\varepsilon_z} \quad (4.49)$$

Pour assurer que l'équation (4.49) soit inférieure ou égale à zéro, on choisit :

$$u_s = \frac{\left( W_f(x) / U + W_g(x) |u_{eq}| \right)}{g_0} \text{sat}(e_s / \varepsilon) = \eta(x) \text{sat}(e_s / \varepsilon)$$

On constate que :  $-(w_f(x) + w_g(x) u_{eq}) e_s \leq (|w_f(x)| + |w_g(x) u_{eq}|) |e_s|$ , donc :

$$\dot{V}_i = -\gamma e_s^2 + \left( -w_f(x) - U w_g(x) u_{eq} - U \left( \frac{g(x)}{g_0} \right) \left( W_f(x) / U + W_g(x) |u_{eq}| \right) \text{sat}\left(\frac{e_s}{\varepsilon}\right) \right) e_\varepsilon$$

$$\dot{V}_i \leq -\gamma e_{\varepsilon_z}^2 + \left( |w_f(x)| + |U w_g(x) u_{eq}| \right) |e_{\varepsilon_z}| - U e_{\varepsilon_z} g(x) \left( \frac{1}{g_0} \left( W_f(x) / U + W_g(x) |u_{eq}| \right) \text{sat}\left(\frac{e_s}{\varepsilon}\right) \right)$$

(4.50)

Maintenant, on considère le dernier terme de l'équation (4.50) et comme  $\frac{g(x)}{g_0} \geq 1$  alors :

$$\dot{V}_i = -\gamma e_{\varepsilon_z}^2 + |w_f(x)| |e_s| - e_s \text{sat}(e_s / \varepsilon) W_f(x) + |U w_g(x) u_{eq}| |e_s| - U e_s \text{sat}(e_s / \varepsilon) W_g(x) |u_{eq}|$$

Avec  $|e_s| = e_s \text{sat}(e_s / \varepsilon)$  et  $|w_f(x)| \leq W_f(x)$  et  $|w_g(x)| \leq W_g(x)$

$$|w_f(x)| |e_\varepsilon| - \underbrace{e_\varepsilon \text{sat}(e_s / \varepsilon)}_{|e_\varepsilon|} W_f(x) = |e_\varepsilon| (|w_f(x)| - W_f(x)) \leq 0 \text{ et}$$

$$|U w_g(x) u_{eq}| |e_\varepsilon| - U e_\varepsilon \text{sat}(e_s / \varepsilon) W_g(x) |u_{eq}| \begin{cases} = U |e_\varepsilon| (|w_g u_{eq}| - W_g |u_{eq}|) \leq 0 \text{ si } U \geq 0 \rightarrow U = |U| \\ = |U| |e_\varepsilon| (|w_g u_{eq}| + W_g |u_{eq}|) \leq 0 \text{ si } U < 0 \rightarrow U = -|U| \end{cases}$$

Donc :

$$\dot{V}_i = -\gamma e_\varepsilon^2 + \underbrace{|e_\varepsilon| (|w_f(x)| - W_f(x))}_{\leq 0} + \underbrace{|u_{eq}| |e_\varepsilon| (|U w_g(x)| - U W_g(x))}_{\leq 0} \Rightarrow \dot{V}_i \leq -\gamma e_\varepsilon^2 \quad (4.51)$$

Puisque  $\gamma e_\varepsilon^2 \geq 0$ , alors  $V_i$  est une fonction non croissante dans le temps.

**Proposition 3:** Pour la commande de  $x$  et  $y$ , les sous systèmes qui gouvernent la dynamique de position  $(x, y)$  sont :

$$\begin{pmatrix} \dot{x}_9 \\ \dot{x}_{10} \end{pmatrix} = \begin{pmatrix} \dot{x} \\ u_x \frac{1}{m} U_1 \end{pmatrix} \quad \begin{pmatrix} \dot{x}_{11} \\ \dot{x}_{12} \end{pmatrix} = \begin{pmatrix} \dot{y} \\ u_y \frac{1}{m} U_1 \end{pmatrix}$$

les commandes adaptatives indirectes de type mode glissant flous (4.52):

$$\left. \begin{aligned} u_x &= \frac{\hat{m}}{U_1} (x_d^{(n)} + \gamma e_{\varepsilon_x} + \bar{e}_{sx}) + \eta_x(x) \text{sat}(e_{s_x} / \varepsilon) \\ u_y &= \frac{\hat{m}}{U_1} (y_d^{(n)} + \gamma e_{\varepsilon_y} + \bar{e}_{sy}) + \eta_y(x) \text{sat}(e_{s_y} / \varepsilon) \end{aligned} \right\} \quad (4.52)$$

assurent la stabilité vers la trajectoire désirée asymptotiquement.

$$\text{où } \eta_x(x) = \frac{(+W_{g_5}(x) |u_{eq}|)}{g_{05}} \text{ et } \eta_y(x) = \frac{(+W_{g_6}(x) |u_{eq}|)}{g_{06}} \text{ avec } W_{g_5}(x) \text{ et } W_{g_6}(x) \text{ sont les bornes}$$

connues de l'erreur de l'approximation des fonctions  $g_5$  et  $g_6$  respectivement.

**Preuve :** Pour le sous-système qui gouverne la variable  $x$ , on a :  $\hat{g} = 1/m \rightarrow u_x \frac{1}{m} U_1 = u_x \hat{g}_x U_1$  et avec le choix de la fonction candidate de Lyapunov (4.25), la dérivée de cette dernière est donnée par:

$$\dot{V}_i = -\gamma e_\varepsilon^2 + (U_1 \tilde{\theta}_g^T \phi_g(x) u_{eq} - U_1 w_g(x) u_{eq} - U_1 (g_x) u_s) e_\varepsilon + \frac{1}{\eta_g} \tilde{\theta}_g^T \dot{\tilde{\theta}}_g \quad (4.53)$$

On prend la loi d'adaptation suivante :  $\dot{\theta}_g(t) = -\eta_g \phi_g(x) e_{\varepsilon_x} u_{eq} U_1$

Si le paramètre  $g$  est constant, alors la loi d'adaptation pour la position  $x, y$  est :

$$\dot{\theta}_g(t) = -\eta_g e_\varepsilon u_{eq} U_1 \text{ et } U_1 \tilde{\theta}_g^T \phi_g(x) u_{eq} \rightarrow U_1 \tilde{\theta}_g^T u_{eq}. \quad (4.54)$$

$$\begin{aligned} \dot{V}_i &= -\gamma e_\varepsilon^2 + \left( U_1 \tilde{\theta}_g^T \phi_g(x) u_{eq} - U_1 w_g(x) u_{eq} - U_1 (g(x)) u_s \right) e_\varepsilon - U_1 \tilde{\theta}_g^T \phi_g(x) e_\varepsilon u_{eq} \\ &\Rightarrow \dot{V}_i \leq -\gamma e_\varepsilon^2 + \left( -U_1 w_g(x) u_{eq} - U_1 (g_x) u_s \right) e_\varepsilon \end{aligned} \quad (4.55)$$

Pour assurer que l'équation (4.55) soit inférieure ou égale à zéro, on choisit  $u_s$  comm suit:

$$u_{s_x} = \frac{\left( +W_{g5}(x) |u_{eq}| \right)}{g_{05}} \text{sat}(e_{s_x} / \varepsilon) \quad (4.56)$$

On constate que :  $-(+w_g(x) u_{ce}) e_\varepsilon \leq \left( +|w_g(x) u_{ce}| \right) |e_\varepsilon|$  donc :

$$\begin{aligned} \dot{V}_i &= -\gamma e_\varepsilon^2 + \left( -U_1 w_g(x) u_{ce} - U_1 \left( \frac{g_5}{g_{05}} \right) \left( W_{g5}(x) |u_{ce}| \right) \text{sat}(e_s / \varepsilon) \right) e_\varepsilon \\ \dot{V}_s &\leq -\gamma e_\varepsilon^2 + \left( |U_1 w_g(x) u_{eq}| |e_\varepsilon| - U_1 e_\varepsilon \left( \frac{g_5}{g_{05}} \left( W_{g5}(x) |u_{eq}| \right) \text{sat}(e_s / \varepsilon) \right) \right) \end{aligned} \quad (4.57)$$

Comme:  $\left( \frac{g_x}{g_{0x}} \right) \geq 1$  et  $U_1 \geq 0$  et  $\forall e_s \neq 0 \quad |e_\varepsilon| = e_\varepsilon \text{sat}(e_s / \varepsilon)$

Alors,  $\dot{V}_i = -\gamma e_\varepsilon^2 + |U_1 w_g(x) u_{eq}| |e_\varepsilon| - U_1 e_\varepsilon \text{sat}(e_s / \varepsilon) W_g(x) |u_{eq}|$

avec  $|w_f(x)| \leq W_f(x)$  et  $|w_g(x)| \leq W_g(x)$  :

$|U_1 w_g(x) u_{eq}| |e_\varepsilon| - U_1 e_\varepsilon \text{sat}(e_s / \varepsilon) W_{g5}(x) |u_{eq}| = U_1 |e_\varepsilon| \left( |w_g(x) u_{eq}| - W_{g5}(x) |u_{eq}| \right) \leq 0$  donc

$$\dot{V}_i \leq -\gamma e_\varepsilon^2 \quad (4.58)$$

Puisque  $\gamma e_\varepsilon^2 \geq 0$ , alors  $V_i$  est une fonction non croissante dans le temps.

Suivant la même procédure pour le sous-système qui gouverne  $y$  on trouve :

$$u_{s_y} = \frac{\left( +W_{g6}(x) |u_{eq}| \right)}{g_{06}} \text{sat}(e_{s_y} / \varepsilon) \quad \text{et} \quad \dot{\theta}_{g6}(t) = -\eta_{g6} e_{sy} u_{eq} U_1 \quad (4.59)$$

Les angles correspondants sont donnés par les deux équation (4.60) suivantes :

$$\begin{cases} \phi_d = \arcsin(u_x \sin \psi_d - u_y \cos \psi_d) \\ \theta_d = \arcsin\left(\frac{1}{\cos \phi} (u_x \cos \psi_d - u_y \sin \psi_d)\right) \end{cases} \quad (4.60)$$

Et pour  $\psi_d \cong 0$  on peut approximer le système (4.54) s'écrit comme:

$$\begin{cases} \cos \psi_d \approx 1 \\ \sin \psi_d \approx 0 \end{cases} \rightarrow \begin{cases} \phi_d = \arcsin(-u_y) \\ \theta_d = \arcsin\left(\frac{1}{\cos \phi} u_x\right) \end{cases}$$

La figure (fig. 4.10) représente la structure globale de contrôleur du quadrirotor.

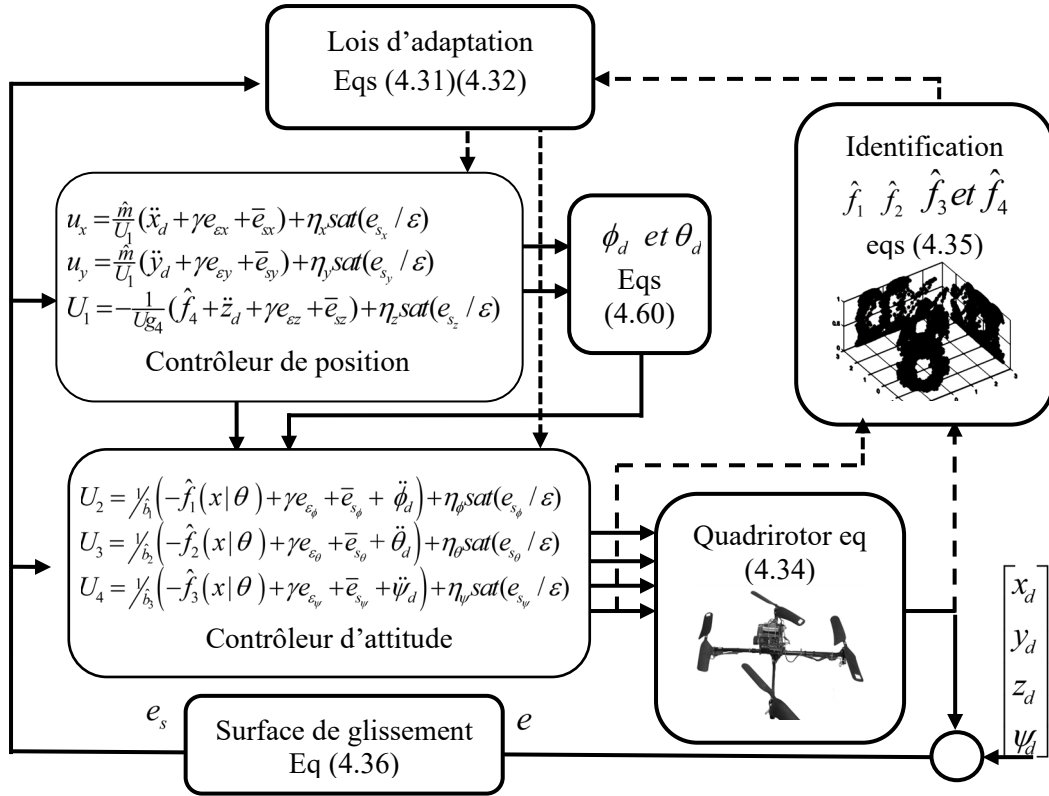


Fig. 4.10 structure globale du contrôleur proposé

### 4.5.3 Résultats et discussions

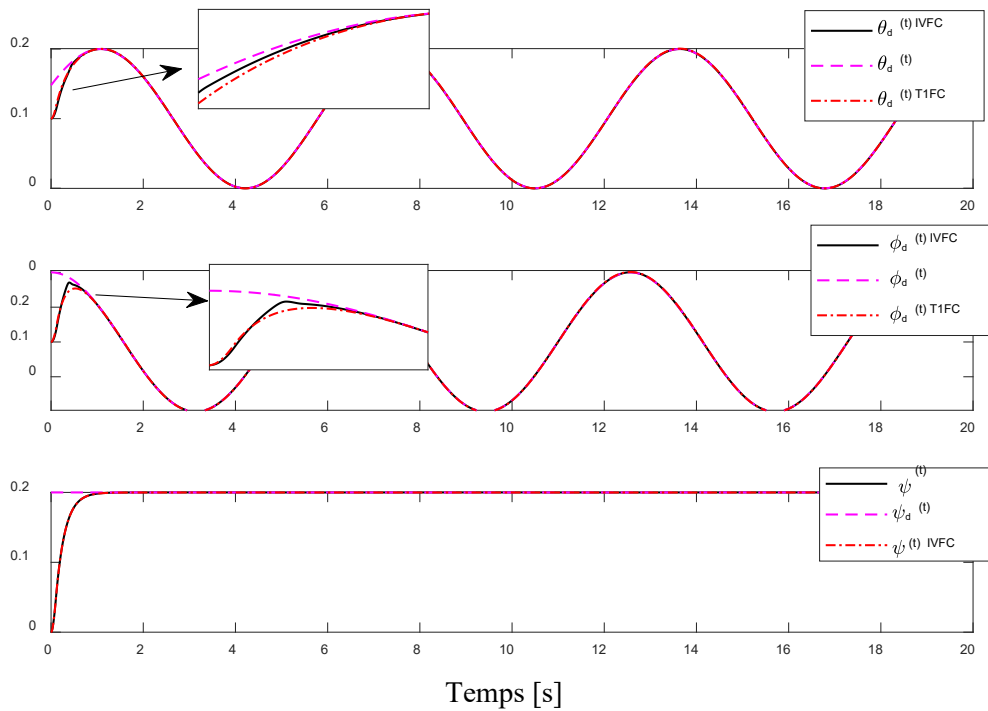
les résultats obtenus en appliquant la loi de commande proposée dans §4.4. Pour cela, on considère différentes trajectoires où le pas d'échantillonnage est fixé à 0.01s. et on choisit  $W_{f_i} = 0.1$  pour  $i = \{1, 2, 3\}$  et  $W_{g_i} = 0.1$  pour  $i = \{1, \dots, 6\}$  pour  $i = \{1, \dots, 6\}$ .  $\gamma = 10$  pour la commande attitude et de la position  $z$ , et  $\gamma = 2$  pour  $x$  et  $y$ . et  $K = [5, 1]$ . Pour vérifier la fiabilité de commande et la robustesse, deux simulations sont réalisées à savoir : le suivi d'une référence variable dans le temps sans perturbation et le suivi d'une référence avec différents niveaux de perturbations.

#### Essai sans perturbation

**Réponse d'attitude :** Cet essai comprend l'application de notre loi de commande au quadrirotor pour étudier ses comportements ainsi que les performances de la commande. Prenant une trajectoire purement sinusoïdale

$$\begin{cases} \theta_d(t) = \sin(t); \varphi_d(t) = \sin(t + \pi) \\ \psi_d(t) = 0.2 \quad z_d(t) = 1 \end{cases} \quad \forall t \quad (4.61)$$

Les angles de référence et les angles réels du système en sortie (en radian) sont donnés par la figure (4.11) et le glissement de l'erreur est présenté dans la figure (fig 4.12).



**Fig. 4.11** Réponses du système avec une trajectoire sinusoïdale

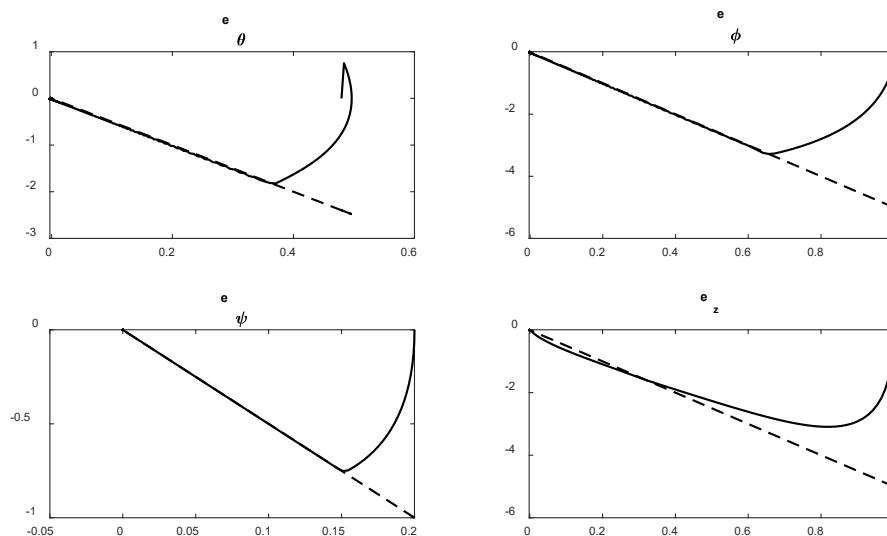


Fig. 4.12 Trajectoire de l'erreur de suivi sur les droites de glissement

Pour une trajectoire plus compliquée de la forme suivante :

$$[\theta_d(t), \varphi_d(t), z_d(t), \psi_d(t)] = \begin{cases} [0, 0, 1, 0.2] & \text{pour } t = [10 \ 12] \\ [\sin(t), \sin(t + \pi), 1, 0.2] & \text{ailleurs} \end{cases} \quad (4.62)$$

Les angles de référence et les angles réels du système en sortie (en radian) sont donnés par la figure (fig. 4.13), la figure (fig. 4.14) indique les signaux des vitesses des rotors.

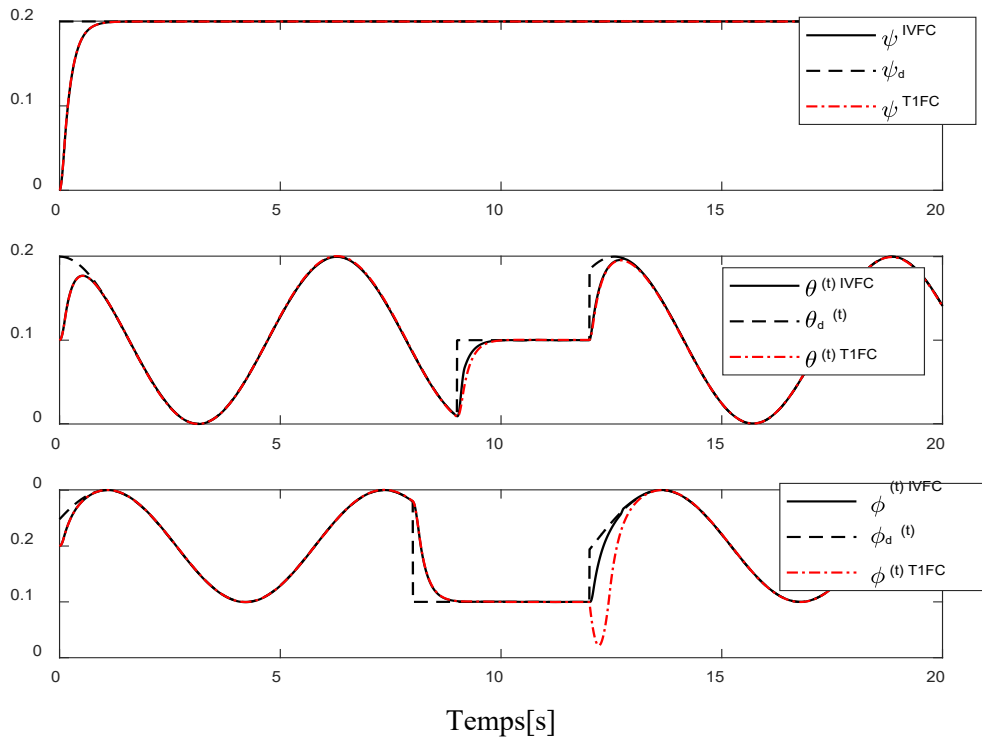


Fig. 4.13 Réponses du système avec une trajectoire sinusoïdale

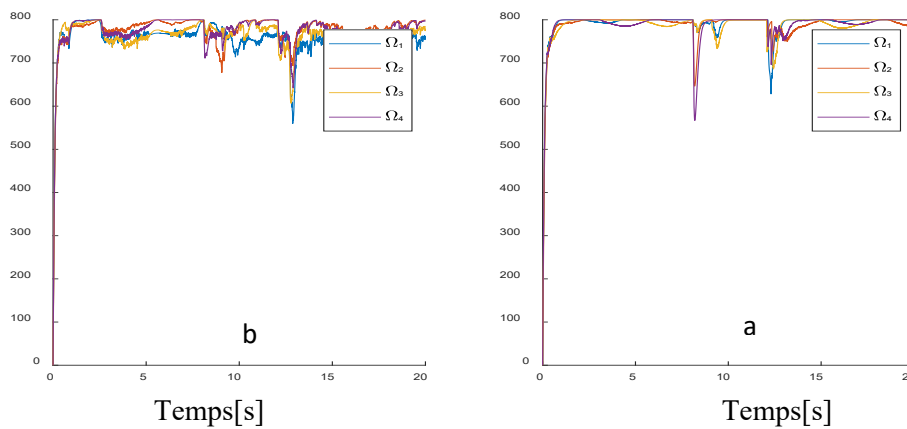
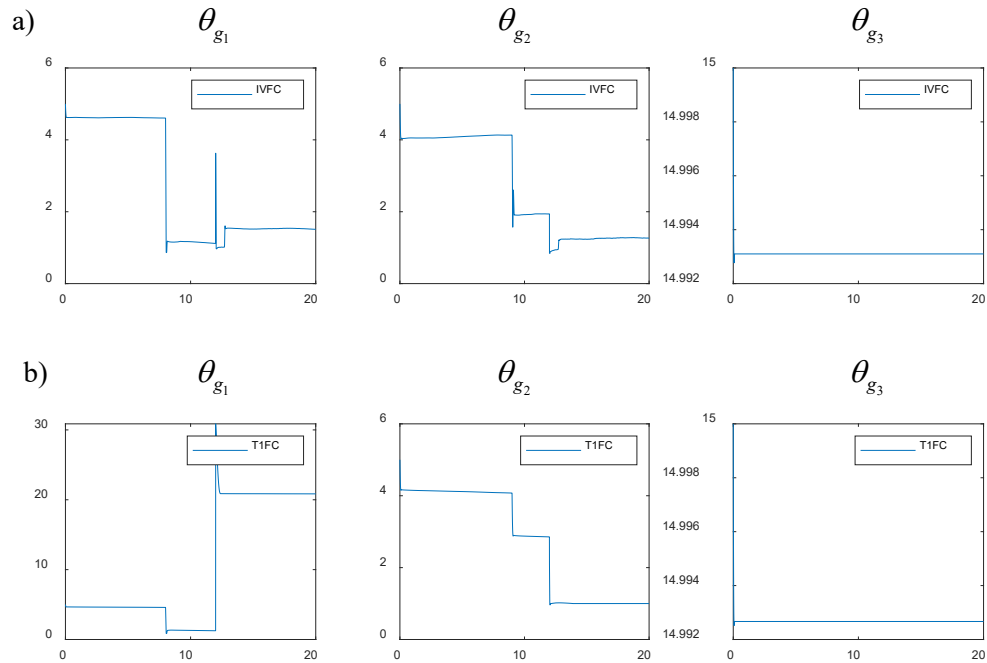
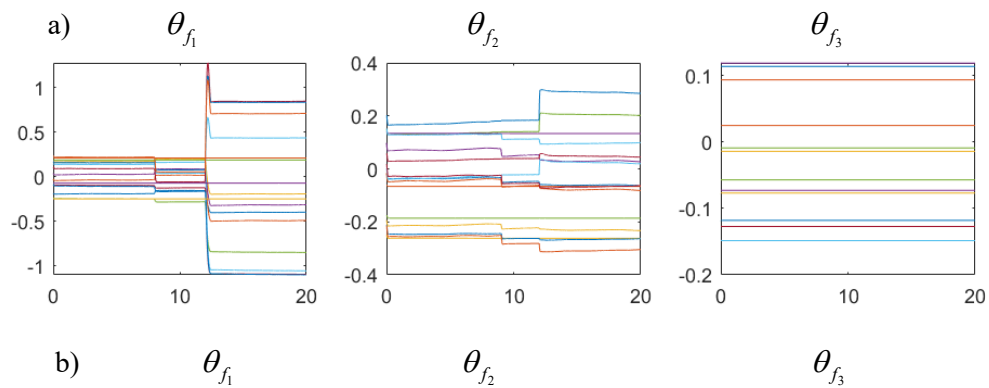


Fig. 4.14 Signaux des vitesses des quatre moteurs (rad/s<sup>2</sup>) (a) commande adaptative IVFC (b) commande adaptative T1FC

Les figures 4.15 et 4.16 montrent l'évolution dans le temps des paramètres d'adaptation du modèle flou. Il est visible que les paramètres convergent très rapidement vers des valeurs stationnaires. Les pics sont dus aux perturbations du système.



**Fig. 4.15** Adaptations des Paramètres  $\theta_{g_1}, \theta_{g_2}$  et  $\theta_{g_3}$  en fonction du temps : a) T1FC b) IVFC



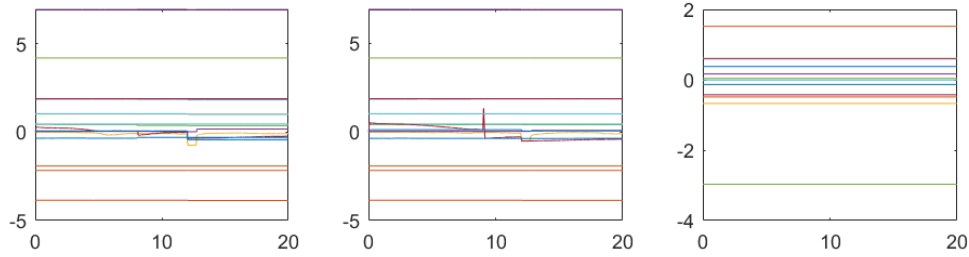


Fig. 4.16 Adaptations des Paramètres  $\theta_{f_1}, \theta_{f_2}$  et  $\theta_{f_3}$  en fonction du temps : a) T1FC b) IVFC

On remarque que le système rejoint le signal de référence. Cependant, en analysant l'erreur de poursuite à travers la figure (4.13) et le tableau (4.2), on note la présence d'une erreur aux extrémums (points maximal et minimal). Ceci peut être expliqué par la présence d'une erreur entre le modèle estimé et le modèle réel simulé. Néanmoins, l'effet de cette erreur est très petit, et on remarque que l'erreur obtenue par notre approche est minimum.

	IVFC	T1FC
MSE $\theta$	22.5090	23.0242
MSE $\varphi$	44.9816	81.8549
MSE $\psi$	1.0554	1.0587

Table. 4.2 Comparaison MSE entre les deux contrôleurs T1FC et IVFC

**Réponse de position:** on considère la trajectoire suivante :

$$\begin{cases} x_d(t) = \sin(t); y_d(t) = \sin(t + \pi) \\ z_d(t) = 1, \psi_d(t) = 0.2 \end{cases} \quad (4.63)$$

En absence de variation paramétrique dans le cas de suivi de position, les résultats de simulation sont presque similaires et suffisamment acceptable ( fig 4.17 ). Les angles correspondants aux réponses de position sont présentées dans (fig 4.18).

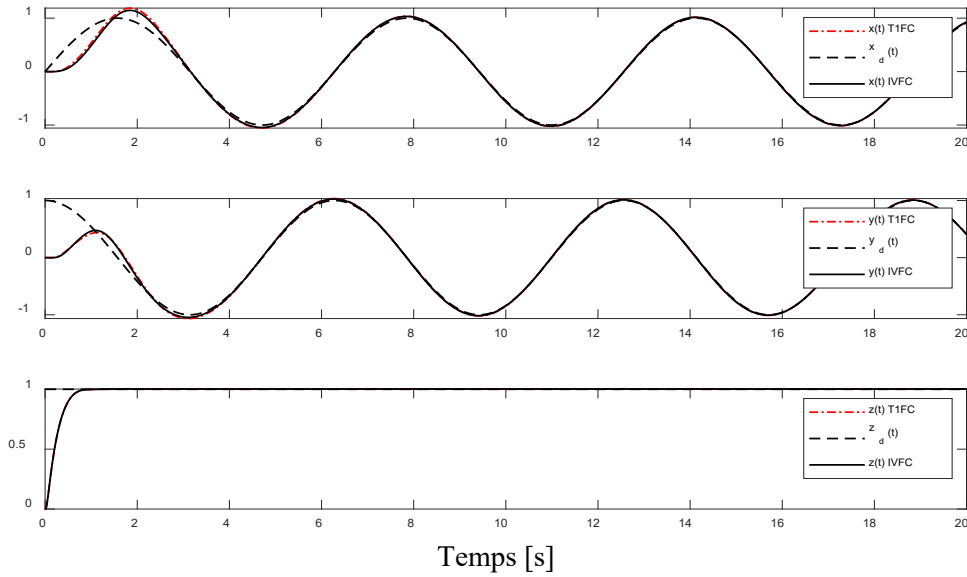


Fig. 4.17 réponse du système avec une trajectoire sinusoïdale

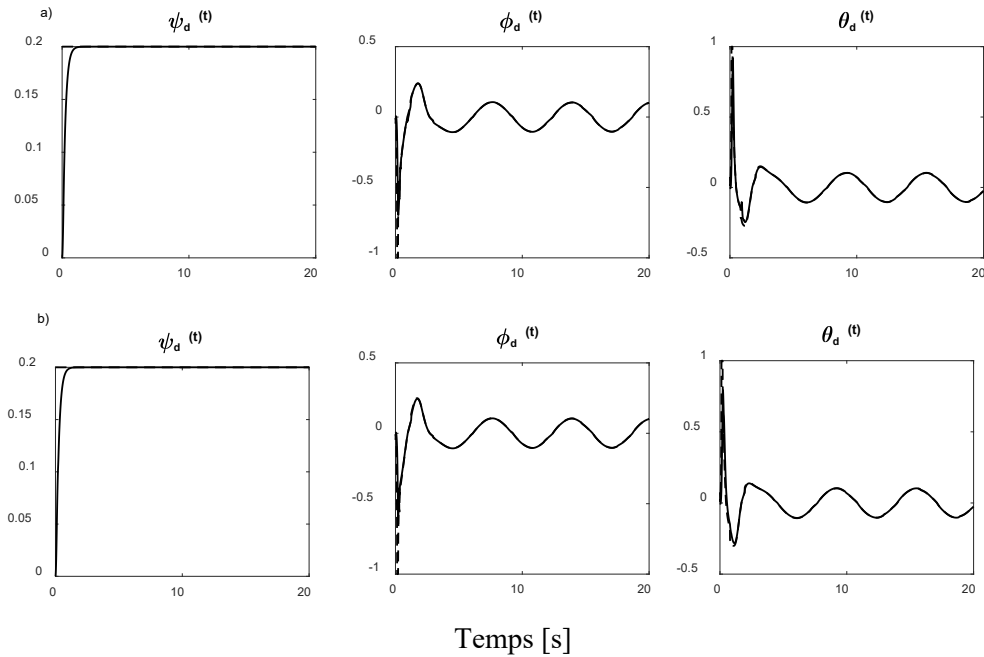


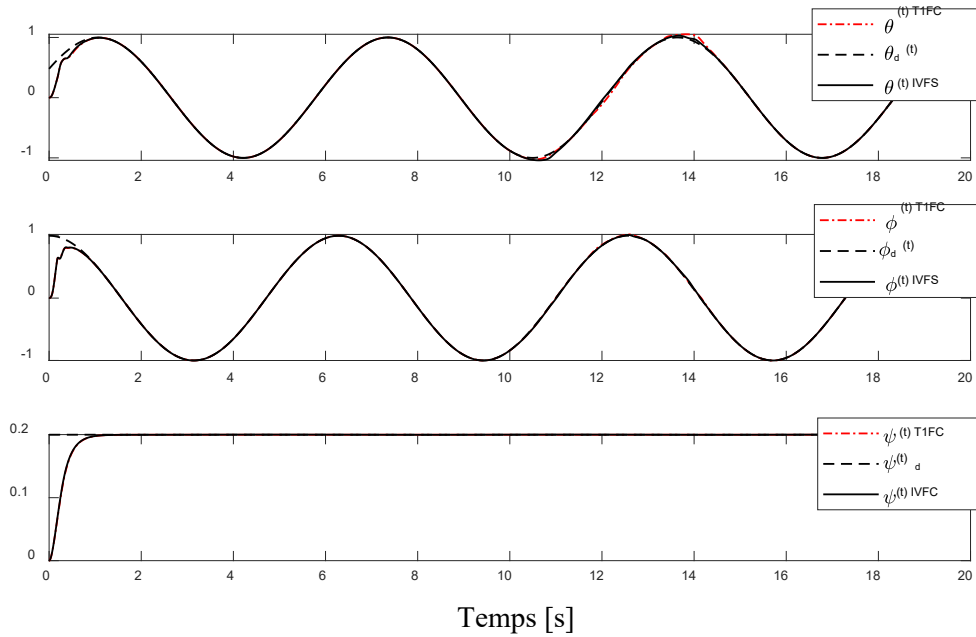
Fig. 4.18 Angles correspondants aux réponses de position a) T1FC b) IVFC

**Essai avec perturbation.**

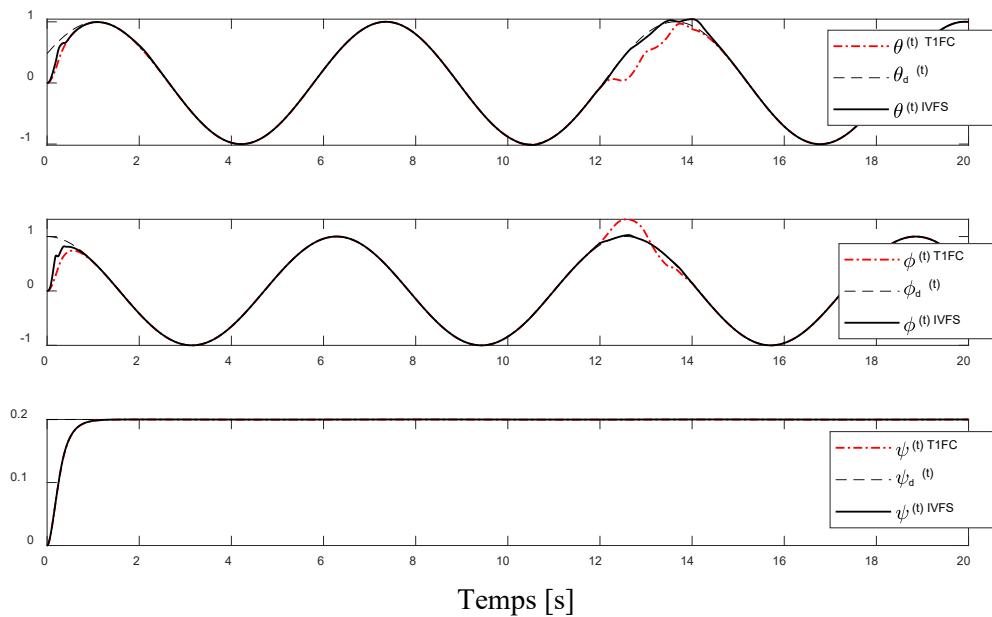
**Réponse d'attitude :** Cet essai comprend l'application de la loi de commande au quadrirotor avec une perturbation dans un intervalle de temps [12s,14s]. Cette dernière est représentée par un changement  $\delta I = 6\%I$  et  $\delta J_r$  sur les valeurs de l'inertie du rotor  $J_r$  et l'inertie du drone ( $I_{xx}, I_{yy}, I_{zz}$ ) qui peut s'écrire :

$$\hat{J}_r = J_r + \delta J_r \text{ et } \hat{I} = I + \delta I \tag{4.64}$$

Les figures (fig. 4.19-4.21) montrent les résultats de poursuite du contrôleur obtenus pour les trois tests mentionnés ci-après. Dans le cas où on introduit des perturbations sur le modèle initial, on remarque une déviation autour des consignes données.



**Fig. 4.19** CMG adaptative en présence de perturbation  $\delta J_r = 10\% J_r$



**Fig. 4.20** CMG adaptative en présence de perturbation  $\delta J_r = 15\% J_r$

Notons que pour toutes les simulations, les perturbations ajoutées sont compensées rapidement par le régulateur IVFC proposé par rapport au régulateur de type-1. Notons aussi dans le tableau (table. 4.3) que la rejection de perturbation par notre approche est plus effective lorsque le système ait subi une perturbation plus importante.

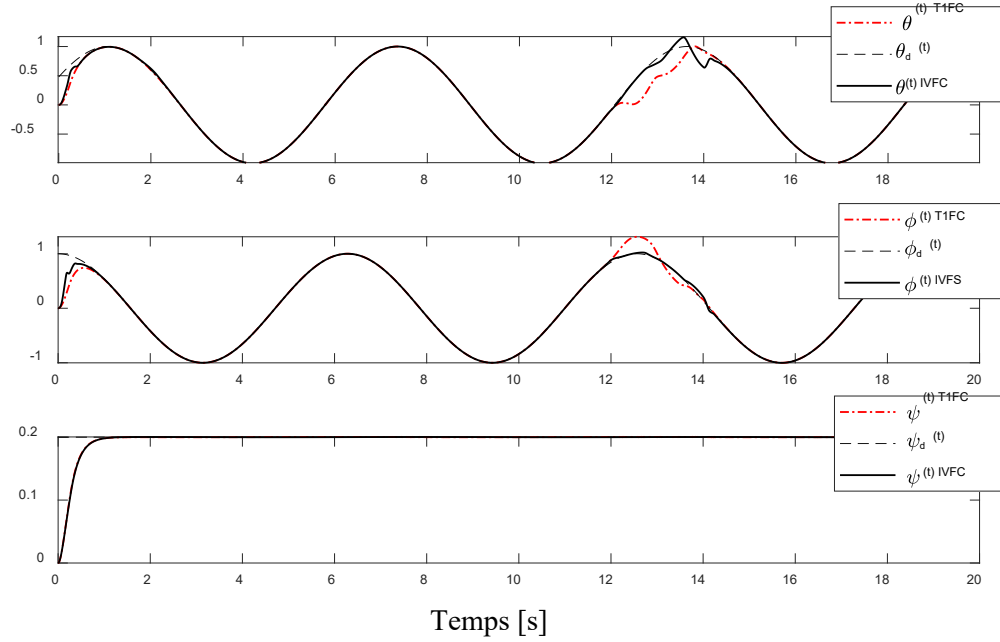


Fig. 4.21 CMG adaptative en présence des perturbations  $\delta J_r = 20\% J_r$

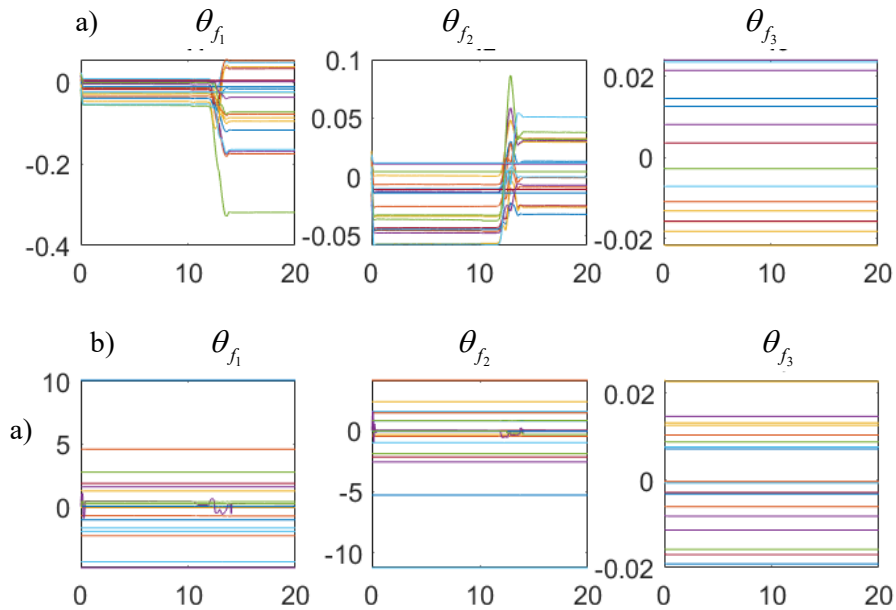


Fig. 4.22 Adaptation des paramètres des  $f_i(x)$  en fonction du temps : a) T1FC b) IVFC

b)	Test3		Test2		Test1	
	IVFC	T1FC	IVFC	T1FC	IVFC	T1FC
MSE $\tilde{\theta}$	7.4922	13.2162	7.5905	9.6931	8.0607	9.1189
MSE $\tilde{\varphi}$	17.5483	49.8297	17.3102	32.5082	17.2950	31.0333
MSE $\tilde{\psi}$	1.0554	1.0561	1.0554	1.0554	1.0554	1.0537

Table 4.3 Comparaison des MSE pour les différents tests

La figure (fig. 4.22) montre l'évolution dans le temps des paramètres du modèle flou  $f_i(x)$ . Il est visible que les paramètres convergent très rapidement vers des valeurs stationnaires. Les pics sont dus aux perturbations du système. La figure (fig. 4.23) montre les commandes  $U_1, U_2, U_3$  et  $U_4$  du contrôleur.

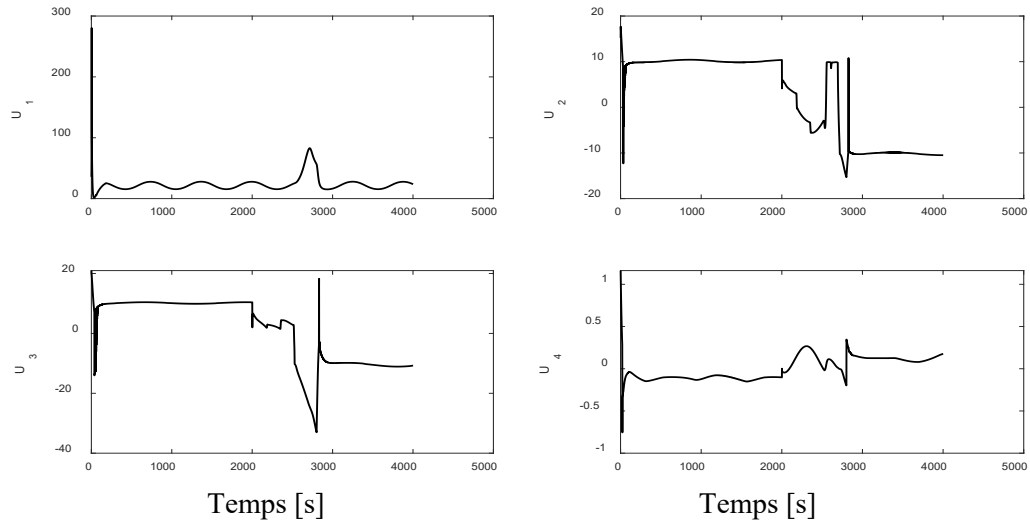


Fig. 4.23 Les signaux de commande  $U_1, U_2, U_3$  et  $U_4$

**Réponse de position :** Dans cette simulation, le modèle a subi une perturbation externe sur les angles  $\varphi_d$  et  $\theta_d$  qui attaquent le contrôleur de position. Cette perturbation additive prend comme valeurs:  $\delta\varphi = 0.25rad$  dans l'intervalle [8s 8.5s] et  $\delta\theta = 0.25rad$  dans l'intervalle [9s 9.5s], comme indiquée dans (fig. 4.24) et la figure (4.25) montre la réponse de suivi de position.

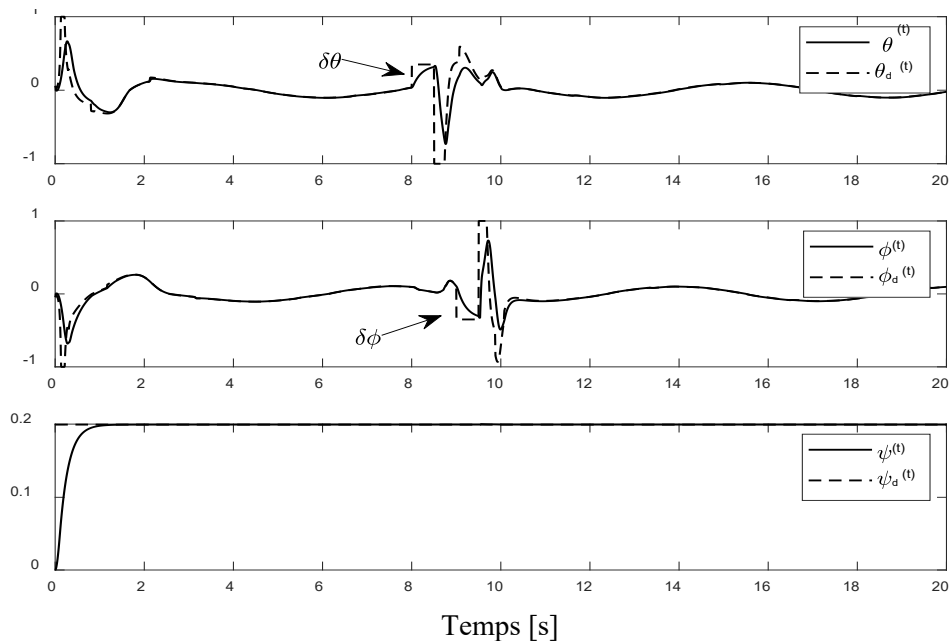


Fig. 4.24 Angles correspondant avec perturbation  $\delta\varphi = \delta\theta = 0.25rad$

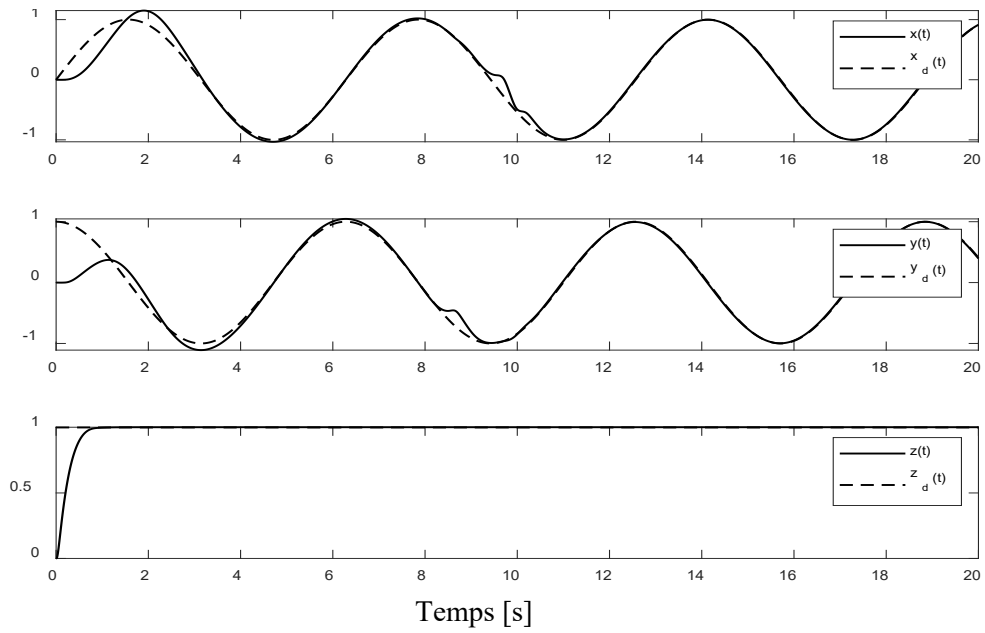


Fig. 4.25 Réponse de système utilisant la CMG adaptative basée sur le modèle flou

les résultats des figures (fig. 4.25) et le tableau de MSE de tableau 4.4 montrent l'efficacité et la robustesse de la commande proposée.

	T1FC	IVFC
MSE $\tilde{x}$	46.6442	39.7494
MSE $\tilde{y}$	127.3148	123.8104
MSE $\tilde{z}$	30.7955	30.7955

Table 4.4 Comparaison des MSE entre T1 et IVFC

## 4.6 CONCLUSION

Dans ce chapitre, nous avons présenté une commande adaptative CMG où les dynamiques inconnues sont regroupées et approximées par un modèle flou. Ce dernier, initialement obtenu par clustering et dont la mise à jour est effectuée à l'aide d'une loi déduite de l'étude de stabilité. Au niveau de la robustesse, nous avons utilisé une commande adaptative type-2 (IVFC) pour son amélioration en boucle fermée.

Une étude comparative a été présentée pour mieux illustrer les améliorations apportées par l'approche proposée avec validation par plusieurs simulations. Différentes situations ont été envisagées afin de tester d'une part, les performances de poursuite et d'autre part la robustesse. Les résultats ont permis de confirmer la convergence de l'erreur de poursuite malgré les perturbations externes et internes.

# Conclusions et perspectives

Durant ces dernières années, la robotique aérienne a fait de grand progrès. Le travail réalisé dans cette thèse entre dans la commande de mini-véhicules aériens quadrirotors. Le but est d'apporter une contribution aux travaux déjà menés dans le cadre de l'association de la logique floue et de la commande nonlinéaire adaptative. Il s'agit de développer des lois de commande adaptatives par mode glissant pour résoudre les problèmes de poursuite du quadrirotors.

Dans le **chapitre 1**, nous avons entamé ce travail par une étude théorique reposant sur une introduction sur les concepts généraux des drones de type quadrirotors. Ensuite, on a présenté la modélisation des quadrirotors et leur analyse détaillée ainsi que les hypothèses fondamentales considérées. Le modèle dynamique a été calculé en utilisant la méthode de Newton-Euler.

Le **chapitre 2** a traité le problème d'identification (construction) des modèles flous à partir de données entrée-sortie. Nous avons introduit d'abord la modélisation floue de systèmes, en se focalisant particulièrement sur le modèle de type Takagi-Sugeno. Enfin, après avoir considéré la méthodologie générale pour la construction de modèles flous Takagi Sugeno à partir des données, les méthodes d'identification des paramètres des conséquents ont été présentées.

Dans le **chapitre 3**, le cadre formel permettant la synthèse du modèle flou de type-2 est présenté. Ensuite, une nouvelle technique floue type-2 est proposée et utilisée pour approximer les différents paramètres des systèmes. Ce modèle flou TS a été construit en trois étapes :

Étape 1 : Génération de clusters de données floues en utilisant le clustering flou dans l'espace produit cartésien des entrées et des sorties.

Étape 2 : Estimation des fonctions d'appartenance de type-2 antécédent utilisant un nouvel algorithme de détection d'enveloppe (EDA).

Étape 3 : Calcul des paramètres des fonctions conséquentes en utilisant les moindres carrés pour les antécédents et le Filtre de Kalman pour les conséquents.

Dans le **chapitre 4**, nous avons présenté une synthèse de la commande adaptative floue indirecte par mode glissant sur le système UAV. Pour cela, nous avons développé une

estimation des fonctions inconnues par le système flou TS. De ce fait, l'adaptation est faite pour ajuster les fonctions estimées pour atteindre l'objectif de contrôle. L'utilisation du modèle initial estimé hors ligne par clustering est introduit pour contourner le retard de la convergence au début du processus d'adaptation.

Notre contribution s'est portée sur deux aspects : La construction du modèle approximé flou type- 2 en utilisant les techniques de clustering. cette approximation prend en compte les incertitudes négligées dans le cas d'un système type-1 et ainsi améliorer l'approximation. Le deuxième aspect, concerne la commande à base de modèle dont les paramètres initiaux de l'approximation ont été auto ajustés.

Nous considérons à présent, par la suite quelques-unes des perspectives qui nous semble les plus intéressantes à savoir :

- L'extension de l'analyse de la stabilité pour le système constitué du modèle flou et commande floue de type affine Takagi-Sugeno en temps discret. Bien que certains travaux ont été réalisés dans cette direction.
- L'étude approfondie des équivalences, au niveau du comportement dynamique, entre le modèle flou de type Takagi-Sugeno et le système non linéaire approximé.
- Le développement d'un observateur flou basé sur le modèle affine Takagi-Sugeno, afin d'estimer les variables non mesurables du système, comme c'est typiquement le cas dans le domaine des drones.
- L'adaptation en ligne dans le processus de clustering, pour obtenir une approximation adaptative dans le cas des systèmes à modèle continu.
- L'extension de la synthèse de lois de commandes pour le suivi d'attitude et de position d'un drone quadrirotor, dont les actionneurs sont soumis à des saturations.

# Bibliographie

- [1]. S. Bouabdallah, Design And Control Of Quadrotors With Application To Autonomous Flying These Doctorate. Doi: 10.5075/Epfl-Thesis-3727 (2007)
- [2]. A. Zanotti, Rotary Wing Aerodynamics. *Energies*, 15, 2072. <https://doi.org/10.3390/en15062072> (2022)
- [3]. H. Shakhathreh, A. Sawalmeh, A. Al-Fuqaha, Z. Dou, E. Almaita, I. Khalil, . Othman, A. Khreishah, M. Guizani “Unmanned Aerial Vehicles: A Survey On Civil Applications And Key Research Challenges” (2019).
- [4]. S. Bouabdallah, P. Murrieri, And R. Siegwart.” Design And Control Of An Indoor Micro Quadrotor. In *Robotics And Automation, 2004. Proceedings. Iccra'04. 2004 Ieee International Conference On, Volume 5*, Pp. 4393-4398,. Isbn 0780382323. (2004)
- [5]. S. Bouabdallah, A. Noth, And R. Siegwart.” Pid Vs Lq Control Techniques Applied To An Indoor Micro Quadrotor”. Technical Report, Swiss Federal Institute Of Technology, (2004).
- [6]. J. Michael; A. Stepaniak “Quadrotor Sensor Platform”. Phd Thesis, Ohio University, (2008)
- [7]. A. Tayebi And S. Mcgilvray. “Attitude Stabilization Of A VTOL Quadrotor Aircraft. *Ieee Transactions On Control Systems Technology*,” 14(3):562-571, Issn 1063-6536. Doi: 10.1109/Tcst.2006.872519. (2006).
- [8]. T. Kim, K. Stol, V. Kecman. “Control Of 3 Dof Quadrotor Model”. *Robot Motion And Control 2007*, Pp 29-38, (2007).
- [9]. P. Pounds, R. Mahony, J. Gresham, P. Corke, J. Roberts. “Towards Dynamically-Favourable Quad-Rotor Aerial Robots”. In *Australian Conference On Robotics And Automation*,(2004).
- [10]. P Pounds; R. Mahony; P. Corke. “Modelling And Control Of A Quad-Rotor Robot”. In *Australian Conference On Robotics And Automation*, Page 10, (2006).
- [11]. T. Hamel, R. Mahony, R. Lozano, And J. Ostrowski. “Dynamic Modelling And Configuration Stabilization For An X4-Flyer”. *IFAC 15th Triennial World Congress*, Page 6, (2002).
- [12]. N., Ben Ammar; S., Bouallegue; J. ,Haggege; “Modeling And Sliding Mode Control Of A Quadrotor Unmanned Aerial Vehicle”. In *Proceedings Of The 3rd International Conference On Automation, Control, Engineering And Computer Science*; (2016) ; Pp. 834–840. Available Online: [Http://Ipco-Co.Com/Pet\\_Journal/Acecs-2016/134.Pdf](http://ipco-co.com/Pet_Journal/Acecs-2016/134.Pdf)
- [13]. K. Runcharoon; V.Srichatrapimuk, “Sliding Mode Control Of Quadrotor”. In *Proceedings Of The 2013 The International Conference On Technological Advances In Electrical, Electronics And Computer Engineering (TAEECE)*, Konya, Turkey, Pp. 552–557. (2013).
- [14]. R. Xu; U. Ozguner, “Sliding Mode Control Of A Quadrotor Helicopter”. In *Proceedings Of The 45th IEEE Conference On Decision And Control*, San Diego, Ca, Usa, Pp. 4957–4962. (2006).

- [15]. A. Sudhir; Swamp, Second Order Sliding Mode Control For Quadrotor. In Proceedings Of The 2016 IEEE First International Conference On Control, Measurement And Instrumentation (Cmi), Kolkata, India, Pp. 92–96. (2016).
- [16]. Y., Shtessel; C., Edwards; Fridman, L.; Levant, A. “Introduction: Intuitive Theory Of Sliding Mode Control”. In Sliding Mode Control And Observation; Springer: New York, Ny, Usa, Pp. 1–42 (2014).
- [17]. K.J, Åström.; B., Wittenmark; “Adaptive Control”; Courier Corporation: Chelmsford, Ma, Usa, (2013).
- [18]. C Nicol; C. Macnab; A Ramirez-Serrano, “Robust Neural Network Control Of A Quadrotor Helicopter”. In Proceedings Of The 2008 Canadian Conference On Electrical And Computer Engineering, Niagara Falls, On, Canada, 4–7, Pp. 1233–1238. (2008);
- [19]. S. Li; Y. Wang; J. Tan; Y. Zheng, “Adaptive Rbfnn/Integral Sliding Mode Control For A Quadrotor Aircraft”. Neurocomputing ,216, Pp.126–134. (2016).
- [20]. H. Boudjedir; O. Bouhali; N. Rizoug, “Adaptive Neural Network Control Based On Neural Observer For Quadrotor Unmanned Aerial Vehicle”. Adv. Robot. 28,Pp. 1151–1164 (2014)
- [21]. D. Lee; H.J. Kim; S. Sastry, “Feedback Linearization Vs. Adaptive Sliding Mode Control For A Quadrotor Helicopter”. Int. J. Control Autom. Syst. 7, Pp 419–428. (2009).
- [22]. O. Mofid; S. Mobayen, “Adaptive Sliding Mode Control For Finite-Time Stability Of Quad-Rotor Uavs With Parametric Uncertainties”. ISA Trans. 72, Pp. 1–14. (2018).
- [23]. S. Islam, M. Faraz, R. Ashour, G. Cai,; J. Dias, L. Seneviratne, Adaptive Sliding Mode Control Design For Quadrotor Unmanned Aerial Vehicle. In Proceedings Of The 2015 International Conference On Unmanned Aircraft Systems (ICUAS), Denver, Co, Usa,; Pp. 34–39. (2015).
- [24]. C. Coza; C.J. Macnab, “A New Robust Adaptive-Fuzzy Control Method Applied To Quadrotor Helicopter Stabilization”. In Proceedings Of The Nafips 2006-2006 Annual Meeting Of The North American Fuzzy Information Processing Society, Montreal, Qc, Canada, Pp. 454–458. (2006).
- [25]. M. Huang; B. Xian; C. Diao; K. Yang; Y. Feng, “Adaptive Tracking Control Of Underactuated Quadrotor Unmanned Aerial Vehicles Via Backstepping”. In Proceedings Of The 2010 American Control Conference, Baltimore, Md, USA, Pp.. 2076–2081 (2010).
- [26]. T. Madani; A. Benallegue, “Adaptive Control Via Backstepping Technique And Neural Networks Of A Quadrotor Helicopter”. IFAC Proc. Vol., 41, 6513–6518. (2008)
- [27]. Y. Morel, A. Leonessa, “Direct Adaptive Tracking Control Of Quadrotor Aerial Vehicles”. In ASME International Mechanical Engineering Congress And Exposition; Parts A And B; Asme: Chicago, Il, Usa, Pp. 155–161. (2006).
- [28]. Z. Fang, W. Gao, “Adaptive Backstepping Control Of An Indoor Micro-Quadrotor”. Res. J. Appl. Sci. Eng. Technol. 4, Pp 4216–4226. 102. (2012).
- [29]. A. A. Ghaffar; T. Richardson, “Model Reference Adaptive Control And Lqr Control For Quadrotor With Parametric Uncertainties”. Int. J. Mech. Mechatron. Eng.,9,Pp. 244–250. (2015).

- [30]. A.J. Koshkouei; A.S. Zinober, “Adaptive Backstepping Control Of Nonlinear Systems With Unmatched Uncertainty”. In Proceedings Of The 39th IEEE Conference On Decision And Control (Cat. No. 00ch37187), Sydney, Nsw, Australia, 12–15 Dec. 2000; Volume 5, Pp. 4765–4770. Technologies, 9, 37 16 Of 22 (2021).
- [31]. D. Travis; S Jagannathan, "Neural Network Output Feedback Control Of A Quadrotor Uav," In IEEE Conference On Decision And Control, (2008).
- [32]. D. Travis; S Jagannathan, "Output Feedback Control Of A Quadrotor Uav Using Neural Networks," IEEE Transactions On Neural Networks, Vol. 21, No1, (2010).
- [33]. C. Nicol, C.J.B. Macnab; A. Ramiraz-Serrano, "Robust Neural Network Control Of A Quadrotor Helicopter," In IEEE, (2008).
- [34]. H.J. Lee; J.B Park; G. Chen, “ Robust Fuzzy Control Of Nonlinear Systems With Parametric Uncertainties”. IEEE Trans. Fuzzy Syst. 9, 369–379 (2001).
- [35]. C.L. ,Chen, M.H. Chang, Optimal Design Of Fuzzy Sliding-Mode Control: A Comparative Study. Fuzzy Sets Syst., 93, Pp.37–48 (1998).
- [36]. T. Taniguchi; K. Tanaka; H. Ohtake; H.O. Wang, “ Model Construction, Rule Reduction, And Robust Compensation For Generalized Form Of Takagi-Sugeno Fuzzy Systems”. IEEE Trans. Fuzzy Syst, 9, Pp. 525–538. (2001).
- [37]. Y. Chen; J. Li; K. Wang; S. Ning, “Robust Trajectory Tracking Control Of Underactuated Underwater Vehicle Subject To Uncertainties”. J. Mar. Sci. Technol., 25, Pp. 283–298 (2017).
- [38]. F. Yacef; O. Bouhali; M. Hamerlain; N. Rizoug, “Observer-Based Adaptive Fuzzy Backstepping Tracking Control Of Quadrotor Unmanned Aerial Vehicle Powered By Li-Ion Battery”. J. Intell. Robot. Syst 84, Pp. 179–197. (2016).
- [39]. F. Baklouti; S. Aloui; A. Chaari, “Adaptive Fuzzy Sliding Mode Tracking Control Of Uncertain Underactuated Nonlinear Systems: A Comparative Study”. J. Control Sci. Eng., (2016).
- [40]. H. Mo; G. Farid, “Nonlinear And Adaptive Intelligent Control Techniques For Quadrotor UAV–A Survey”. Asian J. Control, 21, 989–1008. (2019).
- [41]. L. Zadeh, “Fuzzy Sets”, Inf. Control, Vol.8, Pp.338-353, (1965).
- [42]. R. Babuška, “Fuzzy Modeling For Control”. Kluwer Academic Publishers, Mass., Usa, (1998).
- [43]. L. X. Wang; “Adaptive Fuzzy Systems And Control, Design And Stability Analysis”. Prentice Hall, New Jersey, (1994).
- [44]. T. Takagi; M. Sugeno, “Fuzzy Identification Of Systems And Its Application To Modeling And Control”, IEEE Trans. On Systems, Man And Cybernetics, Vol.15, N1, Pp.116-132, (1985).
- [45]. J.-S.R. Jang, “Anfis: Adaptive-Network-Based Fuzzy Inference Systems”, IEEE Transactions On Systems, Man And Cybernetics, Vol.23, N3, Pp.665-685, (1993).
- [46]. H. Hellendoorn; D. Driankov, “Fuzzy Model Identification – Selected Approaches”, Springer-Verlag, Berlin, (1997).
- [47]. L.X. Wang, “Adaptive Fuzzy Systems And Control” (Prentice Hall, Englewood Cliffs, Nj) (1994).

- [48]. J. S. Jang: "Neuro-Fuzzy Modeling: Architectures, Analyses And Applications," Ph.D. Thesis, University Of Berkeley-California. (1992).
- [49]. N. Lori, P. Costa Branco, "Autonomous Mountain-Clustering Method Applied To Fuzzy Systems Modeling," In Intelligent Engineering Systems Through Artificial Neural Networks, Smart Engineering Systems: Fuzzy Logic And Evolutionary Programming , Ed. By C. Dagli, M. Akay, C. Philip, C. Chen, B. Fern'andez, J. Ghosh, Vol. 5 (Asme Press, New York, U.S.A.), Pp. 311– 316 (1995).
- [50]. R. Yager, D. Filev: "Essentials Of Fuzzy Modeling And Control" (John Wiley & Sons, Hoboken, Nj, USA) (1994).
- [51]. U. Bodenhofer: "Tuning Of Fuzzy Systems Using Genetic Algorithms," Master's Thesis, Institut F"Ur Mathematik Johannes Kepler Universitat, Linz, Austria (1996).
- [52]. C. Pena: "Coevolutionary Fuzzy Modeling," Ph.D. Thesis, Ecole Polytechnique Fédérale De Lausanne. (2002).
- [53]. V. H. Grisales Palacio, « Modélisation Et Commande Floues De Type Takagi-Sugeno Appliquées A Un Bioprocédé De Traitement Des Eaux Usées ». Thèse De Doctorat De L'université Paul Sabatier - Toulouse Iii, (2007).
- [54]. E. Ruspini, "A New Approach To Clustering, Information And Control", Vol.15, Pp.22-32, (1969).
- [55]. R. Krishnapuram; J. Keller, "A Possibilistic Approach To Clustering", IEEE Transactions On Fuzzy Systems, Vol.1, N2, Pp.98-110, (1993).
- [56]. L. Khodja ; « Contribution A La Classification Floue Non Supervisée ». Thèse De Doctorat De L'université De Savoie, France, (1997).
- [57]. J. Dunn; "A Fuzzy Relative To Isodata Process And Its Use In Detecting Compact Well Separated Clusters", Journal Of Cybernetics, Vol.3, N3, Pp.32-57, (1974).
- [58]. J. Bezdek "Pattern Recognition With Fuzzy Objective Function". Plenum Press, New York, (1981).
- [59]. D.E. Gustafson; W.C. Kessel, "Fuzzy Clustering With A Fuzzy Covariance Matrix", Proceedings Of The IEEE Cdc, San Diego, Ca, Usa, Pp.761-766, (1979).
- [60]. I. Gath, A. Geva, "Unsupervised Optimal Fuzzy Clustering", IEEE Transactions On Pattern Analysis And Machine Intelligence, Vol.7, Pp.773-781, (1989).
- [61]. N. Pal, J.Bezdek, "On Cluster Validity For The Fuzzy C-Means Model", IEEE Transactions On Fuzzy Systems, Vol.3, N3, Pp.370-379, (1995).
- [62]. J. Yu, Q. Cheng, H. Huang, " Analysis Of The Weighting Exponent In The Fcm", IEEE Transactions On Systems, Man And Cybernetics-Part B: Cybernetics, Vol.34, N1, Pp.634-639, (2004).
- [63]. W. Hardle, "Applied Nonparametric Regression". Cambridge, Ma: Cambridge Univ. Press, (1990).
- [64]. J.C. Bezdek And J.C. Dunn. "Optimal Fuzzy Partitions: A Heuristic For Estimating The Parameters In A Mixture Of Normal Distributions". IEEE Transactions On Computers, Pp. 835–838, (1975).
- [65]. R. E. Kalman, "A New Approach To Linear Filtering And Prediction Problems," ASME Transactions, Part D, J. Of Basic Engineering, Vol. 82, Pp. 35-45, (1960).

- [66]. S. G. Mohinder And A. P. Angus, “Kalman Filtering : Theory And Practice Using Matlab,” John Wiley & Sons, (2001).
- [67]. T. K. Nathabandu; R. Renzo, “Statistics, Probability, And Reliability For Civil And Environmental Engineers,” Mc Graw-Hill, (1997).
- [68]. K. Chafaa, “Structures D’identification Et De Commande Des Systèmes Non Linéaires », Thèses De Doctorat , Université De Batna (2006).
- [69]. L.A. Zadeh, “The Concept Of A Linguistic Variable And Its Application To Approximate Reasoning”. *Information Sciences*, Vol. 8, Pp. 199-249, (1975).
- [70]. J.M. Mendel. “Uncertain Rule-Based Fuzzy Logic Systems: Introduction And New Directions”. Prentice-Hall, (2001).
- [71]. J.M. Mendel Et R.I.B. John, Type-2 Fuzzy Sets Made Simple. *IEEE Trans. Fuzzy Systems*, Vol. 10, No. 2, Pp. 117-127, (2002).
- [72]. H. Hagrais, Type-2 Flcs: “A New Generation Of Fuzzy Controllers”. *IEEE Computational Intelligence Magazine*, Vol. 2, No. 1, Pp. 30-43, (2007).
- [73]. D. Wu And J.M. Mendel. Enhanced Karnik–Mendel Algorithms. *IEEE Transactions On Fuzzy Systems*, 17(4) :923–934, (2009).
- [74]. J.M. Mendel, Type-2 Fuzzy Sets And Systems And Overview. *IEEE Computational Intelligence Magazine*, Vol. 2, No. 1, Pp. 20-29, (2007).
- [75]. K. Chafaa, L. Saidi, M. Ghanai, And K. Benmahammed. Indirect Adaptive Interval Type-2 Fuzzy Control For Nonlinear Systems. *International Journal Of Modelling, Identification And Control*, 2(2) :Pp. 106–119, (2007).
- [76]. R. J. M. M. Antão, “Type-2 Fuzzy Logic: Uncertain Systems’ Modeling And Control”. Universidade De Aveiro Departamento De Electrónica, Telecomunicações E Informática. (2016).
- [77]. M. Bouhental; M. Ghanai; Kh. Chafaa, “Interval-Valued Membership Function Estimation For Fuzzy Modeling”, *Fuzzy Sets And Systems*, Volume 361, Pp. 101-113, (2019)
- [78]. M. S. Chen, S.W. Wang, “Fuzzy Clustering Analysis For Optimizing Fuzzy Membership Functions”, *Fuzzy Sets Syst.* 103 (2) Pp. 239–254. (1999).
- [79]. F. Zarandi, R. Gamasae, I.B. Turksen, A Type-2 Fuzzy C-Regression Clustering Algorithm For Takagi–Sugeno System Identification And Its Application In The Steel Industry, *Inf. Sci.* 187 (15) 179–203. (2012).
- [80]. R. C. Hall, D. Seborg, “Modelling And Self-Tuning Control Of A Multivariable Ph Neutralization Process” Part I: Modelling And Multiloop Control 1989 American Control Conf. (1989).
- [81]. R. Babuška, H. Verbruggen, “Neuro-Fuzzy Methods For Nonlinear System Identification”, *Annu. Rev. Control* 27 (1) 73–85. (2003).
- [82]. M. Sugeno, T. Yasukawa, A Fuzzy-Logic-Based Approach To Qualitative Modeling, *Ieee Trans. Fuzzy Syst.* 1 (1) 7. (1993).
- [83]. W.A. Farag, V.H. Quintana, G. Lambert-Torres, “A Genetic Based Neuro-Fuzzy Approach For Modeling And Control Of Dynamical Systs”, *IEEE Trans. Neural Netw.* 9 (5) Pp. 756–767. (1998).

- [84]. S. Wang, C. Lee, Fuzzy System Modeling Using Linear Distance Rules, *Fuzzy Sets Syst.* 108 (2) 179–191. (1999)
- [85]. A. Evsukoff, A.C.S. Branco, S. Galichet, “Structure Identification And Parameter Optimization For Non-Linear Fuzzy Modeling”, *Fuzzy Sets Syst.* 132 (2) 173–188. (2002).
- [86]. A. Bagis, Fuzzy Rule Base Design Using Tabu Search Algorithm For Nonlinear System Modeling, *Isa Trans.* 47 (1) 32–44 (2008).
- [87]. K. S. Narendra, K. Parthasarathy. “Identification And Control Of Dynamical Systems Using Neural Networks”. *IEEE Trans Neural Netw.* (1990).
- [88]. R., Boukezzoula ; S. Galichet ; L. Foulloy ; “Linéarisation Entrée-Sortie Floue. Dans : Foulloy, L., Galichet, S., Titli, A., (Dir.), *Commande Floue 1 – De La Stabilisation A La Supervision. Série Systèmes Automatisés - Traité Ic2*”, Hermès Science Publications - Lavoisier, Paris, (2003).
- [89]. R. Antão; A. Mota; R. Escadas Martins; J. Tenreiro Machado “Model Predictive Control Using Type-2 Takagi-Sugeno Fuzzy Systems”. In: *Type-2 Fuzzy Logic. Nonlinear Physical Science.* Springer, Singapore. [https://doi.org/10.1007/978-981-10-4633-9\\_5](https://doi.org/10.1007/978-981-10-4633-9_5) (2017).
- [90]. A. Alasty, H. Salarieh “Identification And Control of Chaos Using Fuzzy Clustering And Sliding Mode Control In Unmodeled Affine Dynamical Systems” *J. Dyn. Sys., Meas., Control.* (2008).
- [91]. B.Zhang; C. Yung; Shin. "A Data-Driven Approach Of Takagi-Sugeno Fuzzy Control Of Unknown Nonlinear Systems" *Applied Sciences* 11, No. 1: 62. <https://doi.org/10.3390/app11010062> (2021).
- [92]. T. Yang, N. Sun And Y. Fang, "Adaptive Fuzzy Control For A Class Of MIMO Underactuated Systems With Plant Uncertainties And Actuator Deadzones: Design And Experiments," In *IEEE Transactions On Cybernetics*, Doi: 10.1109/Tcyb.3050475. (2021)
- [93]. E. Rouhani; Y. Fathi; “Robust Multi-Input Multi-Output Adaptive Fuzzy Terminal Sliding Mode Control of Deep Brain Stimulation In Parkinson’s Disease: A Simulation Study”. *Sci Rep* **11**, 21169\_(2021).
- [94]. J.-J. E. Slotine Et W. Li, “Applied Nonlinear Control”, Prentice Hall, N.J., (1991).
- [95]. H. K. Khalil, “Nonlinear Systems”, Macmillan Publishing Company, New York, (1992).

## ملخص

النمذجة الضبابية هي من المواضيع المهمة في نظرية المجموعات الضبابية و تطبيقاتها. و من اهم استعمالاتها هو التحكم التكيفي . و عليه ، في هذه الأطروحة سنهتم بالنمذجة الضبابية و التحكم في الانظمة غير الخطية ، تحديدا الطائرة رباعية المراوح . فهذا العمل المقدم يتمركز حول نقطتين هما :

- اقتراح طريقة نمذجة ضبابية بالاعتماد على المجموعات الضبابية من صنف-2 و ذلك بهدف احتواء المزيد من الخطأ و عدم دقة و تحديد النموذج الفيزيائي. و هذه الطريقة تعتمد على معطيات الادخال و الاخراج للنظام المراد نمذجته . في هذا الصدد ، النموذج يتم بناؤه أليا بناء على ثلاث خطوات :1 التعرف على البنية و 2 كشف المغلف و 3 التعرف على البارامترات المثالية.

- النموذج المقترح و الذي يتم الحصول عليه وفق بيانات الادخال و الاخراج للنظام موضع الدراسة ، يتم استعماله في المتحكم بغية الوصول بالنظام الى المسار المراد. و المتحكم المختار هو متحكم تكيفي غير مباشر ، حيث يستعمل النموذج الابتدائي و خلال مسار التحكم يتم تعديل النموذج بحيث يكون الخطأ أصغري و لتجاوز الخطأ بين النموذج و النظام الحقيقي.

## Abstract :

Fuzzy control and modeling are important topics in the theory of fuzzy sets and its application. In this work we are interested in the fuzzy modeling and the control of nonlinear system, more precisely the quadrotor. This work is focused on two points:

- A proposed method to model functions and dynamics using fuzzy type-2 systems based on input output data and clustering technique. This model is built in three steps: 1) build type-2 fuzzy membership functions of the TS model by clustering technique 2) detection of envelope 3) construction and identification of the parameters of the model.

- The obtained model is used to control quadrotor system. The chosen controller is an adaptive fuzzy sliding mode controller. the controller adapts its initial fuzzy parameters when running online to reach the minimum errors of tracking and to overcome the uncertainties between the fuzzy model and the real system.

## Résumé :

La modélisation et la commande flous sont des axes d'application très importants dans la théorie des ensembles. Dans ce travaille on s'intéresse à la modélisation floue et à la commande adaptative floue des systèmes non linéaires, le quadrirotor dans notre cas. Ce travail s'articule autour de deux points :

- Une méthode proposée pour modéliser les fonctions et les dynamiques en utilisant la logique floue de type 2 basée sur l'ensemble de données d'entrée-sortie. Ce modèle est construit en trois étapes : 1) construction des fonctions d'appartenance floues de type 2 du modèle TS par technique de clustering 2) détection de l'enveloppe 3) construction et identification des paramètres du modèle.

- Le modèle obtenu est utilisé pour contrôler le système quadrirotor. Le contrôleur choisi est un contrôleur adaptatif flou en mode glissant. Le contrôleur adapte ses paramètres flous initiaux lors de l'exécution en ligne pour atteindre le minimum d'erreurs de poursuite et pour surmonter les incertitudes entre le modèle flou et le système réel.