



المدرسة الوطنية المتعددة التقنيات  
Ecole Nationale Polytechnique

École Nationale Polytechnique  
Département d'Automatique



**Mémoire de projet de fin d'études**  
pour l'obtention du diplôme d'ingénieur d'état en Automatique

# Modélisation et commande d'un quadrirotor par l'approche multimodèle

*Réalisé par :*

OUAHID ABDELATIF AOUADI  
OUALID TEDJANI

*Encadré par :*

M. ACHOUR Hakim  
M. STIHI Omar

*Présenté et soutenu publiquement le 09/07/2023.*

## Composition du jury :

- Président	M. BOUKHETALA Djamal	Pr	ENP
- Examineur	M. BERKOUK El Madjid	Pr	ENP
- Promoteur	M. ACHOUR Hakim	MCB	ENP
- Co-promoteur	M. STIHI Omar	MAA	ENP





المدرسة الوطنية المتعددة التقنيات  
Ecole Nationale Polytechnique

École Nationale Polytechnique  
Département d'Automatique



**Mémoire de projet de fin d'études**  
pour l'obtention du diplôme d'ingénieur d'état en Automatique

# Modélisation et commande d'un quadrirotor par l'approche multimodèle

*Réalisé par :*

OUAHID ABDELATIF AOUADI  
OUALID TEDJANI

*Encadré par :*

M. ACHOUR Hakim  
M. STIHI Omar

*Présenté et soutenu publiquement le 09/07/2023.*

## Composition du jury :

- Président	M. BOUKHETALA Djamal	Pr	ENP
- Examineur	M. BERKOUK El Madjid	Pr	ENP
- Promoteur	M. ACHOUR Hakim	MCB	ENP
- Co-promoteur	M. STIHI Omar	MAA	ENP

هذه الأطروحة مخصصة لنظرية الضباب متعددة النماذج بطريقة تاكاجي سوجينو للرباعي. نحن مهتمون أولاً بنمذجة نموذج خطي تاكاجي سوجينو يستخدم لكتابة اللاخطية للنموذج الحقيقي بناء على طريقة القطاعات غير الخطية العالمية. بعد بناء التمثيل المكافئ، ننتقل إلى توليف قوانين التحكم. تم تصميم التحكم في تتبع المسار بشكل أساسي لنشر أهداف التحكم لتحقيق الاستقرار، بما في ذلك السعي وراء المراجع المطلوبة. ومن المتوخى أيضاً استمرار النموذج المرجعي من قبل قيادة التحكم في تتبع المسار لتكون قادرة على فرض سلوك ديناميكي على سلسلة التحكم في هيكل كوادكوبتر الذي سيعمل على جلب النظام إلى نقطة التشغيل المطلوبة وفقاً لمسار الحالة الذي يسلكه النموذج المرجعي. في النهاية، سيسمح لنا تمثيل تاكاجي سوجينو الخطي المكافئ لنموذج كوادكوبتر بتطبيق نهج تحكم مثالي يسمى التحكم التنبئي، وهو معروف تماماً بأدائه الخاص مقارنة بالطرق التقليدية. سيتم دراسة النهج وتطبيقه على النموذج الحقيقي القائم على نموذج تاكاجي سوجينو من أجل مقارنته بالإجراءات التي تم معالجتها سابقاً، من أجل التمكن من تلخيص إمكانيات وأوجه القصور في كل نهج للحالة طائرة رباعية.

---

**كلمات مفتاحية :** النمذجة والتحكم في كوادكوبتر من خلال النهج متعدد النماذج، الأمر التنبئي على أساس النموذج، نماذج تاكاجي-سوجينو، عدم المساواة في المصفوفة الخطية، طائرة رباعية.

---

## Abstract

This present dissertation is dedicated to multimodal fuzzy theory using the Takagi-Sugeno method for the quadrotor. Our primary focus lies in the modeling of a TS linear model, which aims to linearize the nonlinearities present in the real model by relying on the global nonlinear sector method. After constructing the equivalent representation, we move on to the synthesis of control laws. The PDC control is primarily studied to achieve control objectives for stabilization, particularly for tracking the desired references. The tracking of a reference model through PDC control is also considered to impose a dynamic behavior on the regulation chain within the quadrotor structure, which will bring the system to a desired operating point according to the state trajectory followed by the reference model. Finally, the linear TS representation equivalent to the quadrotor model will allow us to apply an optimal control approach called "Predictive Control," which is well-known for its specific performance compared to classical methods. The approach will be studied and applied to the real model based on the TS model, in order to compare it with the previously discussed procedures and summarize the potential and shortcomings of each approach for the quadrotor case. .

---

**Keywords :** Multimodel Approach, Model (Based) Predictive Control (MPC), Takagi-Sugeno fuzzy models, linear matrix inequality (LMI), quadcopter.

---

## Résumé

Le présent mémoire est dédié à la théorie floue multimodèles par la méthode de Takagi-Sugeno pour le quadrirotor. Nous nous intéressons dans un premier lieu à la modélisation d'un modèle linéaire TS servant à linéariser les non-linéarités que comporte le modèle réel en s'appuyant sur la méthode des secteurs non linéaires globaux. Après avoir construit la représentation équivalente, nous passons à la synthèse des lois de commande. La commande PDC est principalement étudiée afin de déployer les objectifs de commande pour la stabilisation, notamment la poursuite des références souhaitées. La poursuite d'un modèle de référence par la commande PDC est également envisagée pour pouvoir imposer un comportement dynamique sur la chaîne de régulation dans la structure du quadrirotor, qui va servir à amener le système à un point de fonctionnement désiré selon la trajectoire d'état parcourue par le modèle de référence. À la fin, la représentation TS linéaire équivalente au modèle du quadrirotor nous permettra d'appliquer une approche de commande optimale intitulée Commande prédictive, qui est assez connue pour ses performances particulières face aux méthodes classiques. L'approche sera étudiée et appliquée sur le modèle réel en se basant sur le modèle TS afin de la comparer avec les procédures traitées précédemment, pour pouvoir récapituler le potentiel et les lacunes de chaque approche pour le cas du quadrirotor..

---

**Mots clés :** L'approche multimodèle, Commande Prédictive Basée sur le Modèle, modèles Takagi-Sugeno, inégalité matricielle linéaire (LMI), quadrirotor.

---

# Remerciements

Nous tenons à exprimer nos sincères remerciements à toutes les personnes qui ont contribué aux efforts et à la réalisation de ce mémoire.

En premier lieu, nous souhaitons exprimer notre profonde gratitude envers notre promoteur, Monsieur Hakim Achour, professeur à l'École Nationale Polytechnique. Sa direction, sa patience, sa disponibilité et ses précieux conseils ont été essentiels pour mener à bien ce travail.

Nous tenons également à remercier chaleureusement nos enseignants au sein de l'école, en particulier ceux du département d'Automatique, ainsi que ceux qui nous ont enseigné les sciences fondamentales. Leur dévouement et leurs efforts constants pour nous transmettre leur savoir ont été d'une valeur inestimable.

Ces enseignants ont été une source d'inspiration et de motivation, nous poussant à repousser nos limites et à exceller dans notre travail. Leurs connaissances approfondies et leur passion pour leur domaine nous ont permis d'acquérir les compétences nécessaires pour réaliser ce mémoire.

Enfin, nous tenons à exprimer notre reconnaissance envers toutes les autres personnes qui ont apporté leur soutien, leur encouragement et leurs conseils tout au long de ce projet. Leurs contributions, qu'elles soient grandes ou petites, ont joué un rôle significatif dans la réussite de ce mémoire.

Nos plus sincères remerciements,

# Table des matières

Liste des figures . . . . .	6
Liste des tableaux . . . . .	8
Liste des acronymes . . . . .	9
Introduction générale . . . . .	11
<b>1 Modélisation floue T-S du quadrirotor . . . . .</b>	<b>13</b>
1.1 Introduction . . . . .	14
1.2 Description générale du quadrirotor . . . . .	14
1.3 Les mouvements du quadrirotor . . . . .	14
1.3.1 Mouvement vertical . . . . .	15
1.3.2 Mouvement de roulis . . . . .	16
1.3.3 Mouvement de tangage . . . . .	17
1.3.4 Mouvement de lacet . . . . .	17
1.3.5 Mouvements de translation . . . . .	18
1.4 Modèle dynamique du quadrirotor . . . . .	18
1.4.1 Angles d'Euler . . . . .	19
1.4.2 Vitesses angulaires . . . . .	20
1.4.3 Vitesses linéaires . . . . .	21
1.4.4 Effets physique agissants sur le quadrirotor . . . . .	21
1.4.5 Développement du Modèle mathématique selon Newton-Euler . . . . .	23
1.4.6 La représentation d'état du système . . . . .	26
1.4.7 La dynamique des rotors . . . . .	27
1.4.8 Les valeurs des paramètres du quadrirotor . . . . .	28
1.5 Stratégie de commande du quadri-rotor . . . . .	29
1.6 La commandabilité et l'observabilité d'un système non linéaire affine en commande . . . . .	29
1.6.1 Dérivée de lie . . . . .	29
1.6.2 Crochet de lie . . . . .	30
1.6.3 Théorème de commandabilité et d'observabilité des systèmes non linéaires . . . . .	30
1.6.4 Le cas du quadri-rotor . . . . .	31
1.6.4.1 Conclure . . . . .	34
1.7 Modélisation floue de type T-S . . . . .	34
1.7.1 Introduction . . . . .	34
1.7.2 Identification . . . . .	35
1.7.3 Linéarisation . . . . .	35
1.7.4 Méthodes des secteurs non linéaires . . . . .	35

1.8	Modélisation TS pour le quadri-rotor . . . . .	37
1.8.1	Validation du modèle TS . . . . .	40
1.9	Conclusion . . . . .	46
<b>2</b>	<b>Commande par poursuite de trajectoire . . . . .</b>	<b>47</b>
2.1	Introduction . . . . .	48
2.2	Stabilisation des états . . . . .	48
2.2.1	Application la stabilisation des angles du quadri-rotor . . . . .	51
2.3	Poursuite de trajectoire . . . . .	54
2.3.1	Généralisation de la stabilisation . . . . .	54
2.3.2	Stabilisation de l'erreur de poursuite . . . . .	55
2.4	Poursuite de trajectoire du quadri-rotor . . . . .	57
2.4.1	Première approche . . . . .	57
2.4.2	Deuxième approche . . . . .	64
2.4.3	Comparaison avec la commande par mode glissant . . . . .	72
2.5	Conclusion . . . . .	74
<b>3</b>	<b>Commande par poursuite d'un modèle de référence . . . . .</b>	<b>75</b>
3.1	Introduction . . . . .	76
3.2	Approche . . . . .	76
3.3	Implémentation de la loi de commande . . . . .	78
3.3.1	Sous-système des angles . . . . .	78
3.3.2	Sous-système de position . . . . .	83
3.3.3	Poursuite par modèle de référence en position (système global en cascade) . . . . .	84
3.4	Comparaison avec l'approche de poursuite de trajectoire . . . . .	88
3.5	Conclusion . . . . .	89
<b>4</b>	<b>Commande prédictive floue optimale . . . . .</b>	<b>90</b>
4.1	Introduction . . . . .	91
4.2	Elaboration de l'espace d'état augmenté . . . . .	91
4.3	Prédiction des états futurs . . . . .	93
4.4	Optimisation dynamique dans l'horizon de prédiction . . . . .	95
4.5	Stabilité du système en boucle fermée . . . . .	97
4.6	Commande prédictive floue T-S du quadri-rotor . . . . .	98
4.7	Conclusion . . . . .	103
<b>5</b>	<b>Implémentation pratique . . . . .</b>	<b>104</b>
5.1	Introduction . . . . .	105
5.2	Structure générale du quadri-rotor . . . . .	105
5.2.1	Moteurs A2212 . . . . .	105
5.2.2	Les variateurs de vitesse ESC (Electronic Speed Controller) . . . . .	106
5.2.3	Le capteur MPU6050 . . . . .	107
5.2.4	La carte de commande . . . . .	108
5.3	Synthèse de commande . . . . .	108
5.4	Implémentation . . . . .	110
5.5	Conclusion . . . . .	111
	<b>Conclusion générale . . . . .</b>	<b>113</b>

<b>Bibliographie</b> . . . . .	<b>114</b>
<b>Annexes</b> . . . . .	<b>117</b>
<b>A Commande par « Sliding Mode »</b> . . . . .	<b>118</b>

# Liste des figures

1.1	Structure générale d'un quadrirotor . . . . .	15
1.2	Illustration du mouvement vertical . . . . .	16
1.3	Illustration du mouvement de roulis . . . . .	16
1.4	Illustration du mouvement de tangage . . . . .	17
1.5	Illustration du mouvement de lacet . . . . .	18
1.6	Illustration du mouvement de translation . . . . .	18
1.7	Géométrie du quadrirotor . . . . .	19
1.8	Structure d'asservissement en cascade du quadri-rotor . . . . .	29
1.9	Secteur non linéaire global . . . . .	35
1.10	Architecture du multi-modèle Takagi Sugeno . . . . .	37
1.11	La sortie du système réel et TS . . . . .	41
1.12	Les erreurs . . . . .	41
1.13	La sortie du système réel et TS . . . . .	44
1.14	Les erreurs . . . . .	45
2.1	Régulateur flou par approche PDC . . . . .	49
2.2	La sortie du système réel et TS . . . . .	52
2.3	Les commandes U1, U2, U3 et U4 . . . . .	53
2.4	Les vitesses des moteurs . . . . .	53
2.5	Les erreurs . . . . .	54
2.6	La sortie du système réel et TS . . . . .	58
2.7	Les commandes U1, U2, U3 et U4 . . . . .	59
2.8	Les vitesses des moteurs . . . . .	59
2.9	Les angles . . . . .	60
2.10	Les erreurs . . . . .	60
2.11	La sortie du système réel et TS . . . . .	61
2.12	Les commandes U1, U2, U3 et U4 . . . . .	62
2.13	Les vitesses des moteurs . . . . .	62
2.14	Les angles . . . . .	63
2.15	Les erreurs . . . . .	63
2.16	La sortie du système réel et TS . . . . .	66
2.17	Les commandes U1, U2, U3 et U4 . . . . .	67
2.18	Les vitesses des moteurs . . . . .	67
2.19	Les angles . . . . .	68
2.20	Les erreurs . . . . .	68
2.21	La sortie du système réel et TS . . . . .	69
2.22	Les commandes U1, U2, U3 et U4 . . . . .	70
2.23	Les vitesses des moteurs . . . . .	70
2.24	Les angles . . . . .	71
2.25	Les erreurs . . . . .	71

2.26	La sortie du système réel et TS . . . . .	72
2.27	Les commandes U1, U2, U3 et U4 . . . . .	73
2.28	Les vitesses des moteurs . . . . .	73
3.1	La sortie du système réel et TS . . . . .	81
3.2	Les commandes U1, U2, U3 et U4 . . . . .	81
3.3	Les vitesses des moteurs . . . . .	82
3.4	Erreur Modèle Référence . . . . .	82
3.5	Les erreurs . . . . .	83
3.6	La sortie du système réel et TS . . . . .	85
3.7	Les commandes U1, U2, U3 et U4 . . . . .	86
3.8	Les vitesses des moteurs . . . . .	86
3.9	Les erreurs . . . . .	87
3.10	Erreur Modele de Référence . . . . .	87
3.11	La sortie du système réel et TS . . . . .	88
4.1	Structure d'asservissement par commande TS-MPC linéaire . . . . .	98
4.2	La sortie du système réel et TS . . . . .	99
4.3	Les commandes U1, U2, U3 et U4 . . . . .	100
4.4	Les vitesses . . . . .	100
4.5	Les angles . . . . .	101
4.6	Erreur . . . . .	101
4.7	Les erreurs du Pousuite . . . . .	102
4.8	Les animations en 3d . . . . .	103
5.1	Aperçu de la plateforme des essais . . . . .	105
5.2	Le moteur A2212 . . . . .	106
5.3	Stratégie de commande des ESC [5] . . . . .	106
5.4	Aperçu du ESC . . . . .	107
5.5	Aperçu du capteur d'inclinaison . . . . .	107
5.6	Aperçu de la carte de commande . . . . .	108
5.7	les angles . . . . .	110
5.8	Système en fonctionnement . . . . .	111
A.1	Attractivité de la surface de glissement vers l'origine . . . . .	122
A.2	Les sortie y1, y2 et y3 . . . . .	125
A.3	Les signaux de commandes . . . . .	126
A.4	Les vitesses des moteurs . . . . .	126
A.5	Les sortie y1, y2 et y3 . . . . .	129
A.6	Les sortie y1, y2 et y3 . . . . .	130
A.7	Les signaux de commandes . . . . .	131
A.8	Les vitesses des moteur . . . . .	131

# Liste des tableaux

1.1	Paramètres du modèle du quadrirotor utilisé [6]-[20]. . . . .	28
5.1	Paramètres du modèle du quadrirotor réel . . . . .	109

# Liste des acronymes

**PDC**      *Parallel Distributed Compensation*

**TS**      *Takagi Sugeno*

**MPC**      *Model Predictive Control*

**LMI**      *Linear Matrix Inequality*

**ODE**      *Ordinary Differential Equation*

**PWM**      *Pulse-Width Modulation*

# Introduction générale

Un système est l'ensemble des interactions physiques qui s'établissent moyennant l'évolution des grandeurs selon la conception du processus, d'un point de vue automatique le système possède des entrées et des sorties qui représentant les variables qu'on s'intéresse à manipuler afin de satisfaire l'objet du phénomène artificiel.

La commande des systèmes se base sur des équations mathématiques gouvernées par les différents travaux de modélisation et d'identification, mais parfois la représentation mathématique obtenue demeure complexe vu les non linéarités que repose sur et c'est du aux interactions intrinsèques dans la nature propre du système.

Une approche pour remédier le problème de cette complexité est l'utilisation du linéaire équivalent au système mais ca persiste valable uniquement dans le voisinage d'évolution des variables d'état autour d'un point d'équilibre, si ca sort des limites de ce voisinage les performances dynamiques ont tendance à atténuer, dans le premier chapitre l'approche floue « Takagi-Sugeno » est l'une des procédure qui traite ce cas de figure, ou plusieurs multi-modèles linéaire se construisent par la méthode des secteurs non linéaire et puis par le moyen des fonctions d'activation chaque modèle se favorise selon l'évolution des grandeurs physiques du processus et qui reflète exactement le comportement réel.

L'obtention de la représentation TS équivalente linéaire facilite l'étude des différents critères de performances du système tel que la stabilité, le temps de réponse et la précision de la poursuite, dans le chapitre 2 pour notre présent document on porte notre étude par l'utilisation de la fonction de « Lyapunov » qui est une fonction décrivant l'objet de la commande qu'on désire établir afin d'imposer une poursuite ou bien une stabilisation pour les grandeurs du système en boucle fermée, l'exploitation de cette fonction donne la possibilité de synthétiser des lois de commandes servant à ramener le système d'un état donné vers un état désiré et parmi les majeurs commandes utilisées la commande PDC sera envisagée.

On passe ensuite dans le chapitre 3 à l'utilisation des modèles de références qui sont généralement des modèles parfaits pour lesquels on désire le système suivre leur comportement dynamique.

L'évolution des états doit être parfois restreinte car les signaux ne doivent pas dépasser une certaines limites, l'approche d'optimalité est l'une des théorie en automatique qui s'intéresse à retrouver des lois de commande qui répondent à un cahier de charge décrivant des contraintes physiques sur le système comme le taux de consommation du carburant ainsi l'évolution des états intrinsèques, la commande prédictive « Model Predictive Control » est l'une des approches optimales linéaires qui sera utilisée en s'appuyant sur le modèle équivalent TS selon un cahier de charge respectant un compromis entre la performance désirée et l'économie d'énergie des actionneurs.

Enfin, des résultats de simulations sont illustrées pour chaque commande afin d'étudier l'inconvénient et l'avantage de chaque méthode et puis la faire comparer avec les procédure de commande classique linéaire ainsi non linéaire pour établir une conclusion rationnelle sur la théorie floue mutli-modèles pour le modèle du quadri-rotor.

# Chapitre 1

## Modélisation floue T-S du quadrirotor

# Modélisation dynamique du quadrirotor

## 1.1 Introduction

Pour concevoir un contrôleur de vol efficace pour un avion, il est impératif de comprendre en profondeur les mouvements de l'appareil, sa dynamique et les équations dynamiques qui le régissent. Cette compréhension est indispensable non seulement pour la conception du contrôleur, mais également pour garantir que les simulations du comportement du véhicule sont le plus proches possible de la réalité lorsque la commande est appliquée.

Le quadrirotor, en particulier, est considéré comme l'un des systèmes volants les plus complexes en raison du grand nombre d'effets physiques qui influencent sa dynamique, tels que les effets aérodynamiques, la gravité, les effets gyroscopiques, les frottements et le moment d'inertie. Cette complexité résulte principalement du fait que l'expression de ces effets varie pour chaque mode de vol. En effet, les modèles dynamiques du quadrirotor proposés sont différents en fonction des tâches planifiées et des environnements de navigation définis préalablement par l'opérateur. Ainsi, pour concevoir un contrôleur de vol efficace pour un quadrirotor, il est nécessaire de prendre en compte cette complexité et de développer des modèles dynamiques adaptés à chaque situation. Les informations sont récoltées de [1] et [2].

## 1.2 Description générale du quadrirotor

Un quadrirotor est un robot mobile aérien équipé de quatre rotors, qui lui permettent de se déplacer dans les six degrés de liberté de l'espace. Les rotors sont généralement disposés en croix, avec l'électronique de contrôle au centre de la croix. Afin d'empêcher le quadrirotor de tourner sur son axe de lacet, il est nécessaire que deux hélices tournent dans un sens et les deux autres dans l'autre sens. Pour diriger l'appareil, chaque paire d'hélices tournant dans le même sens doit être placée aux extrémités opposées d'une branche de la croix.

Le fonctionnement d'un quadrirotor est assez particulier, car il peut se déplacer dans toutes les directions en variant la puissance des moteurs. Il peut monter et descendre, s'incliner à gauche ou à droite (roulis), s'incliner en avant ou en arrière (tangage) et même pivoter sur lui-même (lacet). Bien qu'il dispose de six degrés de liberté, qui correspondent à trois mouvements de rotation et trois mouvements de translation, le quadrirotor est un système sous-actionné, c'est-à-dire qu'il dispose de moins d'entrées que de sorties. Pourtant, malgré cette limitation, il peut être contrôlé efficacement à l'aide de seulement quatre déclencheurs.

## 1.3 Les mouvements du quadrirotor

Dans les hélicoptères conventionnels, le rotor principal produit un couple réactif qui tend à faire tourner le corps de l'appareil dans la direction opposée. Pour contrer ce couple, on utilise habituellement un rotor de queue qui génère une poussée latérale. Cependant, ce rotor, ainsi que l'alimentation électrique qui l'accompagne, ne contribuent pas à la force de poussée de l'appareil. En revanche, dans le cas d'un quadrirotor, les rotors droit et gauche tournent dans le sens des aiguilles d'une montre, tandis que les rotors avant et arrière tournent dans le sens inverse, ce qui permet de neutraliser efficacement le couple réactif indésirable et de maintenir l'appareil en vol stationnaire sans dérive.

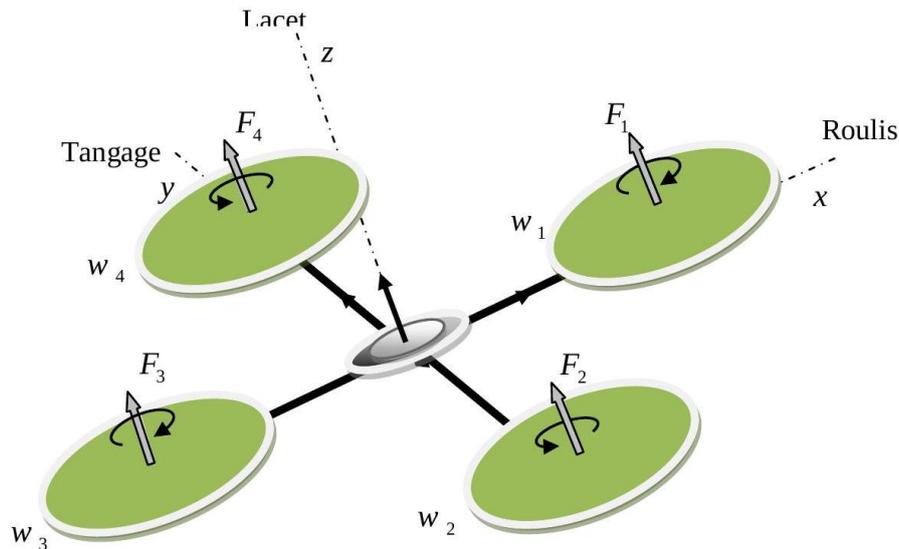


FIG. 1.1 : Structure générale d'un quadrirotor

Contrairement aux hélicoptères conventionnels, toute l'énergie dépensée pour contrer le mouvement de rotation contribue à la force de poussée du quadrirotor. Les mouvements de base du quadrirotor sont obtenus en variant la vitesse de chaque rotor, ce qui modifie la poussée produite. L'appareil s'incline alors dans la direction du rotor qui tourne plus lentement, ce qui provoque une translation le long de cet axe. Comme pour un hélicoptère, les mouvements sont couplés, ce qui signifie que le quadrirotor ne peut pas réaliser une translation sans un mouvement de roulis ou de tangage, ce qui se traduit par un changement de la vitesse d'au moins un rotor et donc un mouvement dans au moins trois degrés de liberté.

Par exemple, en augmentant la vitesse du rotor gauche, le quadrirotor s'incline vers la droite (mouvement de roulis), subit un mouvement de lacet (perturbation de l'équilibre entre les rotors tournant dans le sens des aiguilles d'une montre et ceux tournant dans le sens inverse) et une translation (le mouvement de roulis incline la structure et donc la direction de la force de poussée). Ce couplage est la raison pour laquelle nous pouvons contrôler les six degrés de liberté du quadrirotor avec seulement quatre commandes (le couple appliqué par les moteurs sur chaque rotor).

Les cinq mouvements principaux du quadrirotor sont :

- Mouvement vertical
- Mouvement de roulis
- Mouvement de tangage
- Mouvement de lacet
- Translations horizontales

### 1.3.1 Mouvement vertical

Quadrirotor en position stationnaire, la force de portance créée par chaque rotor doit être égale en grandeur et dirigée uniquement le long de l'axe vertical  $z$ , de sorte que la force de portance totale compense exactement la force de gravité agissant sur l'appareil. En d'autres

termes, la somme des forces de portance doit être égale au poids du quadrirotor pour qu'il puisse planer sans accélérer dans aucune direction.

Pour produire un mouvement ascendant ou descendant, la vitesse de rotation des moteurs doit être augmentée ou diminuée de manière égale. Si la force de portance est supérieure au poids du quadrirotor, il commencera à s'élever, et s'il est inférieur, il commencera à descendre. Le contrôle de la vitesse de rotation des moteurs permet également de réguler la vitesse de montée ou de descente.

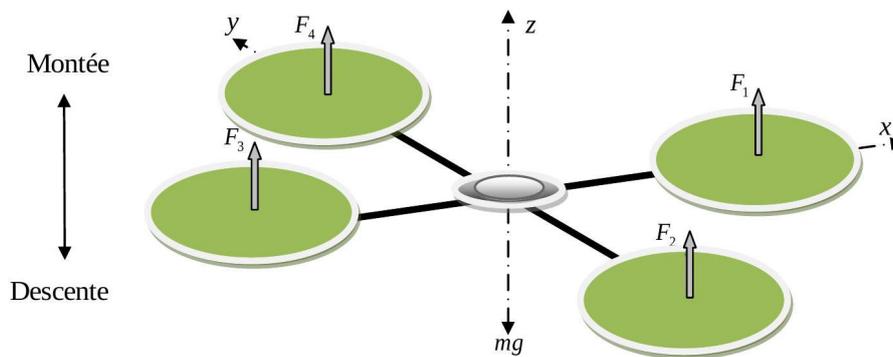


FIG. 1.2 : Illustration du mouvement vertical

### 1.3.2 Mouvement de roulis

La figure (1.3) illustre comment générer un mouvement de roulis. Pour ce faire, il faut appliquer un couple autour de l'axe  $x$ , ce qui se traduit par une différence de poussée entre le rotor 2 et le rotor 4. Ce mouvement de rotation autour de l'axe  $x$  est couplé avec un mouvement de translation selon l'axe  $y$ .

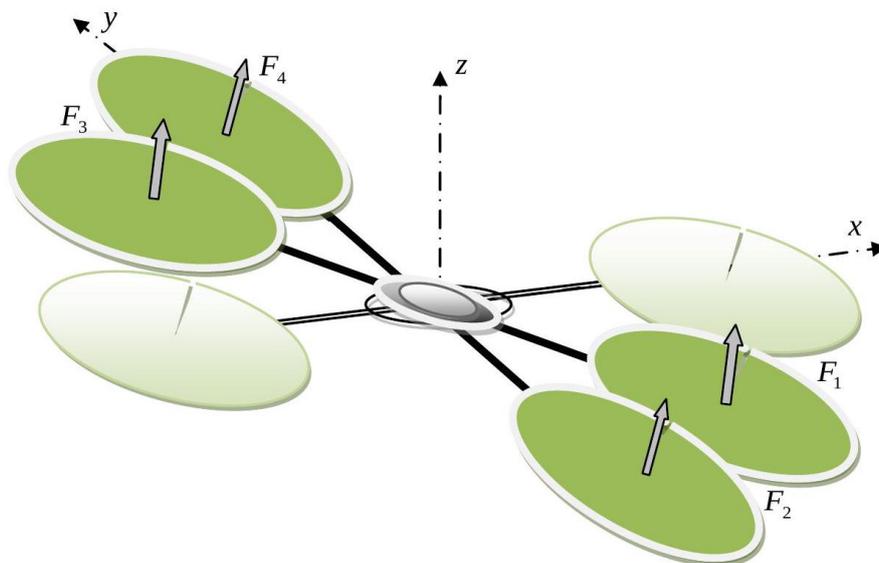


FIG. 1.3 : Illustration du mouvement de roulis

### 1.3.3 Mouvement de tangage

La figure (1.4) illustre comment un mouvement de tangage est généré. Pour ce faire, il faut appliquer un couple autour de l'axe  $y$  en produisant une différence de poussée entre le rotor 1 et le rotor 3. Ce mouvement (rotation autour de l'axe  $y$ ) est lié à un mouvement de translation le long de l'axe  $x$ .

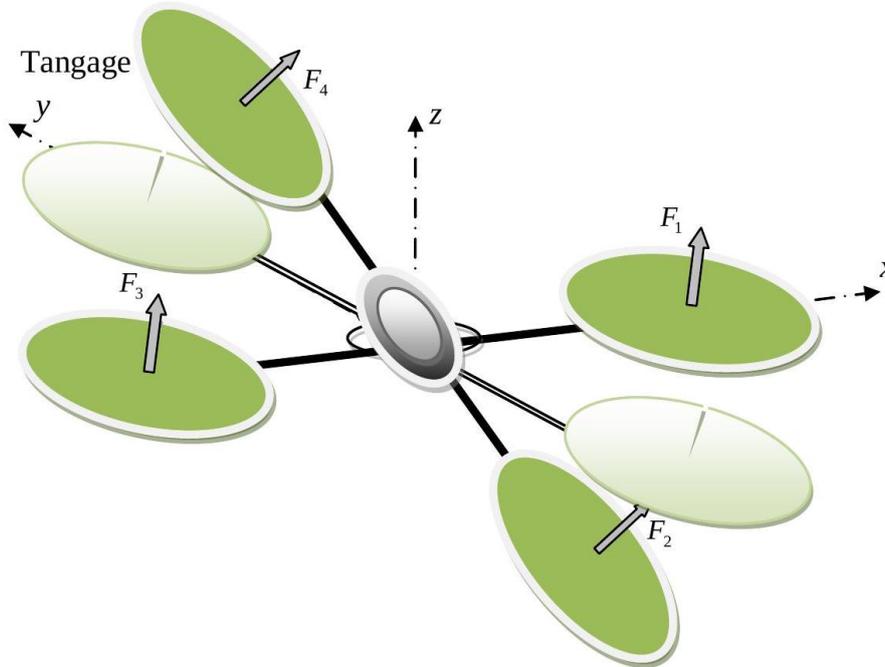


FIG. 1.4 : Illustration du mouvement de tangage

### 1.3.4 Mouvement de lacet

La figure (1.5) illustre la façon dont le mouvement de lacet est généré. Pour  $y$  parvenir, un couple doit être appliqué autour de l'axe  $z$  en produisant une différence de vitesse entre les rotors  $\{1, 3\}$  et  $\{2, 4\}$ . Contrairement aux mouvements de roulis et de tangage, le mouvement de lacet n'est pas directement le résultat de la poussée produite par les hélices, mais plutôt des couples réactifs produits par la rotation des rotors. Pendant le mouvement, la direction de la force de poussée ne change pas, mais l'augmentation de la force de portance dans une paire de rotors doit être égale à la diminution des autres paires pour s'assurer que la force de poussée totale reste constante.

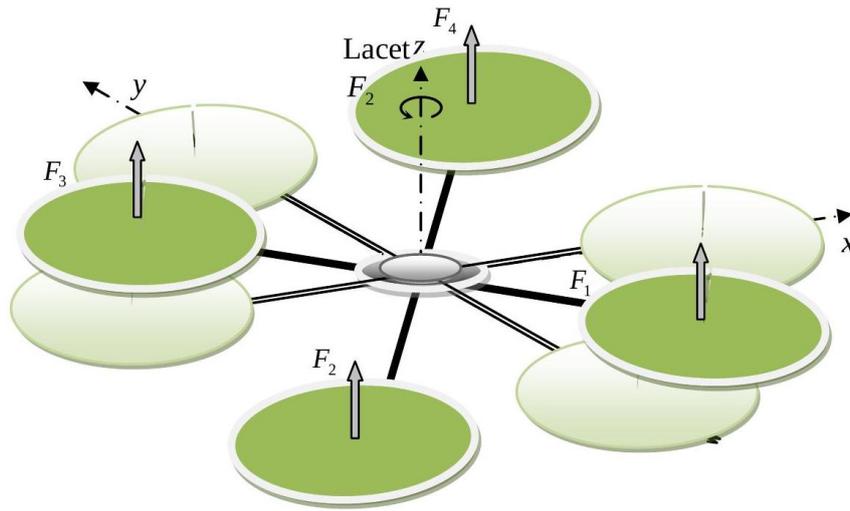


FIG. 1.5 : Illustration du mouvement de lacet

### 1.3.5 Mouvements de translation

La figure (1.6) illustre la façon dont la translation horizontale est effectuée. Pour cela, nous inclinons le corps selon le tangage ou le roulis, et augmentons la poussée produite de manière à maintenir la composante  $z$  de la force de portance égale à la force de pesanteur. Ainsi, une force est appliquée le long de l'axe  $x$  ou de l'axe  $y$ .

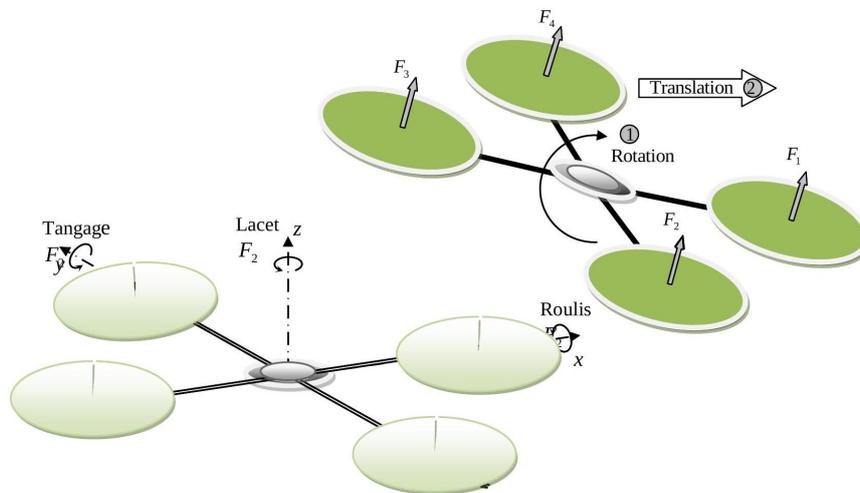


FIG. 1.6 : Illustration du mouvement de translation

## 1.4 Modèle dynamique du quadrirotor

La modélisation des robots volants est une tâche délicate en raison de la forte non-linéarité et de la couplage total de la dynamique du système. Pour mieux comprendre le modèle dynamique développé ci-dessous, nous avons fait les hypothèses de travail suivantes :

- La structure du quadrirotor est supposée rigide et symétrique, ce qui induit que la matrice d'inertie sera supposée diagonale,

- Les hélices sont supposées rigides pour pouvoir négliger l'effet de leur déformation lors de la rotation.
- Le centre de masse et l'origine du repère lié à la structure coïncident.
- Les forces de portance et de traînée sont proportionnelles aux carrés de la vitesse de rotation des rotors, ce qui est une approximation très proche du comportement aérodynamique.

Pour évaluer le modèle mathématique du quadrirotor, nous utilisons deux repères : un repère fixe lié à la Terre noté  $R^b$  et un autre mobile noté  $R^m$ . La transformation entre le repère mobile et le repère fixe est donnée par une matrice appelée matrice de transformation  $T$ , qui contient l'orientation et la position du repère mobile par rapport au repère fixe.

Nous avons choisi la convention d'axes suivante :

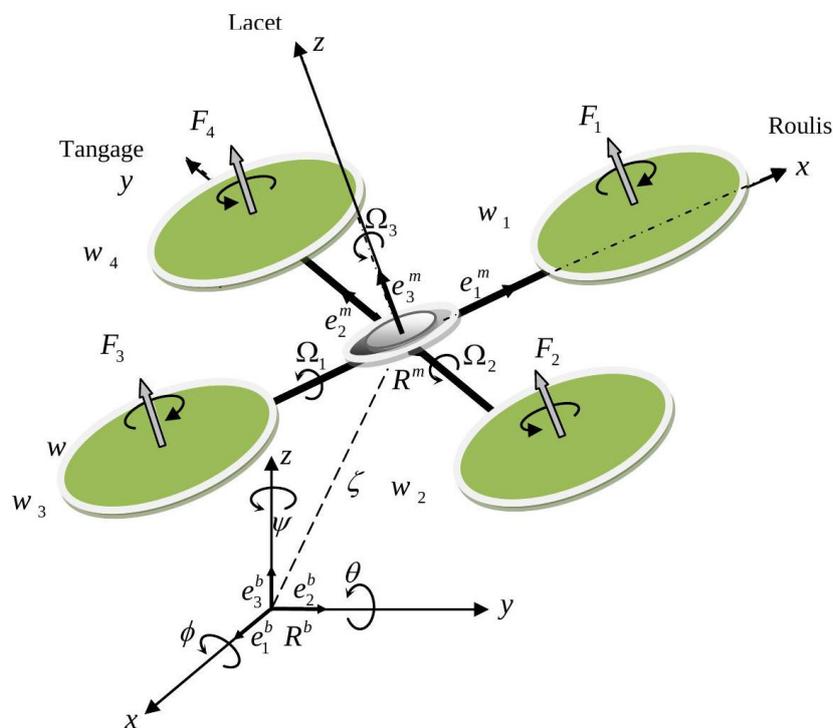


FIG. 1.7 : Géométrie du quadrirotor

$$T = \begin{bmatrix} R & \zeta \\ 0 & 1 \end{bmatrix} \quad (1.1)$$

La matrice de rotation  $R$  décrit l'orientation de l'objet mobile, et le vecteur  $\zeta = [x \ y \ z]^T$  représente sa position. Les angles d'Euler sont utilisés pour déterminer les éléments de la matrice de rotation  $R$ .

### 1.4.1 Angles d'Euler

Au début, le repère mobile coïncide avec le repère fixe. Ensuite, le repère mobile subit une rotation autour de l'axe  $x$  d'un angle de roulis  $\phi$  tel que  $-\frac{\pi}{2} < \phi < \frac{\pi}{2}$ , suivie d'une rotation

autour de l'axe  $y$  d'un angle de tangage  $\theta$  tel que  $-\frac{\pi}{2} < \theta < \frac{\pi}{2}$ , puis d'une rotation autour de l'axe  $z$  d'un angle de lacet  $\psi$  tel que  $-\pi < \psi < \pi$ . La matrice de rotation  $R$  est alors donnée par :

$$R = \text{Rot}_z(\psi) \text{Rot}_y(\theta) \text{Rot}_x(\phi) \quad (1.2)$$

$$R = \begin{bmatrix} c\psi & -s\psi & 0 \\ s\psi & c\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c\theta & 0 & s\theta \\ 0 & 1 & 0 \\ -s\theta & 0 & c\theta \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\phi & -s\phi \\ 0 & s\phi & c\phi \end{bmatrix} \quad (1.3)$$

$$R = \begin{bmatrix} c\psi c\theta & s\phi s\theta c\psi - s\psi c\phi & c\phi s\theta c\psi + s\psi s\phi \\ s\psi c\theta & s\phi s\theta s\psi + c\psi c\theta & c\phi s\theta s\psi - s\phi c\psi \\ -s\theta & s\phi c\theta & c\phi c\theta \end{bmatrix} \quad (1.4)$$

avec :  $c = \cos$ , et  $s = \sin$

## 1.4.2 Vitesses angulaires

Les vitesses de rotations  $\Omega_1, \Omega_2, \Omega_3$  dans le repère fixe sont exprimées en fonction des vitesses de rotations  $\dot{\phi}, \dot{\theta}, \dot{\psi}$  dans le repère mobile, on a :

$$\Omega = \begin{bmatrix} \Omega_1 \\ \Omega_2 \\ \Omega_3 \end{bmatrix} = \begin{bmatrix} \dot{\phi} \\ 0 \\ 0 \end{bmatrix} + \text{Rot}_x(\phi)^{-1} \begin{bmatrix} 0 \\ \dot{\theta} \\ 0 \end{bmatrix} + (\text{Rot}_y(\theta) \text{Rot}_x(\phi))^{-1} \begin{bmatrix} 0 \\ 0 \\ \dot{\psi} \end{bmatrix} \quad (1.5)$$

Effectivement, la rotation en roulis a lieu lorsque les repères sont encore confondus. En ce qui concerne le tangage, le vecteur représentant la rotation doit être exprimé dans le repère fixe, ce qui nécessite une multiplication par l'inverse de la matrice de rotation correspondante, soit  $\text{Rot}_x(\phi)^{-1}$ . De même, pour la rotation en lacet, le vecteur représentant la rotation doit être exprimé dans le repère fixe qui a déjà subi deux rotations. On obtient ainsi :

$$\begin{aligned} \Omega = \begin{bmatrix} \Omega_x \\ \Omega_y \\ \Omega_z \end{bmatrix} &= \begin{bmatrix} \dot{\phi} \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ \dot{\theta} c\phi \\ -\dot{\theta} s\phi \end{bmatrix} + \begin{bmatrix} -\dot{\psi} s\theta \\ \dot{\psi} s\phi c\theta \\ \dot{\psi} c\phi c\theta \end{bmatrix} = \begin{bmatrix} \dot{\phi} - \dot{\psi} s\theta \\ \dot{\theta} c\phi + \dot{\psi} s\phi c\theta \\ \dot{\psi} c\phi c\theta - \dot{\theta} s\phi \end{bmatrix} \\ & \Omega = \begin{bmatrix} 1 & 0 & -s\theta \\ 0 & c\phi & s\phi c\theta \\ 0 & -s\phi & c\phi c\theta \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \end{aligned} \quad (1.6)$$

Quand le quadrirotor fait des petites rotations, on peut faire les approximations suivantes :

$$c\phi = c\theta = c\psi = 1, \text{ et } s\phi = s\theta = s\psi = 0.$$

donc la vitesse angulaire sera :

$$\Omega = [ \dot{\phi} \quad \dot{\theta} \quad \dot{\psi} ]^T \quad (1.7)$$

### 1.4.3 Vitesses linéaires

Les vitesses linéaires  $v_x^b, v_y^b, v_z^b$  dans le repère fixe en fonction des vitesses linéaires  $v_x^m, v_y^m, v_z^m$  dans le repère mobile sont données par :

$$v = \begin{bmatrix} v_x^b \\ v_y^b \\ v_z^b \end{bmatrix} = R \begin{bmatrix} v_x^m \\ v_y^m \\ v_z^m \end{bmatrix} \quad (1.8)$$

### 1.4.4 Effets physique agissants sur le quadrirotor

#### Les forces

Les forces agissant sur le système sont :

Le poids du quadrirotor : il est donné par  $P = mg$ , où :  $m$  est la masse totale et  $g$  la gravité.

Les forces de poussée : qui sont des forces provoquées par la rotation des moteurs, elles sont perpendiculaires sur le plan des hélices. Ces forces sont proportionnelles au carré de la vitesse de rotation des moteurs :

$$F_i = bw_i^2 \quad (1.9)$$

avec  $i = \overline{1 : 4}$ , et  $b$  est le coefficient de portance, il dépend de la forme et le nombre des pales et la densité de l'air.

Les forces de traînée : la force de traînée est le couplage entre une force de pression et la force de frottement visqueux, dans ce cas on a deux forces de traînée agissant sur le système qu'elles sont :

- La traînée dans les hélices : elle agisse sur les pales, elle est proportionnelle à la densité de l'air, à la forme des pales et au carré de la vitesse de rotation de l'hélice, elle est donnée par la relation suivante :

$$T_h = dw^2 \quad (1.10)$$

avec  $d$  est le coefficient de drag il dépend de la fabrication de l'hélice.

- La traînée selon les axes  $(x, y, z)$  : elle est due au mouvement du corps du quadrirotor

$$F_t = K_{ft}v \quad (1.11)$$

avec :  $K_{ft}$  le coefficient de traînée de translation et  $v$  la vitesse linéaire.

### Les moments

Le quadrirotor subit plusieurs moments qui résultent des forces de poussée et de traînée, ainsi que des effets gyroscopiques.

#### Moments dus aux forces de poussée :

- En ce qui concerne la rotation autour de l'axe  $x$ , elle est causée par la différence entre les forces de portance des rotors 2 et 4, générant ainsi un moment qui peut être calculé par :

$$M_x = l(F_4 - F_2) = lb(w_4^2 - w_2^2) \quad (1.12)$$

où  $l$  représente la longueur du bras entre le rotor et le centre de gravité du quadrirotor.

- Quant à la rotation autour de l'axe  $y$ , elle est due à la différence entre les forces de portance des rotors 1 et 3, créant ainsi un moment qui peut être calculé par :

$$M_y = l(F_3 - F_1) = lb(w_3^2 - w_1^2) \quad (1.13)$$

#### Moments dus aux forces de traînée :

- Distingue deux moments qui agissent sur le quadrirotor. D'une part, la rotation autour de l'axe  $z$  est causée par un couple réactif qui résulte des couples de traînée engendrés par chaque hélice. On peut exprimer ce moment sous la forme suivante :

$$M_z = d(w_1^2 - w_2^2 + w_3^2 - w_4^2) \quad (1.14)$$

- D'autre part, il y a un moment résultant des frottements aérodynamiques qui s'applique sur le quadrirotor. On peut le définir comme suit :

$$M_a = K_{fa}\Omega^2 \quad (1.15)$$

où  $K_{fa}$  : Le coefficient des frottements aérodynamiques et  $\Omega$  est la vitesse angulaire.

## Effet gyroscopique

L'effet gyroscopique fait référence à la difficulté de modifier la position ou l'orientation du plan de rotation d'une masse en rotation. Ce phénomène est ainsi nommé en référence au gyroscope, un appareil de contrôle de mouvement utilisé en aviation (gyro pour rotation et scope pour observer).

Dans le cas du quadrirotor, il y a deux moments gyroscopiques à considérer. Le premier est le moment gyroscopique des hélices, tandis que le second est dû aux mouvements du quadrirotor lui-même.

- Moment gyroscopique des hélices : il est donné par la relation suivante :

$$M_{gh} = \sum_1^4 \Omega \wedge J_r [ 0 \quad 0 \quad (-1)^{i+1} w_i ]^T \quad (1.16)$$

où  $J_r$  est l'inertie des rotors.

- Moment gyroscopique dû aux mouvements de quadrirotor : il est donné par la relation suivante :

$$M_{gm} = \Omega \wedge J\Omega \quad (1.17)$$

où  $J$  est l'inertie du système.

### 1.4.5 Développement du Modèle mathématique selon Newton-Euler

Les équations sont exprimées sous la forme suivante en utilisant la formulation de Newton-Euler :

$$\begin{cases} \dot{\zeta} = v \\ m\ddot{\zeta} = F_f + F_t + F_g \\ \dot{R} = RS(\Omega) \\ J\dot{\Omega} = -\Omega \wedge J\Omega + M_f - M_a - M_{gh} \end{cases} \quad (1.18)$$

où  $\zeta$  : est le vecteur de position du quadrirotor

$m$  : la masse totale du quadrirotor

$\Omega$  : La vitesse angulaire exprimée dans le repère fixe

$R$  : La matrice de rotation

$\wedge$  : Le produit vectoriel

$J$  : matrice d'inertie symétrique de dimension (3x3), elle est donnée par :

$$J = \begin{bmatrix} I_x & 0 & 0 \\ 0 & I_y & 0 \\ 0 & 0 & I_z \end{bmatrix} \quad (1.19)$$

$S(\Omega)$  : est la matrice antisymétrique ; pour un vecteur de vélocité  $\Omega = [ \Omega_1 \quad \Omega_2 \quad \Omega_3 ]^T$ , elle est donnée par :

$$S(\Omega) = \begin{bmatrix} 0 & -\Omega_3 & \Omega_2 \\ \Omega_3 & 0 & -\Omega_1 \\ -\Omega_2 & \Omega_1 & 0 \end{bmatrix} \quad (1.20)$$

$F_f$  : est la force totale générée par les quatre rotors, elle est donnée par :

$$F_f = R [ 0 \quad 0 \quad \sum_{i=1}^4 F_i ]^T \quad (1.21)$$

$$F_i = bw_i^2 \quad (1.22)$$

$F_t$  : la force de traînée selon les axes  $(x, y, z)$ , elle est donnée par :

$$F_t = \begin{bmatrix} -K_{ftx} & 0 & 0 \\ 0 & -K_{fity} & 0 \\ 0 & 0 & -K_{ftz} \end{bmatrix} \dot{\zeta} \quad (1.23)$$

$K_{ftx}, K_{fity}, K_{ftz}$  : Les coefficients de traînée de translation,

$F_g$  : force de gravité, elle est donnée par :

$$F_g = \begin{bmatrix} 0 \\ 0 \\ -mg \end{bmatrix} \quad (1.24)$$

$M_f$  : moment provoqué par les forces de poussée et de traînée.

$$M_f = \begin{bmatrix} l(F_4 - F_2) \\ l(F_3 - F_1) \\ d(w_1^2 - w_2^2 + w_3^2 - w_4^2) \end{bmatrix} \quad (1.25)$$

$M_a$  : moment résultant des frottements aérodynamiques, il est donnée par :

$$M_a = \begin{bmatrix} K_{fax}\dot{\phi}^2 \\ K_{fay}\dot{\theta}^2 \\ K_{faz}\dot{\psi}^2 \end{bmatrix} \quad (1.26)$$

$K_{fax}, K_{fay}, K_{faz}$  : Les coefficients des frottements aérodynamiques.

### Equations de mouvement de translation

On a :

$$m\ddot{\zeta} = F_f + F_t + F_g \quad (1.27)$$

On remplace chaque force par sa formule, on trouve :

$$m \begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = \begin{bmatrix} c\phi c\psi s\theta + s\phi s\psi \\ c\phi s\theta s\psi - s\phi c\psi \\ c\phi c\theta \end{bmatrix} \sum_{i=1}^4 F_i - \begin{bmatrix} K_{ftx}\dot{x} \\ K_{fity}\dot{y} \\ K_{ftz}\dot{z} \end{bmatrix} - \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix} \quad (1.28)$$

On obtient alors les équations différentielles qui définissent le mouvement de translation :

$$\begin{cases} \ddot{x} = \frac{1}{m}(c\phi c\psi s\theta + s\phi s\psi) \left( \sum_{i=1}^4 F_i \right) - \frac{K_{ftx}}{m} \dot{x} \\ \ddot{y} = \frac{1}{m}(c\phi s\theta s\psi - s\phi c\psi) \left( \sum_{i=1}^4 F_i \right) - \frac{K_{fity}}{m} \dot{y} \\ \ddot{z} = \frac{1}{m}(c\phi c\theta) \left( \sum_{i=1}^4 F_i \right) - \frac{K_{ftz}}{m} \dot{z} - g \end{cases} \quad (1.29)$$

### Equations de mouvement de rotation

On a :

$$J\dot{\Omega} = -M_{gh} - M_{gh} - M_a + M_f \quad (1.30)$$

On remplace chaque moment par la formule correspondant, on trouve :

$$\begin{bmatrix} I_x & 0 & 0 \\ 0 & I_y & 0 \\ 0 & 0 & I_z \end{bmatrix} \begin{bmatrix} \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \end{bmatrix} = - \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \wedge \left( \begin{bmatrix} I_x & 0 & 0 \\ 0 & I_y & 0 \\ 0 & 0 & I_z \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \right) - \begin{bmatrix} J_r \bar{\Omega}_r \dot{\theta} \\ -J_r \bar{\Omega}_r \dot{\phi} \\ 0 \end{bmatrix} \\ - \begin{bmatrix} K_{fax} \dot{\phi}^2 \\ K_{fay} \dot{\theta}^2 \\ K_{faz} \dot{\psi}^2 \end{bmatrix} + \begin{bmatrix} lb(w_4^2 - w_2^2) \\ lb(w_3^2 - w_1^2) \\ d(w_1^2 - w_2^2 + w_3^2 - w_4^2) \end{bmatrix} \quad (1.31)$$

On obtient alors les équations différentielles définissant le mouvement de rotation :

$$\begin{cases} I_x \ddot{\phi} = -\dot{\theta} \dot{\psi} (I_z - I_y) - J_r \bar{\Omega}_r \dot{\theta} - K_{fax} \dot{\phi}^2 + lb(w_4^2 - w_2^2) \\ I_y \ddot{\theta} = \dot{\phi} \dot{\psi} (I_z - I_x) + J_r \bar{\Omega}_r \dot{\phi} - K_{fay} \dot{\theta}^2 + lb(w_3^2 - w_1^2) \\ I_z \ddot{\psi} = -\dot{\phi} \dot{\theta} (I_y - I_x) - K_{faz} \dot{\psi}^2 + d(w_1^2 - w_2^2 + w_3^2 - w_4^2) \end{cases} \quad (1.32)$$

avec :

$$\bar{\Omega}_r = w_1 - w_2 + w_3 - w_4 \quad (1.33)$$

En conséquence, le modèle dynamique complet qui régit le quadrirotor est le suivant :

$$\left\{ \begin{array}{l} \ddot{\phi} = \frac{(I_y - I_z)}{I_x} \dot{\theta} \dot{\psi} - \frac{J_r}{I_x} \bar{\Omega}_r \dot{\theta} - \frac{K_{fa_x}}{I_x} \dot{\phi}^2 + \frac{l}{I_x} u_2 \\ \ddot{\theta} = \frac{(I_z - I_x)}{I_y} \dot{\phi} \dot{\psi} + \frac{J_r}{I_y} \bar{\Omega}_r \dot{\phi} - \frac{K_{fa_y}}{I_y} \dot{\theta}^2 + \frac{l}{I_y} u_3 \\ \ddot{\psi} = \frac{(I_x - I_y)}{I_z} \dot{\theta} \dot{\phi} - \frac{K_{fa_z}}{I_z} \dot{\psi}^2 + \frac{1}{I_z} u_4 \\ \ddot{x} = -\frac{K_{ftx}}{m} \dot{x} + \frac{1}{m} u_x u_1 \\ \ddot{y} = -\frac{K_{fty}}{m} \dot{y} + \frac{1}{m} u_y u_1 \\ \ddot{z} = -\frac{K_{ftz}}{m} \dot{z} - g + \frac{\cos(\phi) \cos(\theta)}{m} u_1 \end{array} \right. \quad (1.34)$$

avec :

$$\left\{ \begin{array}{l} u_x = (c\phi c\psi s\theta + s\phi s\psi) \\ u_y = (c\phi s\theta s\psi - s\phi c\psi) \end{array} \right. \quad (1.35)$$

et :

$$\begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} = \begin{bmatrix} b & b & b & b \\ 0 & -lb & 0 & lb \\ -lb & 0 & lb & 0 \\ d & -d & d & -d \end{bmatrix} \begin{bmatrix} w_1^2 \\ w_2^2 \\ w_3^2 \\ w_4^2 \end{bmatrix} \quad (1.36)$$

a partir de (1.35), on trouve :

$$\left\{ \begin{array}{l} \phi_d = \arcsin(u_x \sin(\psi_d) - u_y \cos(\psi_d)) \\ \theta_d = \arcsin\left(\frac{(u_x \cos(\psi_d) + u_y \sin(\psi_d))}{\cos(\phi_d)}\right) \end{array} \right.$$

### 1.4.6 La représentation d'état du système

Dans le cadre de la modélisation d'un système physique, il existe de nombreuses représentations d'état possibles. Dans notre cas, nous avons choisi de représenter l'état du système à l'aide d'un vecteur d'état défini comme suit :

$$\begin{aligned} X &= [\phi \ \dot{\phi} \ \theta \ \dot{\theta} \ \psi \ \dot{\psi} \ x \ \dot{x} \ y \ \dot{y} \ z \ \dot{z}]^T \\ &= [x_1 \ x_2 \ x_3 \ x_4 \ x_5 \ x_6 \ x_7 \ x_8 \ x_9 \ x_{10} \ x_{11} \ x_{12}]^T \end{aligned}$$

On obtient la représentation d'état suivante :

$$\left\{ \begin{array}{l} \dot{x}_1 = x_2 \\ \dot{x}_2 = a_1 x_4 x_6 + a_2 x_2^2 + a_3 \bar{\Omega}_r x_4 + b_1 u_2 \\ \dot{x}_3 = x_4 \\ \dot{x}_4 = a_4 x_2 x_6 + a_5 x_4^2 + a_6 \bar{\Omega}_r x_2 + b_2 u_3 \\ \dot{x}_5 = x_6 \\ \dot{x}_6 = a_7 x_2 x_4 + a_8 x_6^2 + b_3 u_4 \\ \dot{x}_7 = x_8 \\ \dot{x}_8 = a_9 x_8 + \frac{1}{m} u_x u_1 \\ \dot{x}_9 = x_{10} \\ \dot{x}_{10} = a_{10} x_{10} + \frac{1}{m} u_y u_1 \\ \dot{x}_{11} = x_{12} \\ \dot{x}_{12} = a_{11} x_{12} + \frac{\cos(\phi) \cos(\theta)}{m} u_1 - g \end{array} \right. \quad (1.37)$$

avec :

$$\left\{ \begin{array}{l} a_1 = \frac{(I_y - I_z)}{I_x}, a_2 = -\frac{K_{fa_x}}{I_x}, a_3 = -\frac{J_r}{I_x}, a_4 = \frac{(I_z - I_x)}{I_y}, a_5 = -\frac{K_{fa_y}}{I_y}, a_6 = \frac{J_r}{I_y} \\ a_7 = \frac{(I_x - I_y)}{I_z}, a_8 = -\frac{K_{f(x)}}{I_z}, a_9 = \frac{K_{ftx}}{m}, a_{10} = -\frac{K_{f(y)}}{m}, a_{11} = -\frac{K_{ftz}}{m}, b_1 = \frac{l}{I_x}, \\ b_2 = \frac{l}{I_y}, b_3 = \frac{1}{I_z}. \end{array} \right. \quad (1.38)$$

### 1.4.7 La dynamique des rotors

Les moteurs habituellement utilisés dans les quadrirotors sont des moteurs à courant continu. Pour approximer la dynamique du rotor, on peut utiliser les équations différentielles suivantes qui sont similaires à celles d'un moteur à courant continu :

$$J_r \dot{w}_i = \tau_i - Q_i \quad , i \in \{1, 2, 3, 4\} \quad (1.39)$$

Ici,  $\tau_i$  est le couple d'entrée et  $Q_i = d w_i^2$  est le couple résistant généré par le rotor  $i$ .

Pour atteindre les objectifs de la commande d'un quadrirotor, une boucle d'asservissement en vitesse est souvent nécessaire. Pour ce faire, il est d'abord nécessaire de déterminer les vitesses désirées  $w_{d,i}$  correspondantes aux valeurs des commandes fournies par le contrôleur. Ces vitesses peuvent être calculées comme suit :

$$\varpi_d = M^{-1} U \quad (1.40)$$

Ici,  $\varpi_d = (w_{d1}^2, w_{d2}^2, w_{d3}^2, w_{d4}^2)^T$ ,  $U = (u_1, u_2, u_3, u_4)^T$  et  $M$  est une matrice non singulière obtenue à partir de (1.36). L'objectif est de synthétiser un contrôleur tel que  $w_i \rightarrow w_{d,i}$  lorsque  $t \rightarrow \infty$ , en utilisant les couples  $\tau_i$ .

On définit l'erreur de vitesse :

$$\tilde{w}_i = w_i - w_{d,i} \quad (1.41)$$

Une loi de commande est développée dans, elle est donnée par :

$$\tau_i = Q_i + J_r \dot{w}_{d,i} - k_i \tilde{w}_i \quad (1.42)$$

avec  $k_i, i \in \{1, 2, 3, 4\}$  sont des gains positifs.

On remplace la loi de commande dans (1.39), on obtient :

$$\dot{\tilde{w}}_i = -\frac{k_i}{J_r} \tilde{w}_i \quad (1.43)$$

Cette relation représente la dynamique de l'erreur et montre la convergence exponentielle de  $w_i$  vers  $w_{d,i}$  lorsque  $t \rightarrow \infty$ . Cela signifie la convergence des commandes du quadrirotor vers ces valeurs désirées, ce qui assure la stabilité du quadrirotor.

En réalité, le quadrirotor est commandé par les tensions d'alimentation de ses quatre moteurs. Pour commander ces moteurs, il est nécessaire d'obtenir la tension d'entrée de chaque moteur. Si l'inductance du moteur est petite et que les moteurs utilisés sont identiques, on peut obtenir la tension d'entrée de chaque moteur comme suit :

$$v_i = \frac{R_a}{k_m k_g} \tau_i + k_m k_g w_i \quad (1.44)$$

avec :  $R_a$  est la résistance du moteur,  $k_m$  est la constant du couple de moteur,  $k_g$  est le gain du réducteur.

### 1.4.8 Les valeurs des paramètres du quadrirotor

Paramètre	Valeur	Paramètre	Valeur
$m$	0,486 kg	$K_{fay}$	$5,567010^{-4}$ N/rad/s
$g$	9,806 m/s <sup>2</sup>	$K_{faz}$	$6,354010^{-4}$ N/rad/s
$l$	0,25 m	$K_{fx}$	$5,567010^{-4}$ N/m/s
$b$	$2,984210^{-5}$ N/rad/s	$K_{fy}$	$5,567010^{-4}$ N/m/s
$d$	$3,232010^{-7}$ N.m/rad/s	$K_{ftz}$	$6,354010^{-4}$ N/m/s
$J_r$	$2,838510^{-5}$ kg · m <sup>2</sup>	$k_i$	$20J_r$
$I_x$	$3,827810^{-3}$ kg · m <sup>2</sup>	$k_m$	$4,310^{-3}$ N · m/A
$I_y$	$3,828810^{-3}$ kg · m <sup>2</sup>	$k_g$	5,6
$I_z$	$7,656610^{-3}$ kg · m <sup>2</sup>	$R_a$	0,67Ω
$K_{fax}$	$5,567010^{-4}$ N/rad/s	$v_n$	12v

TAB. 1.1 : Paramètres du modèle du quadrirotor utilisé [6]-[20].

## 1.5 Stratégie de commande du quadri-rotor

Après avoir établi le modèle d'état du processus, il serait judicieux de bien analyser la description analytique du système on aperçoit qu'il est composé de deux sous-systèmes l'un représente la dynamique des inclinaisons de l'appareil « les angles » et l'autre est consacré pour l'évolution de la position dans l'espace 3D.

L'évolution de la position se manifeste par l'inclinaison instantanée des angles  $Ux, Uy, Uz$  ce qui est visible par le couplage en commande et qui sont toutes fonctions des angles.

Pour le présent mémoire on s'intéresse à développer des lois de commandes pour chaque sous-système individuellement et puis on met l'ensemble en cascade afin de simuler la réalité du fonctionnement, la méthode est assez connue par la commande décentralisée comme illustrée dans la figure :

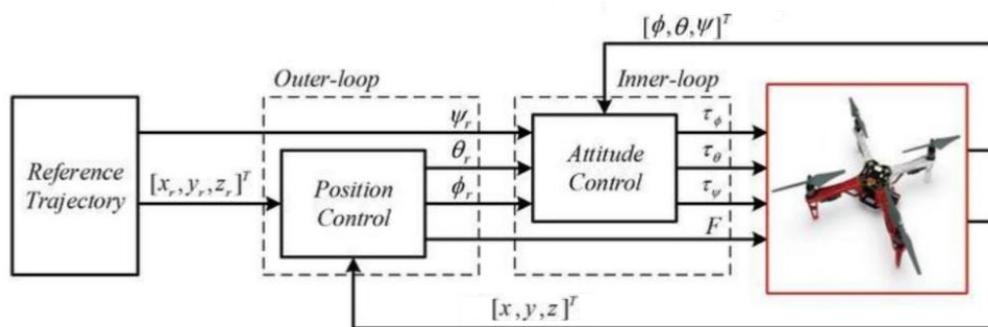


FIG. 1.8 : Structure d'asservissement en cascade du quadri-rotor

## 1.6 La commandabilité et l'observabilité d'un système non linéaire affine en commande

On introduit la notion de la dérivée de lie et le crochet lie

### 1.6.1 Dérivée de lie

Soit un champs de vecteur  $f(x)$  avec  $x \in \mathbb{R}^n$ ,  $f(x) \in \mathbb{R}^n$  et la fonction  $h(x)$  selon le champ de vecteur  $f(x)$  est donnée par :

$$L_f h(x) = \frac{\delta h}{\delta x} f \quad (1.45)$$

On peut étirer la dérivée de lie à l'ordre  $k$

$$L_f^k h(x) = \frac{\delta}{\delta x} (L_f^{k-1} h(x)) f(x) \quad \text{avec } k = 1, 2, \dots \quad (1.46)$$

$$\begin{aligned} L_f^0 h(x) &= h(x) \\ L_f^2 h(x) &= L_f L_f h(x) \end{aligned}$$

## 1.6.2 Crochet de lie

Soit  $x \in \mathbb{R}^n$  (vecteur d'état) et un autre champs de vecteur  $g(x) \in \mathbb{R}^n$ .

Le crochet de lie permet de générer un nouveau champ de vecteur

$$[f \ g] = Ad_f g(x) = \frac{\delta g}{\delta x} f(x) - \frac{\delta f}{\delta x} g(x) \quad (1.47)$$

$$[f \ g] = \begin{pmatrix} \frac{\delta g_1}{\delta x_1} & \frac{\delta g_1}{\delta x_2} & \dots & \frac{\delta g_1}{\delta x_n} \\ \vdots & \vdots & & \vdots \\ \frac{\delta g_n}{\delta x_1} & \frac{\delta g_n}{\delta x_2} & \dots & \frac{\delta g_n}{\delta x_n} \end{pmatrix} \begin{pmatrix} f_1 \\ \vdots \\ f_n \end{pmatrix} - \begin{pmatrix} \frac{\delta f_1}{\delta x_1} & \frac{\delta f_1}{\delta x_2} & \dots & \frac{\delta f_1}{\delta x_n} \\ \vdots & \vdots & & \vdots \\ \frac{\delta f_n}{\delta x_1} & \frac{\delta f_n}{\delta x_2} & \dots & \frac{\delta f_n}{\delta x_n} \end{pmatrix} \begin{pmatrix} g_1 \\ \vdots \\ g_n \end{pmatrix} \quad (1.48)$$

On peut étirer la crochet de lie a l'ordre  $k$

$$Ad_f^k g(x) = [f \ Ad_f^{k-1} g] \quad avec \quad k \geq 1 \quad Ad_f^0 g(x) = g(x) \quad (1.49)$$

$$k = 1 \implies Ad_f^1 g(x) = [f \ Ad_f^0 g] = [f \ g]$$

$$k = 2 \implies Ad_f^2 g(x) = [f \ Ad_f g] = [f \ [f \ g]]$$

$$k = 3 \implies Ad_f^3 g(x) = [f \ Ad_f^2 g] = [f \ [f \ [f \ g]]]$$

## 1.6.3 Théorème de commandabilité et d'observabilité des systèmes non linéaires

Soit le système non linéaire suivant [8] :

$$\dot{X} = f(x) + g(x)U$$

$$y = h(x)$$

avec

$$C_{om} = [g(x), Ad_f g(x), \dots, Ad_f^{n-1} g(x)] \quad (1.50)$$

et

$$O_{bs} = \frac{\delta}{\delta x} \begin{pmatrix} h(x) \\ L_f h(x) \\ \vdots \\ L_f^{n-1} h(x) \end{pmatrix} \quad (1.51)$$

**Th 1 :** Un système non linéaire est globalement faiblement commandable en  $x$  si seulement si  $Rang(C_{om}) = n$  en  $x$ .

**Th 2 :** Un système non linéaire est globalement faiblement observable en  $x$  si seulement si  $Rang(O_{bs}) = n$  en  $x$ .

## 1.6.4 Le cas du quadri-rotor

A- Etude commandabilité et d'observabilité pour le sous système des angles :

1. Etude de commandabilité pour le sous système des angles  $X_{angles} = [\phi \ \dot{\phi} \ \theta \ \dot{\theta} \ \psi \ \dot{\psi}]^T$

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = a_1 x_4 x_6 + a_2 x_2^2 + a_3 \bar{\Omega}_r x_4 + b_1 u_2 \\ \dot{x}_3 = x_4 \\ \dot{x}_4 = a_4 x_2 x_6 + a_5 x_4^2 + a_6 \bar{\Omega}_r x_2 + b_2 u_3 \\ \dot{x}_5 = x_6 \\ \dot{x}_6 = a_7 x_2 x_4 + a_8 x_6^2 + b_3 u_4 \end{cases} \quad (1.52)$$

$$f(x) = \begin{pmatrix} x_2 \\ a_1 x_4 x_6 + a_2 x_2^2 + a_3 \bar{\Omega}_r x_4 \\ x_4 \\ a_4 x_2 x_6 + a_5 x_4^2 + a_6 \bar{\Omega}_r x_2 \\ x_6 \\ a_7 x_2 x_4 + a_8 x_6^2 \end{pmatrix}, \quad g(x) = \begin{pmatrix} 0 & 0 & 0 \\ b_1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & b_2 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & b_3 \end{pmatrix} \quad (1.53)$$

$$Ad_f g(x) = \frac{\delta g}{\delta x} f(x) - \frac{\delta f}{\delta x} g(x) \quad (1.54)$$

$$\begin{aligned} Ad_f g(x) &= - \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 2a_2 x_2 & 0 & a_1 x_6 + a_3 \bar{\Omega}_r & 0 & a_1 x_4 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & a_4 x_6 + a_6 \bar{\Omega}_r & 0 & 2a_5 x_4 & 0 & a_4 x_2 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & a_7 x_4 & 0 & a_7 x_2 & 0 & 2a_8 x_6 \end{pmatrix} \cdot \begin{pmatrix} 0 & 0 & 0 \\ b_1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & b_2 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & b_3 \end{pmatrix} \\ &= - \begin{pmatrix} b_1 & 0 & 0 \\ 2b_1 a_2 x_2 & b_2(a_1 x_6 + a_3 \bar{\Omega}_r) & b_3 a_1 x_4 \\ 0 & b_2 & 0 \\ b_1(a_4 x_6 + a_6 \bar{\Omega}_r) & 2b_2 a_5 x_4 & b_3 a_4 x_2 \\ 0 & 0 & b_3 \\ b_1 a_7 x_4 & b_2 a_7 x_2 & 2b_3 a_8 x_6 \end{pmatrix} \quad (1.55) \end{aligned}$$

$$C_{om1} = [g(x), Ad_f g(x)] = \begin{pmatrix} 0 & 0 & 0 & -b_1 & 0 & 0 \\ b_1 & 0 & 0 & -2b_1 a_2 x_2 & -b_2(a_1 x_6 + a_3 \bar{\Omega}_r) & -b_3 a_1 x_4 \\ 0 & 0 & 0 & 0 & -b_2 & 0 \\ 0 & b_2 & 0 & -b_1(a_4 x_6 + a_6 \bar{\Omega}_r) & -2b_2 a_5 x_4 & -b_3 a_4 x_2 \\ 0 & 0 & 0 & 0 & 0 & -b_3 \\ 0 & 0 & b_3 & -b_1 a_7 x_4 & -b_2 a_7 x_2 & -2b_3 a_8 x_6 \end{pmatrix} \quad (1.56)$$

$$\implies \det(C_{om1}) = -b_1^2 b_2^2 b_3^2 \neq 0 \quad (1.57)$$

Donc  $\forall x \in \mathbb{R}^6 : Rang(C_{om1}) = 6 \implies$  sous-système des angles est globalement faiblement commandable.

2. Etude d'observabilité pour le sous système des angles  $X_{angles} = [\phi \ \dot{\phi} \ \theta \ \dot{\theta} \ \psi \ \dot{\psi}]^T$

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = a_1 x_4 x_6 + a_2 x_2^2 + a_3 \bar{\Omega}_r x_4 + b_1 u_2 \\ \dot{x}_3 = x_4 \\ \dot{x}_4 = a_4 x_2 x_6 + a_5 x_4^2 + a_6 \bar{\Omega}_r x_2 + b_2 u_3 \\ \dot{x}_5 = x_6 \\ \dot{x}_6 = a_7 x_2 x_4 + a_8 x_6^2 + b_3 u_4 \end{cases} \quad (1.58)$$

$$f(x) = \begin{pmatrix} x_2 \\ a_1 x_4 x_6 + a_2 x_2^2 + a_3 \bar{\Omega}_r x_4 \\ x_4 \\ a_4 x_2 x_6 + a_5 x_4^2 + a_6 \bar{\Omega}_r x_2 \\ x_6 \\ a_7 x_2 x_4 + a_8 x_6^2 \end{pmatrix}, \quad h(x) = \begin{pmatrix} x_1 \\ x_3 \\ x_5 \end{pmatrix} \quad (1.59)$$

$$L_f h(x) = \frac{\delta h}{\delta x} f \quad (1.60)$$

$$L_f h(x) = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} x_2 \\ a_1 x_4 x_6 + a_2 x_2^2 + a_3 \bar{\Omega}_r x_4 \\ x_4 \\ a_4 x_2 x_6 + a_5 x_4^2 + a_6 \bar{\Omega}_r x_2 \\ x_6 \\ a_7 x_2 x_4 + a_8 x_6^2 \end{pmatrix} = \begin{pmatrix} x_2 \\ x_4 \\ x_6 \end{pmatrix} \quad (1.61)$$

$$O_{bs1} = \frac{\delta}{\delta x} \begin{pmatrix} h(x) \\ L_f h(x) \end{pmatrix} = \frac{\delta}{\delta x} \begin{pmatrix} x_1 \\ x_3 \\ x_5 \\ x_2 \\ x_4 \\ x_6 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \quad (1.62)$$

$$\implies \det(O_{bs1}) = -1 \neq 0 \quad (1.63)$$

Donc  $\forall x \in \mathbb{R}^6 : \text{Rang}(O_{bs1}) = 6 \implies$  sous-système des angles est globalement faiblement observable.

**B-** Etude commandabilité et d'observabilité pour le sous système de position :

1. Etude de commandabilité pour le sous système de position  $X_{\text{position}} = [x \ \dot{x} \ y \ \dot{y} \ z \ \dot{z}]^T$

on prend  $u_x u_1 = \bar{u}_x$  et  $u_y u_1 = \bar{u}_y$  et  $\cos(\phi)\cos(\theta)u_1 = \bar{u}_z$

$$\begin{cases} \dot{x}_7 = x_8 \\ \dot{x}_8 = a_9 x_8 + \frac{1}{m} \bar{u}_x \\ \dot{x}_9 = x_{10} \\ \dot{x}_{10} = a_{10} x_{10} + \frac{1}{m} \bar{u}_y \\ \dot{x}_{11} = x_{12} \\ \dot{x}_{12} = a_{11} x_{12} + \frac{1}{m} \bar{u}_z - g \end{cases} \quad (1.64)$$

$$f(x) = \begin{pmatrix} x_8 \\ a_9 x_8 \\ x_{10} \\ a_{10} x_{10} \\ x_{12} \\ a_{11} x_{12} - g \end{pmatrix}, \quad g(x) = \begin{pmatrix} 0 & 0 & 0 \\ \frac{1}{m} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & \frac{1}{m} & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \frac{1}{m} \end{pmatrix}$$

$$Ad_f g(x) = \frac{\delta g}{\delta x} f(x) - \frac{\delta f}{\delta x} g(x) \quad (1.66)$$

$$Ad_f g(x) = - \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & a_9 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & a_{10} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & a_{11} \end{pmatrix} \cdot \begin{pmatrix} 0 & 0 & 0 \\ \frac{1}{m} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & \frac{1}{m} & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \frac{1}{m} \end{pmatrix} = - \begin{pmatrix} \frac{1}{m} & 0 & 0 \\ \frac{1}{m} a_9 & 0 & 0 \\ 0 & \frac{1}{m} & 0 \\ 0 & \frac{1}{m} a_{10} & 0 \\ 0 & 0 & \frac{1}{m} \\ 0 & 0 & \frac{1}{m} a_{11} \end{pmatrix} \quad (1.67)$$

$$C_{om2} = [g(x), Ad_f g(x)] = \begin{pmatrix} 0 & 0 & 0 & -\frac{1}{m} & 0 & 0 \\ \frac{1}{m} & 0 & 0 & -\frac{1}{m} a_9 & 0 & 0 \\ 0 & 0 & 0 & 0 & -\frac{1}{m} & 0 \\ 0 & \frac{1}{m} & 0 & 0 & -\frac{1}{m} a_{10} & 0 \\ 0 & 0 & 0 & 0 & 0 & -\frac{1}{m} \\ 0 & 0 & \frac{1}{m} & 0 & 0 & -\frac{1}{m} a_{11} \end{pmatrix} \quad (1.68)$$

$$\implies \det(C_{om2}) = -\frac{1}{m^6} \neq 0 \quad (1.69)$$

Donc  $\forall x \in \mathbb{R}^6 : Rang(C_{om2}) = 6 \implies$  sous-système de position est globalement faiblement commandable.

2. Etude d'observabilité pour le sous système de position  $X_{position} = [x \ \dot{x} \ y \ \dot{y} \ z \ \dot{z}]^T$

$$\begin{cases} \dot{x}_7 = x_8 \\ \dot{x}_8 = a_9 x_8 + \frac{1}{m} \bar{u}_x \\ \dot{x}_9 = x_{10} \\ \dot{x}_{10} = a_{10} x_{10} + \frac{1}{m} \bar{u}_y \\ \dot{x}_{11} = x_{12} \\ \dot{x}_{12} = a_{11} x_{12} + \frac{1}{m} \bar{u}_z - g \end{cases} \quad (1.70)$$

$$f(x) = \begin{pmatrix} x_8 \\ a_9 x_8 \\ x_{10} \\ a_{10} x_{10} \\ x_{12} \\ a_{11} x_{12} \end{pmatrix}, \quad h(x) = \begin{pmatrix} x_7 \\ x_9 \\ x_{11} \end{pmatrix} \quad (1.71)$$

$$L_f h(x) = \frac{\delta h}{\delta x} f \quad (1.72)$$

$$L_f h(x) = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} x_8 \\ a_9 x_8 \\ x_{10} \\ a_{10} x_{10} \\ x_{12} \\ a_{11} x_{12} \end{pmatrix} = \begin{pmatrix} x_8 \\ x_{10} \\ x_{12} \end{pmatrix} \quad (1.73)$$

$$O_{bs2} = \frac{\delta}{\delta x} \begin{pmatrix} h(x) \\ L_f h(x) \end{pmatrix} = \frac{\delta}{\delta x} \begin{pmatrix} x_7 \\ x_9 \\ x_{11} \\ x_8 \\ x_{10} \\ x_{12} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \quad (1.74)$$

$$\implies \det(O_{bs2}) = -1 \neq 0 \quad (1.75)$$

Donc  $\forall x \in \mathbb{R}^6 : \text{Rang}(O_{bs2}) = 6 \implies$  sous-système de position est globalement faiblement observable.

#### 1.6.4.1 Conclure

D'après A et B le Système  $X = [\phi \ \dot{\phi} \ \theta \ \dot{\theta} \ \psi \ \dot{\psi} \ x \ \dot{x} \ y \ \dot{y} \ z \ \dot{z}]^T$  est globalement faiblement observable et globalement faiblement commandable.

## 1.7 Modélisation floue de type T-S

### 1.7.1 Introduction

Dans l'annexe (5.5), des lois de commandes non linéaires ont été implémentées en se basant sur la modélisation de la dynamique du corps rigide du quadri-rotor, ce dernier est gouverné par des approches d'identifications, des approximations et surtout des négligences des non linéarités quasi-invisibles, sauf ci ca pourrait marcher pour certaines situations comme notre système qui nous a donné de très bonnes performances de poursuite, dans d'autres cas de figures il est impératif de prendre en considération toutes les aspects physiques mais la considération de telles aspects engendre une complexité et une difficulté dans la représentation de l'espace d'état réel un empêchement qui peut être surpassé dans le présent chapitre.

La commande floue a connu un large emploi dans le plan pratique grâce à ces avantages qui emploient juste des instructions logiques qualitatives dépendant à la connaissance du procédé réel sans avoir besoin d'une représentation dynamique.

La méthode de « Takagi-Sugeno » est un extrait de la théorie générale du floue, elle s'intéresse à représenter d'une manière linéaire les non linéarités que présentent l'espace d'état selon trois méthodes :

## 1.7.2 Identification

Ca revient à faire appel aux méthodes d'identification paramétrique dont laquelle on peut avoir un modèle de connaissance basé sur un registre de données entrée-sortie soit au préalable, soit pendant le fonctionnement du procédé.

## 1.7.3 Linéarisation

Elle consiste à calculer plusieurs modèle linéaire autour des points de fonctionnement bien choisi, comme ca été illustré dans [18] un seul point de fonctionnement est considéré, dans l'approche générale on envisage à chaque points une favorisation d'un modèle linéaire approprié et qui est proche du système réel , en s'appuyant sur chacun de ceux-ci on peut retrouver la réaction ce qui résulte une commande globale selon l'évolution du système réel.

## 1.7.4 Méthodes des secteurs non linéaires

La trajectoire d'état peut être considérée comme étant une fonction scalaire qui évolue dans le temps, cette évolution est supposée bornée intrinsèquement, une transformation appelée polytopique convexe de ces fonctions s'établie pour pouvoir retrouver des fonctions globales ou locales intitulées « secteurs » servant à encadrer la trajectoire d'état comme étant les extrémités comme illustré dans la figure.

La méthode a été initiée par [15]

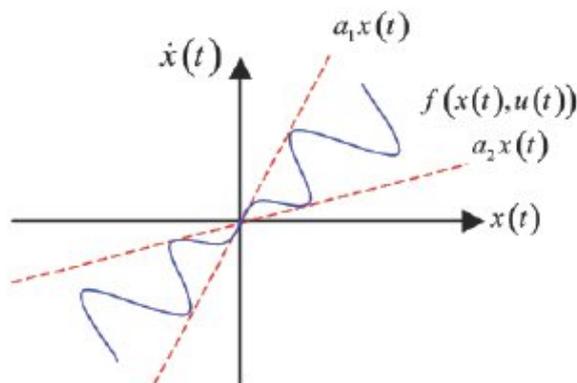


FIG. 1.9 : Secteur non linéaire global

**Lemme :**

Soit  $f(x)$  une fonction scalaire bornée dans le temps

$$f : \mathbb{R}^+ \rightarrow \mathbb{R}$$

Il existe toujours  $\alpha$  et  $\beta$  dans  $[0,1]$  tel que :

$$f = \alpha \cdot \max(f) + \beta \cdot \min(f) \tag{1.76}$$

avec

$$\alpha + \beta = 1 \quad (1.77)$$

Ce qui donne

$$\alpha = \frac{f - \min(f)}{\max(f) - \min(f)}, \quad \beta = \frac{\max(f) - f}{\max(f) - \min(f)} \quad (1.78)$$

Chaque non linéarité possède deux pondérations pour pouvoir l'approximer d'une manière exacte donc si un système donné ayant  $r$  non linéarités les multi-modèles engendrés valent  $n = 2^r$ .

Si on considère un système non linéaire affine en commande tel que :

$$\dot{X} = f(x) + g(x) \cdot U \quad (1.79)$$

Et qui a  $r$  non linéarités qu'on appelle variables de prémisse  $Z_i = T_i(X)$ .

Donc pour chaque sous modèle on a le système d'inférence flou suivant :

Si  $Z_1$  est  $F_{1i}$  et  $Z_2$  est  $F_{2i}$  et ...et  $Z_r$  est  $F_{ri}$  alors

$$\begin{cases} \dot{X} = A_i \cdot X + B_i \cdot U \\ y_i = C_i \cdot X \end{cases} \quad (1.80)$$

Avec  $i = 1, 2, \dots, r$

Et pour chaque sous-modèle on associe une pondération normalisée appelée fonction d'activation

$$\mu_i = \prod_{j=1}^r F_{ji}(Z_j(X)) \quad (1.81)$$

avec

$$\sum_{j=1}^r \mu_j = 1 \quad (1.82)$$

Ce qui nous ramène à introduire le modèle flou global équivalent au système réel.

$$\dot{X} = \sum_{i=1}^r \mu_i \cdot (A_i \cdot X + B_i \cdot U) \quad (1.83)$$

$$y = \sum_{i=1}^r \mu_i \cdot C_i \cdot X \quad (1.84)$$

Le modèle résultant étant une représentation purement linéaire nous offrant la possibilité de synthétiser des lois de commandes plus facilement, on est dans le cas où on arrive à transformer une complexité non linéaire vers un modèle de connaissance linéaire et qui évolue en plus exactement comme le processus réel ce qui montre bien le potentiel de cette théorie. Dans les sections qui viennent nous nous intéresserons à développer des lois de commande pour contrôler le système réel au moyen de la représentation TS.

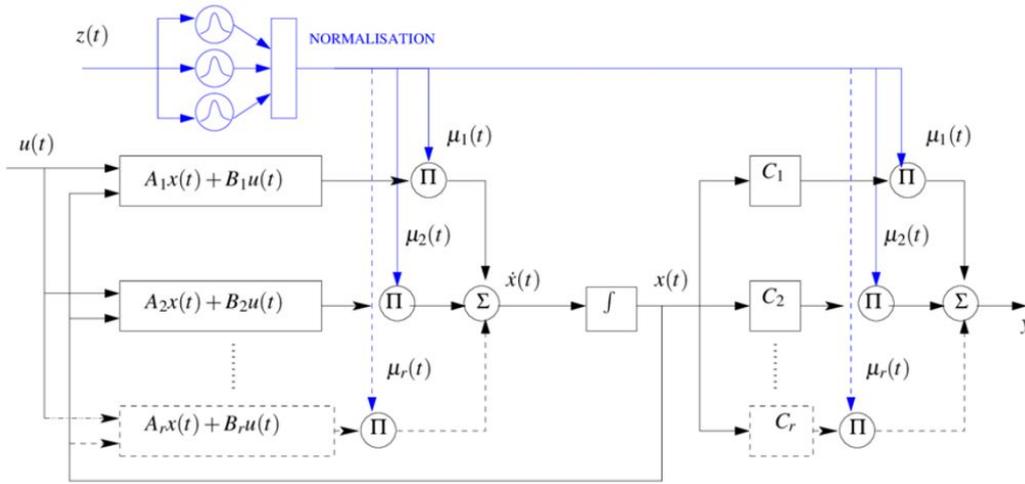


FIG. 1.10 : Architecture du multi-modèle Takagi Sugeno

## 1.8 Modélisation TS pour le quadri-rotor

Le modèle du quadri-rotor comporte certaines non linéarités et des couplages entre les signaux de commande qui rendent difficile une implémentation exacte du modèle vers des simulations pour dépasser ces contraintes on doit proposer des simplifications et des négligences qui vont servir à faciliter l'exécution tout en gardant la précision envers le processus réel.

On traite chaque sous-système à part et puis on les met en cascade pour simuler la situation réelle.

On s'intéresse à développer un modèle flou TS linéaire équivalent au système réel pour se faire, on reprend d'abord le sous-modèle des angles :

$$\begin{cases} \dot{X}_1 = X_2 \\ \dot{X}_2 = a_1 \cdot X_4 \cdot X_6 + a_2 \cdot X_2^2 + a_3 \cdot \varpi \cdot X_4 + b_1 \cdot U_2 \\ \dot{X}_3 = X_4 \\ \dot{X}_4 = a_4 \cdot X_2 \cdot X_6 + a_5 \cdot X_4^2 + a_6 \cdot \varpi \cdot X_2 + b_2 \cdot U_3 \\ \dot{X}_5 = X_6 \\ \dot{X}_6 = a_7 \cdot X_2 \cdot X_4 + a_8 \cdot X_6^2 + b_3 \cdot U_4 \end{cases} \quad (1.85)$$

$$X = \begin{pmatrix} X_1 \\ X_2 \\ X_3 \\ X_4 \\ X_5 \\ X_6 \end{pmatrix} \quad (1.86)$$

Qui peut être mis sous la forme normale des systèmes affine en commande

$$\dot{X} = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & a_2 X_2 & 0 & a_1 X_6 + a_3 \varpi & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & a_4 X_6 + a_6 \varpi & 0 & a_5 X_4 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & a_7 X_4 & 0 & 0 & 0 & a_8 X_6 \end{pmatrix} X$$

$$+ \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & b_1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & b_2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & b_3 & 0 \end{pmatrix} \bar{U} \quad (1.87)$$

Avec

$$\bar{U} = \begin{pmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \\ \varpi \end{pmatrix} \quad (1.88)$$

On remarque que le sous-modèle ayant quatre non linéarités trois non linéarités dans  $f(X)$  qui sont  $X_2, X_4, X_6$  et une non linéarité à coté de la matrice de commande  $g(X)$  qui est  $\varpi$ .

Les non linéarités des angles engendrent  $n = 2^3 = 8$  multi-modèles Afin de réduire la procédure, on néglige totalement la variable  $X_6$  et meme la variable  $\varpi$  ce qui diminue un peu la précision mais ca arrive comme meme à donner des résultats satisfaisants.

On a donc  $n = 2^2 = 4$  multi-modèles et le signal de commande  $\bar{U}$  devient  $U$  Les hypothèses considérées engendrent le modèle non linéaire suivant :

$$\dot{X} = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & a_2 X_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & a_5 X_4 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & a_7 X_4 & 0 & 0 & 0 & 0 \end{pmatrix} X + \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & b_1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & b_2 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & b_3 \end{pmatrix} U \quad (1.89)$$

$$U = \begin{pmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \end{pmatrix} \quad (1.90)$$

Le modèle est maintenant non linéaire uniquement dans la fonction  $f(X)$  On fait appel à l'approche des secteurs non linéaires en prenant les deux variables de prémisses :

$$Z_1(X) = X_2 \quad \text{et} \quad Z_2(X) = X_4 \quad (1.91)$$

Avec  $Z_k(X) \in [-a, +a], \quad a > 0$

Chaque prémisses s'écrit sous la forme :

$$Z_k(X) = \alpha a - \beta a \quad (1.92)$$

avec

$$\alpha_k = \frac{Z_k + a}{2a}, \quad \beta_k = \frac{a - Z_k}{2a} \quad (1.93)$$

Il est clair que seulement la matrice  $f(X)$  pouvant avoir les quatres multi-modèle tandis que la matrice  $g(X)$  étant purement linéaire et on la garde tel qu'elle est.

Ce qui permet d'envisager les quatre règles floues suivantes :

**Si**  $X_2 \approx \alpha_1$  **et**  $X_4 \approx \alpha_2$  **alors :**

$$A1 = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & a_2a & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & a_5a & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & a_7a & 0 & 0 & 0 & 0 \end{pmatrix}, \mu_1 = \alpha_1\alpha_2 \quad (1.94)$$

**Si**  $X_2 \approx \alpha_1$  **et**  $X_4 \approx \beta_2$  **alors :**

$$A2 = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & a_2a & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -a_5a & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & -a_7a & 0 & 0 & 0 & 0 \end{pmatrix}, \mu_2 = \alpha_1\beta_2 \quad (1.95)$$

**Si**  $X_2 \approx \beta_1$  **et**  $X_4 \approx \alpha_2$  **alors :**

$$A3 = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & -a_2a & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & a_5a & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & a_7a & 0 & 0 & 0 & 0 \end{pmatrix}, \mu_3 = \beta_1\alpha_2 \quad (1.96)$$

**Si**  $X_2 \approx \beta_1$  **et**  $X_4 \approx \beta_2$  **alors :**

$$A4 = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & -a_2a & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -a_5a & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & -a_7a & 0 & 0 & 0 & 0 \end{pmatrix}, \mu_4 = \beta_1\beta_2 \quad (1.97)$$

$$g(x) = B = B1 = B2 = B3 = B4 \quad (1.98)$$

La modélisation finale du système TS donne le modèle flou suivant

$$\dot{X}_{ts} = \sum_{i=1}^4 \mu_i(A_i X_{ts} + B_i U) \approx \dot{X} \quad (1.99)$$

$$y = C \cdot X = \begin{pmatrix} X_1 \\ X_3 \\ X_5 \end{pmatrix} \quad (1.100)$$

### 1.8.1 Validation du modèle TS

Au moyen de la ODE du Koutta Runge utilisée dans l'annexe du mode glissant [5.5], on procède selon le meme outil pour simuler les deux système en boucle ouverte.

La simulation en boucle ouverte se fait par le moyen des excitations initiales ayant des valeurs faibles  $(U_2, U_3, U_4)$ , pour éviter que les signaux divergent vers des valeurs très importantes généralement on opte à utiliser la valeur  $U_1 = m * g$  pour compenser la force gravitationnelle qui s'oppose à l'élévation du corps rigide, pour le présent cas on utilise une excitation plus importante pour que le corps prend une certaine altitude.

On prend donc Le choix de  $[-10,10]$  comme bornes d'inclinaisons depend au type du quadrirotor, en effet le modèle utilisé revient à un petit drone Dji ayant une très faible inertie donc si l'inclinaison dépasse ces valeurs le corps risque de se translater très rapidement dans l'espace 3D ce qui est indésirable.

$$U_1 = m \cdot g + 1, \quad U_2 = U_3 = U_4 = 1$$

$$a = 100, \quad dt = 0.01s$$

Avec les conditions initiales :  $X_0 = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$

Dans la réalité les angles du quadri-rotor ne doivent pas dépasser certaines limites d'inclinaisons on se limite donc pour  $X_1, X_3, X_5$  dans  $[-10,10]$  degré.

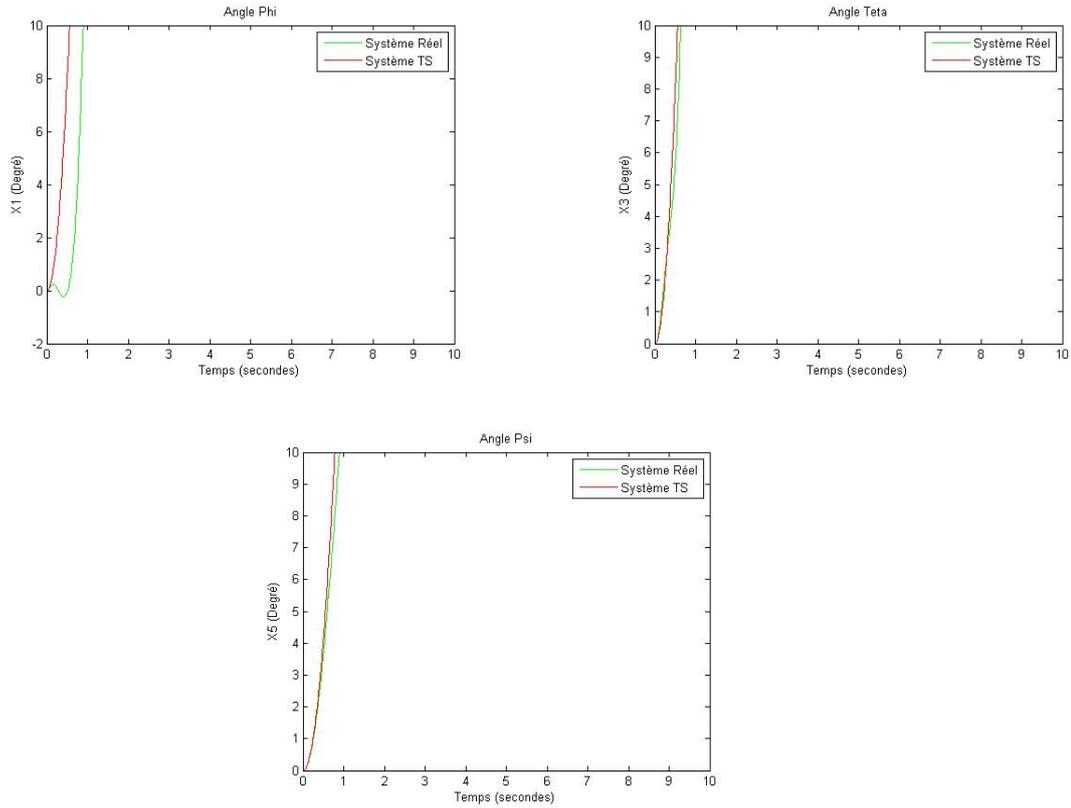


FIG. 1.11 : La sortie du système réel et TS

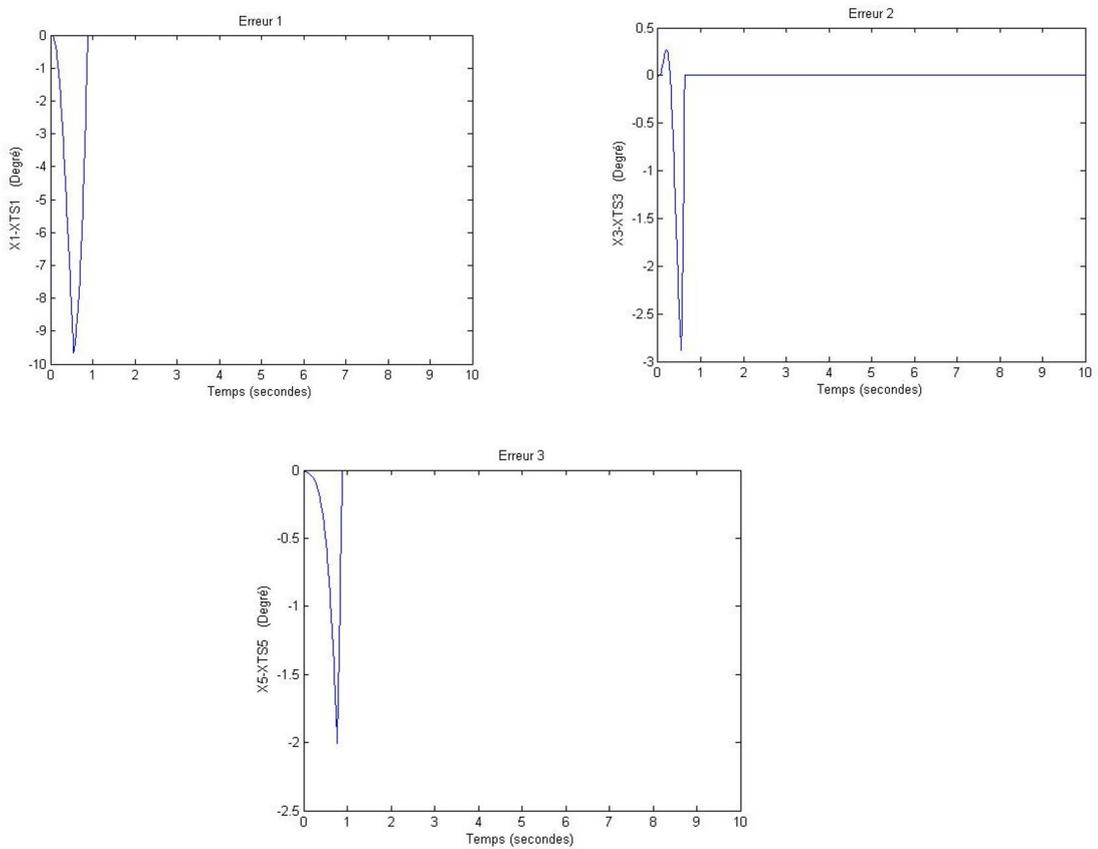


FIG. 1.12 : Les erreurs

Les deux sous-systèmes évoluent selon une dynamique proche cependant, une erreur se présente entre eux et c'est dû aux hypothèses simplificatrices qui ont été considérées la variable  $X_6$  qui a été négligée et également l'effet de la variation des vitesses des propulseurs sur les angles représenté par la variable  $\varpi$ .

Mais ça reste valide pour implémenter une régulation car le modèle TS obtenu possède des caractéristiques dynamiques très proches de celui du réel et c'est visible d'après les simulations précédentes.

On rappelle qu'une erreur s'est présentée dans la linéarisation du modèle réel via [13] ça donné des résultats un peu différents dans la boucle ouverte mais ça nous a résulté des conséquences qui coïncident entre elles dans la régulation en boucle fermée et c'est le même cas ici, dans les sections qui viennent nous allons montrer le tau de précision du TS par rapport au réel.

**Sous-système de position :**

On procède selon la même manière en reprenant le modèle non linéaire de la position (1.37) :

$$\begin{cases} \dot{X}_7 = X_8 \\ \dot{X}_8 = a_9 X_8 + \frac{1}{m} U_x U_1 \\ \dot{X}_9 = X_{10} \\ \dot{X}_{10} = a_{10} X_{10} + \frac{1}{m} U_y U_1 \\ \dot{X}_{11} = X_{12} \\ \dot{X}_{12} = a_{11} X_{12} + \frac{\cos(X_1) \cos(X_3)}{m} U_1 - g \end{cases} \quad (1.101)$$

$$X = \begin{pmatrix} X_7 \\ X_8 \\ X_9 \\ X_{10} \\ X_{11} \\ X_{12} \end{pmatrix} \quad (1.102)$$

On met le sous modèle dans la forme normale des systèmes affine en commande :

$$\dot{X} = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & a_9 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & a_{10} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & a_{11} \end{pmatrix} X + \begin{pmatrix} 0 & 0 & 0 \\ 0 & \frac{U_x}{m} & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \frac{U_y}{m} \\ 0 & 0 & 0 \\ \frac{\cos(X_1) \cdot \cos(X_3)}{m} & 0 & 0 \end{pmatrix} U_{pos} + \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ -g \end{pmatrix} \quad (1.103)$$

avec

$$U_{pos} = \begin{pmatrix} U_x \\ U_y \\ U_z \end{pmatrix}, \quad U_z = U_1 \quad (1.104)$$

On constate qu'il y a une seule non linéarité qui existe dans la matrice  $g(X)$  et un couplage entre les commandes.

On effectue les deux changements de variables pour simplifier l'écriture

$$\bar{U}_x = U_x U_1 \quad \text{et} \quad \bar{U}_y = U_y U_1 \quad (1.105)$$

On utilise ces commandes comme étant des commandes fictives pour faire disparaître le couplage avec la commande  $U_1$  et puis on déduit les deux signaux  $U_x$  et  $U_y$ .

Le nombre des règles floues vaut donc  $n = 2$  multi-modèles

On procède selon la même méthode vue dans le sous-modèle des angles :

On prend la variable de prémisse,

$$Z(X) = \cos(X_1) \cdot \cos(X_3) \quad (1.106)$$

Qui est une variable bornée dans  $\mathbb{R}$  :

$$Z : \mathbb{R} \rightarrow [-1, +1]$$

Ce qui permet d'écrire que,

$$Z(X) = \alpha \max(Z) + \beta \min(z) = \alpha - \beta \quad (1.107)$$

et donc

$$\alpha = \frac{Z + 1}{2}, \quad \beta = \frac{1 - Z}{2} \quad (1.108)$$

avec

$$\alpha + \beta = 1 \quad (1.109)$$

Si  $Z \approx \alpha$  alors :

$$B_1 = \begin{pmatrix} 0 & 0 & 0 \\ 0 & \frac{1}{m} & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \frac{1}{m} \\ 0 & 0 & 0 \\ \frac{1}{m} & 0 & 0 \end{pmatrix}, \quad \mu_1 = \alpha \quad (1.110)$$

Si  $Z \approx \beta$  alors :

$$B_2 = \begin{pmatrix} 0 & 0 & 0 \\ 0 & \frac{1}{m} & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \frac{1}{m} \\ 0 & 0 & 0 \\ -\frac{1}{m} & 0 & 0 \end{pmatrix}, \quad \mu_2 = \beta \quad (1.111)$$

et  $A = f(x) = A1 = A2$

Le système de TS équivalent au modèle de position se met sous la forme :

$$\dot{X}_{ts} = \sum_{i=1}^2 \mu_i(A_i X_{ts} + B_i U_{pos}) + \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ -g \end{pmatrix} = \dot{X} \quad (1.112)$$

$$y = C \cdot X = \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \begin{pmatrix} X_7 \\ X_9 \\ X_{11} \end{pmatrix} \quad (1.113)$$

Au contraire dans la section des angles, le système TS obtenu se comporte exactement comme le réel puisque on a rien négligé dans le modèle et c'est ce qui va être illustré prochainement dans la simulation en boucle ouverte et fermée.

On prend,

$$U_x = U_y = 1, \quad U_z = m \cdot g + 1$$

$$dt = 0.01s$$

$$\forall t > 0, \quad X_1 = X_3 = 0$$

Le système est considéré tout seul sans le couplage avec le sous-modèle des angles.

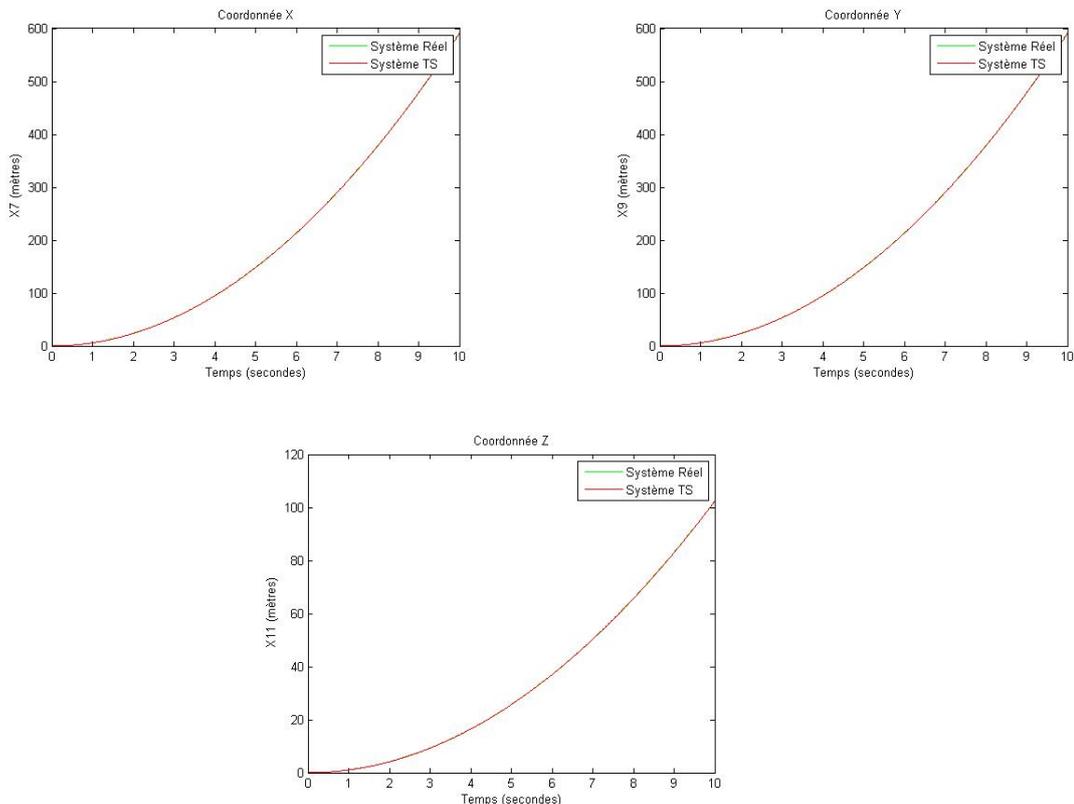


FIG. 1.13 : La sortie du système réel et TS

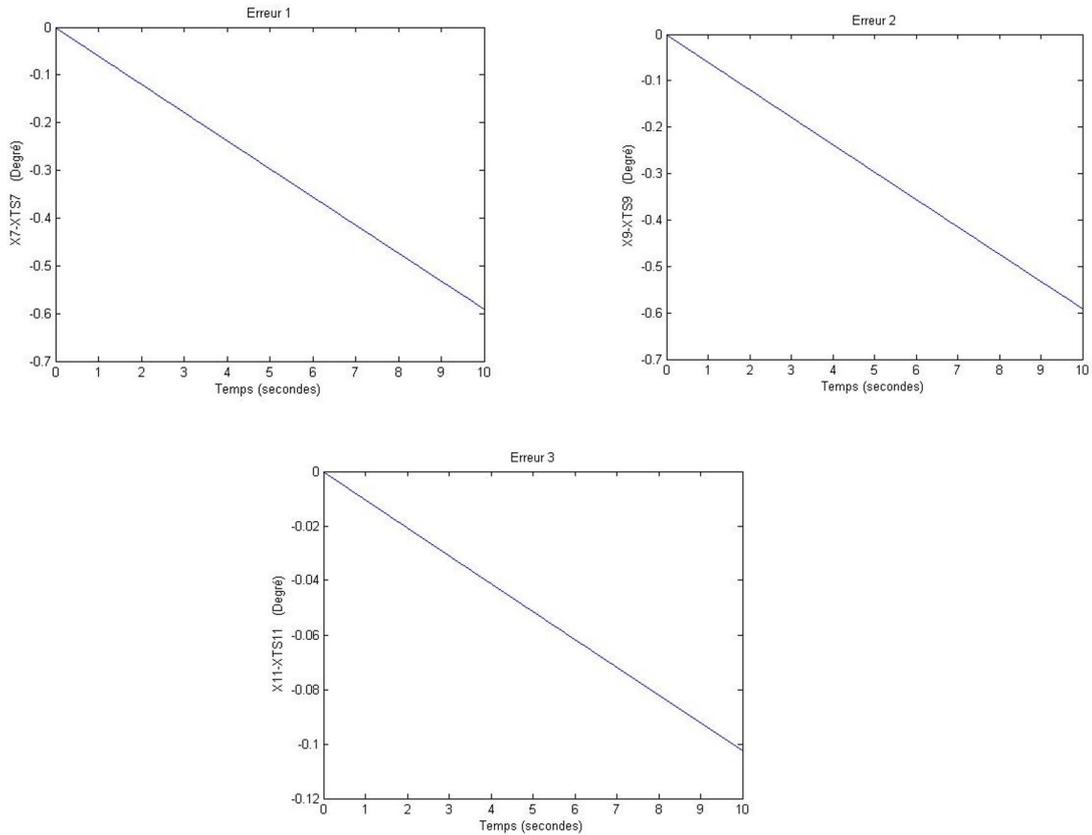


FIG. 1.14 : Les erreurs

Les signaux de sortie des deux systèmes coïncident l'un sur l'autre ce qui montre bien le degré de précision que porte l'approche de « Takagi-Sugeno » qui est bien illustré en regardant l'erreur entre eux qui évolue au voisinage de 0.01 qui représente le pas de la résolution numérique si on diminue le pas la précision sera plus importante mais pour le présent cas les résultats sont satisfaisants.

Après avoir validé en boucle ouverte les deux sous-systèmes on peut passer vers la régulation, nous allons traiter en premier lieu la stabilisation des angles du quadri-rotor.

### 1.9 Conclusion

Le présent chapitre offre au lecteur une introduction aux concepts de base des robots volants et leur principe de fonctionnement. Parmi les robots volants qui font l'objet d'investigations ces dernières années, le quadrirotor est l'un des plus étudiés. Ce système est équipé de quatre rotors, deux tournant dans un sens et les deux autres dans le sens inverse. En faisant varier les vitesses de rotation de ces rotors, le quadrirotor peut effectuer différents mouvements, tant en translation qu'en rotation.

En utilisant le formalisme de Newton-Euler, nous avons établi le modèle dynamique du quadrirotor. À partir de ce modèle, nous avons conclu que le quadrirotor est un système sous-actué, et que la complexité du modèle, la non-linéarité et l'interaction entre les états du système sont clairement visibles.

Après avoir retrouvé le modèle non linéaire, nous nous sommes intéressé par la modélisation floue au sens « Takagi-Sugeno » dont on a pu calculé les multi-modèles linéaires qui servent à introduire le modèle TS équivalent se comportant selon une trajectoire d'état quasi-exacte à celle du système réel malgré les négligences qui ont été imposées.

Dans le prochain chapitre, par le moyen du modèle TS obtenu, on s'occupe par la synthèse des lois de commandes selon l'approche PDC.

# Chapitre 2

## Commande par poursuite de trajectoire

## 2.1 Introduction

Dans un premier lieu, on s'intéresse à développer une lois de commande stabilisante qui sert à ramener la trajectoire des états vers son origine, cela peut s'effectuer par le moyen d'une fonction candidate de « Lyapounov » servant à conclure sur l'expression de la loi de commande résultante or, puisque il s'agit d'un modèle TS le calcul de la dérivée conduit à une inégalité matricielle sous forme d'une somme des multi-modèles, on opte à employé les outils de calcul des inégalités matricielles linéaire et leurs régions afin de calculer les paramètres du régulateur multi-modèle TS. On s'oriente dans un deuxième lieu par la partie poursuite des trajectoires dont on cherche à ramener la trajectoire des états vers un point de fonctionnement donné, ceci est abordé selon deux approche l'une consiste à généraliser l'aspect de la stabilisation des états qu'on vient de mentionné et la seconde repose sur la stabilisation de l'espace d'état de l'écart entre le modèle TS et le vecteur d'état contenant les consignes désirées. La représentation TS nous donne le chemin de synthétiser ces régulateur en utilisant son aspect linéaire et qui peuvent être implémentés directement dans le modèle réel puisque il s'agit du meme comportement dynamique ce qui montre bien le potentiel qui peut être illustré en faisant la comparaison avec les approches classiques tel que le mode glissant afin conclure sur les avantages que nous offre et les inconvénients qui peuvent s'opposer. [1]

## 2.2 Stabilisation des états

Dans [13], la stabilité au sens de « Lyapounov » est détaillée , et ca montre qu'une fonction candidate peut nous donner un jugement critique sur la stabilité du système réel. Pour la plus part des cas les fonctions quadratiques sont considérées :

$$V = X^T P X \quad (2.1)$$

Tel que P est une solution de l'équation de « Lyapounov » :

$$A^T P + P A < 0 \quad (2.2)$$

La stabilisation du système TS s'effectue à travers la meme manière toute en respectant la forme normale du modèle linéaire :

Etant donnée une représentation TS équivalente à un système non linéaire affine en commande

$$\dot{X} = f(x) + g(x)U = \sum_{i=1}^r \mu_i (A_i X + B_i U) \quad (2.3)$$

On s'intéresse à la stabilisation des états vers l'origine en reprenant la formule précédente

On procède par une dérivation :

$$\dot{V} = \dot{X}^T P X + X^T P \dot{X} \quad (2.4)$$

On remplace par les expressions :

$$\dot{V} = \sum_{i=1}^r \mu_i.(X^T.Ai^T + U^T.Bi^T).P.X + \sum_{i=1}^r \mu_i.(Ai.X + Bi.U) \quad (2.5)$$

$$\dot{V} = \sum_{i=1}^r \mu_i.(X^T.(Ai^T.P + P.Ai)X + U^T.Bi^T.P.X + X^T.P.Bi.U) \quad (2.6)$$

On pose

$$Gi = Ai^T.P + P.Ai \quad (2.7)$$

Et donc

$$\dot{V} = \sum_{i=1}^r \mu_i(X^T.Gi.X + X^T.P.Bi.U + (X^T.P.Bi.U)^T) \quad (2.8)$$

Le système en boucle fermée étant asymptotiquement stable dans  $\mathbb{R}^n$  si :

$$\dot{V} = \sum_{i=1}^r \mu_i(X^T.Gi.X + X^T.P.Bi.U + (X^T.P.Bi.U)^T) < 0 \quad (2.9)$$

On introduit la commande PDC sous forme d'un retour d'état [10] :

$$U = - \sum_{j=1}^r \mu_j.K_j.X \quad (2.10)$$

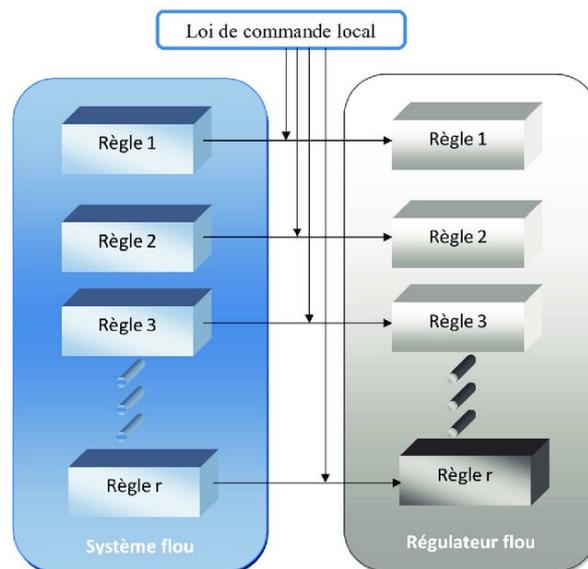


FIG. 2.1 : Régulateur flou par approche PDC

Ce qui résulte dans l'équation (2.8) :

$$\dot{V} = \sum_{i=1}^r \mu_i (X^T \cdot G_i \cdot X - X^T \cdot P \cdot B_i \cdot (\sum_{j=1}^r \mu_j \cdot K_j \cdot X) - ((\sum_{j=1}^r \mu_j \cdot X^T \cdot K_j^T) \cdot B_i^T \cdot P \cdot X)) \quad (2.11)$$

Par simplifications,

$$\dot{V} = \sum_{i=1}^r \mu_i (X^T \cdot G_i \cdot X - X^T \cdot P \cdot B_i \cdot \sum_{j=1}^r \mu_j \cdot K_j \cdot X - X^T \cdot \sum_{j=1}^r \mu_j \cdot K_j^T \cdot B_i^T \cdot P \cdot X) \quad (2.12)$$

$$\dot{V} = \sum_{i=1}^r \mu_i (X^T \cdot G_i \cdot X - X^T \cdot (\sum_{j=1}^r \mu_j \cdot (P \cdot B_i \cdot K_j + K_j^T \cdot B_i^T \cdot P)) \cdot X) \quad (2.13)$$

$$\dot{V} = \sum_{i=1}^r \mu_i \cdot (X^T \cdot (G_i - \sum_{j=1}^r \mu_j \cdot (P \cdot B_i \cdot K_j + K_j^T \cdot B_i^T \cdot P)) \cdot X) \quad (2.14)$$

$$\dot{V} = \sum_{i=1}^r \sum_{j=1}^r \mu_i \cdot \mu_j \cdot (X^T \cdot (G_i - P \cdot B_i \cdot K_j - K_j^T \cdot B_i^T \cdot P) \cdot X) \quad (2.15)$$

On explicite  $G_i = A_i^T \cdot P + P \cdot A_i$

$$\dot{V} = \sum_{i=1}^r \sum_{j=1}^r \mu_i \cdot \mu_j \cdot (X^T \cdot (A_i^T \cdot P + P \cdot A_i - P \cdot B_i \cdot K_j - K_j^T \cdot B_i^T \cdot P) \cdot X) \quad (2.16)$$

Finalement

$$\dot{V} = X^T \cdot \sum_{i=1}^r \sum_{j=1}^r \mu_i \cdot \mu_j \cdot ((A_i^T - K_j^T \cdot B_i^T) \cdot P + P \cdot (A_i - B_i \cdot K_j)) \cdot X \quad (2.17)$$

On pose  $G_{ij} = A_i - B_i \cdot K_j$  Par conséquent,

$$\dot{V} = X^T \cdot (\sum_{i=1}^r \sum_{j=1}^r \mu_i \cdot \mu_j \cdot (G_{ij}^T \cdot P + P \cdot G_{ij})) \cdot X \quad (2.18)$$

L'équation obtenue étant une équation quadratique augmentée de celle de (2.8), celle-ci représente le système TS en boucle fermée après avoir introduit la commande PDC et comme ca a été cité précédemment, les états du système se stabiliseront vers l'origine si seulement si ils présentent une stabilité asymptotique au sens de « Lyapounov » et qui sera gouvernée par la condition nécessaire suivante :

$$\sum_{i=1}^r \sum_{j=1}^r (G_{ij}^T \cdot P + P \cdot G_{ij}) < 0 \quad (2.19)$$

Le calcul des gains nécessaires pour stabiliser le système revient à résoudre l'inégalité (2.19) qui est un problème d'inégalité matricielle linéaire (LMI).

## 2.2.1 Application la stabilisation des angles du quadri-rotor

Nous nous intéressons par la stabilisation des angles du quadri-rotor vers l'origine selon des conditions initiales non nulles, par le moyen de la fonction de « Lyapounov » choisie comme celle là :

$$V = X^T.P.X \quad (2.20)$$

Qui est fonction quadratique définie positive et qui passe par l'origine.

On fait appel aux outils de résolution des LMI sur « Matlab » pour résoudre l'inégalité trouvée dans ce qui précède :

$$\dot{V} = X^T. \left( \sum_{i=1}^4 \sum_{j=1}^4 \mu_i.\mu_j.(G_{ij}^T.P + P.G_{ij}) \right).X < 0 \quad (2.21)$$

Le calcul direct des gains par LMI nous a donné des résultats inappropriés portant des harmoniques pendant l'évolution et parfois pour certaines conditions initiales ça n'arrive pas à se stabiliser, pour remédier ce problème nous avons remarqué que les gains obtenus sont très faibles ceci est du par la condition (2.21) qui est nécessaire mais pas suffisante le placement de pôles aléatoires que fait l'algorithme sur « Matlab » assure uniquement la satisfaction d'une stabilité asymptotique ce qui engendre un placement de pole ne convenant pas la nature instable du modèle afin de dépasser cet anomalie on se propose d'effectuer un placement de pole sur la région des LMI une approche qui sera bien détaillée [17], cela permet de choisir la région adéquate des pôles nous donnant plus de degrés de liberté afin de satisfaire les performances attendues par le quadri-rotor.

On exploite les trois contraintes sur la région et on choisi les paramètres de la région comme ceux-ci :

$$\alpha = -0.1 \quad \beta = 0 \quad \theta = 0 \quad (2.22)$$

On trouve les gains qui valent :

$$K1 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 1.1264 & 0.1825 & 0 & 0 & 0.0009 & 0.0001 \\ 0 & 0 & 1.1263 & 0.1826 & 0 & 0 \\ 0.0004 & 0.0001 & 0 & 0 & 0.6227 & 0.1365 \end{bmatrix} \quad (2.23)$$

$$K2 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 1.0813 & 0.1308 & 0 & 0 & -0.0006 & -0.0001 \\ 0 & 0 & 1.1997 & 0.3471 & 0 & 0 \\ -0.0003 & 0 & 0 & 0 & 0.7082 & 0.1535 \end{bmatrix} \quad (2.24)$$

$$K3 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 1.1998 & 0.3471 & 0 & 0 & -0.0008 & -0.0002 \\ 0 & 0 & 1.0813 & 0.1309 & 0 & 0 \\ -0.0005 & -0.0002 & 0 & 0 & 0.7081 & 0.1535 \end{bmatrix} \quad (2.25)$$

$$K4 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 1.2262 & 0.3314 & 0 & 0 & 0.0010 & 0.0003 \\ 0 & 0 & 1.2261 & 0.3313 & 0 & 0 \\ 0.0006 & 0.0001 & 0 & 0 & 0.6967 & 0.1512 \end{bmatrix} \quad (2.26)$$

Avec les conditions initiales :

$$U1 = m.g, \quad U2 = U3 = U4 = 0 \quad (2.27)$$

$$X0 = \begin{bmatrix} 5 \\ 0 \\ -2 \\ 0 \\ 1 \\ 0 \end{bmatrix} \quad (2.28)$$

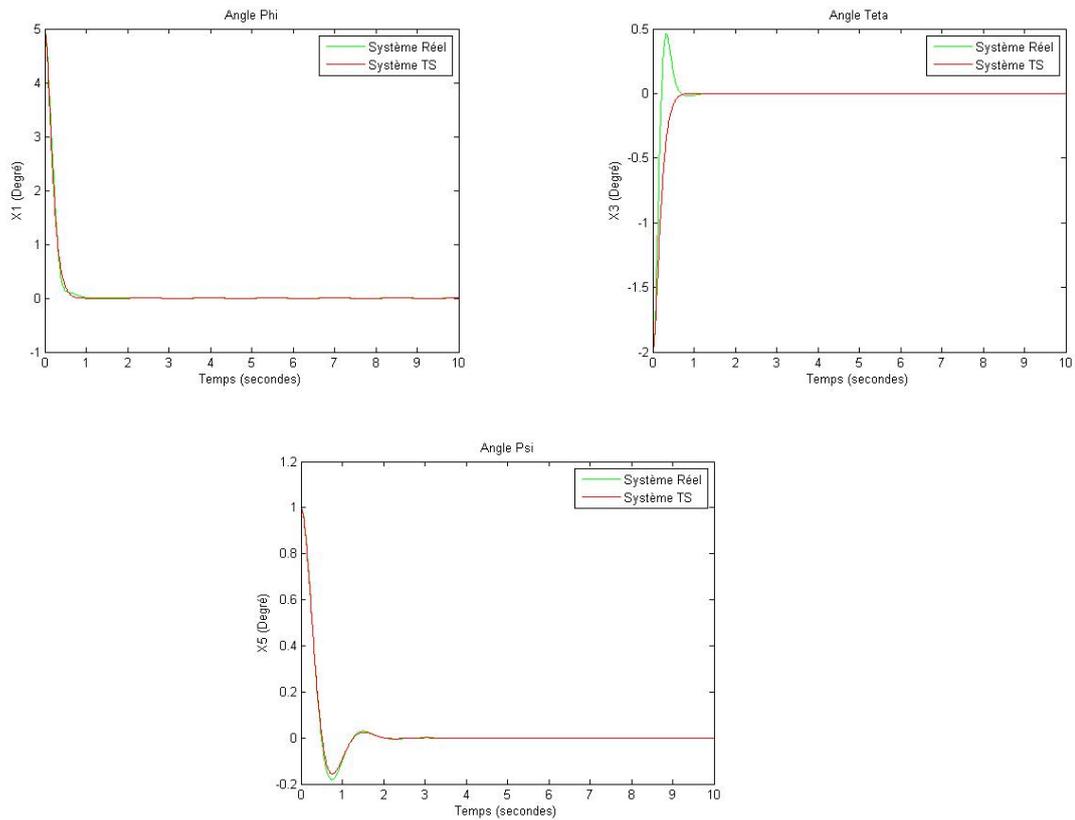


FIG. 2.2 : La sortie du système réel et TS

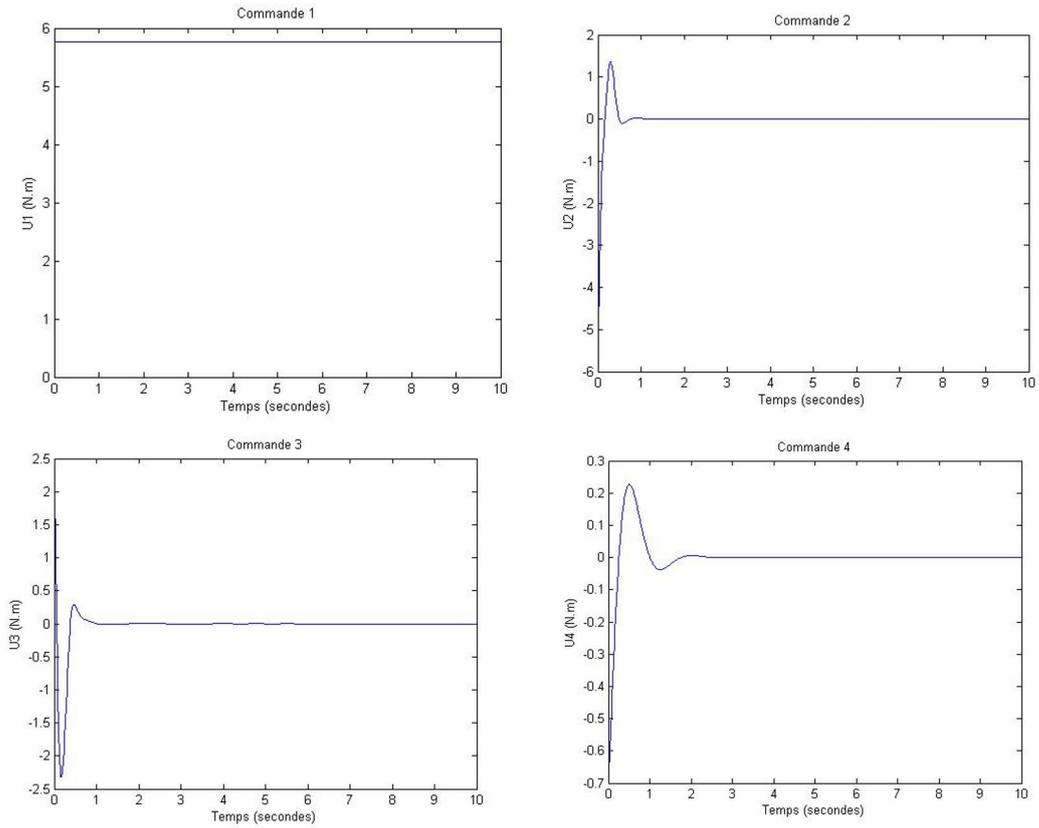


FIG. 2.3 : Les commandes U1, U2, U3 et U4

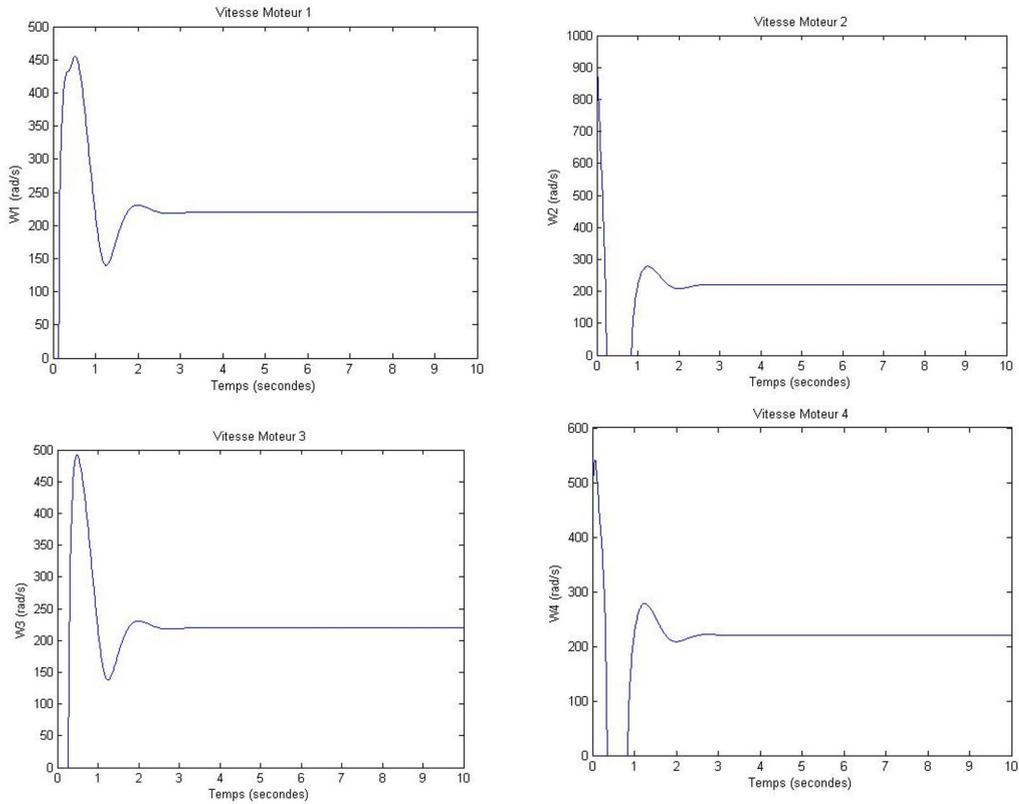


FIG. 2.4 : Les vitesses des moteurs

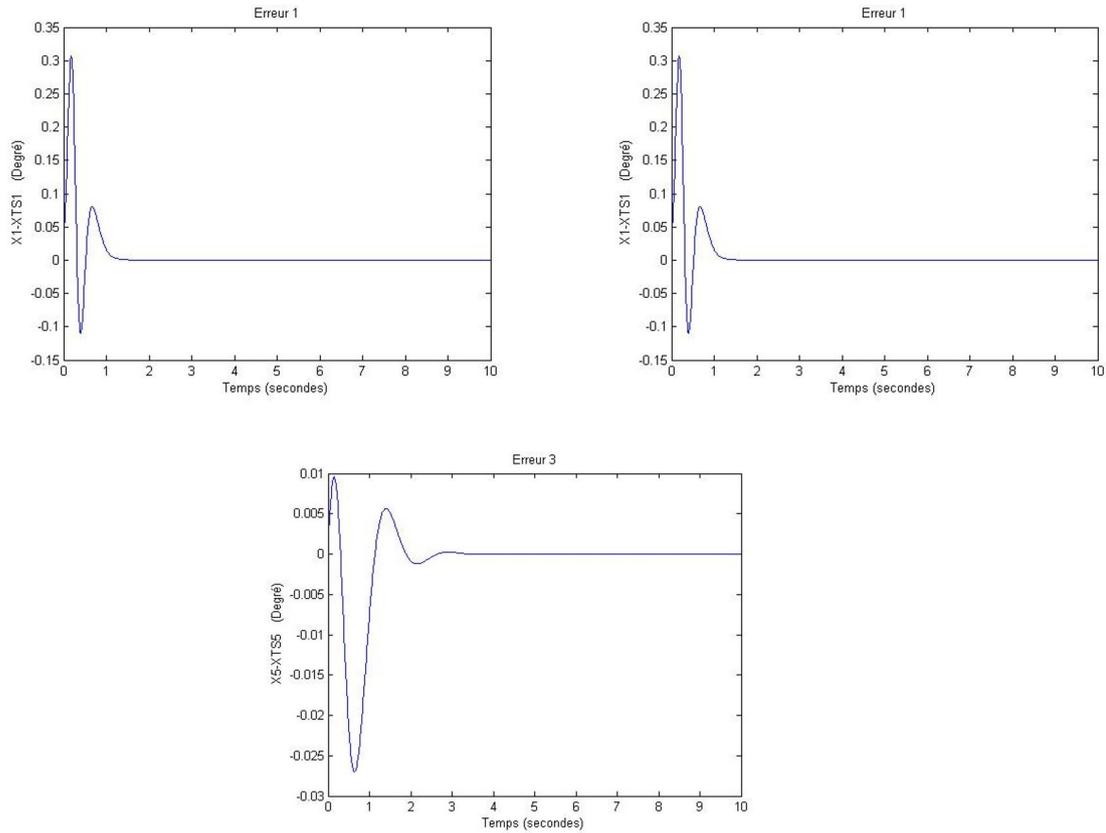


FIG. 2.5 : Les erreurs

On aperçoit une stabilisation des états pour les deux systèmes selon la meme dynamique et la meme trajectoire avec une légère erreur qui apparait dans l'évolution de la sortie  $y_2$  mais ca reste comme meme proche du système réel. Les négligences de la variable  $X_6$  et  $\varpi$  font leur impacts à travers les erreurs occurrentes. Les performances étant satisfaisantes étant donné que les angles arrivent à rejoindre leurs origines dans le voisinage d'une secondes une aptitude qui sera primordiale lorsqu'on désire faire la régulation de la boucle esclave pour la navigation en trajectoire et ca sera notre sujet pour la prochaine section.

## 2.3 Poursuite de trajectoire

On opte pour le présent document à essayer deux chemins afin d'effectuer une poursuite vers consigne donnée.

### 2.3.1 Généralisation de la stabilisation

La généralisation de la stabilisation comme son non indique porte sur la meme procédure qu'on vient de traiter dans la section précédente sauf ici au lieu d'utiliser un retour d'état on opte à le remplacer par un retour d'état d'erreur qui se manifeste par l'écart entre l'état du système TS et l'état qu'on désire aller vers nous donnant l'expression de commande qui suit :

$$U_{poursuite} = - \sum_{j=1}^r \mu_j \cdot K_j \cdot e \quad \text{avec} \quad e = X - X_d \quad (2.29)$$

Le système en boucle fermée sera forcé à se stabiliser vers l'origine de l'erreur qui est une translation de l'origine naturelle.

Les gains calculés pour la stabilisation classique sont les mêmes que ceux-ci et ca n'a aucune influence sur la stabilité de la boucle fermée, et c'est ce qui va être illustré dans l'implémentation du modèle du quadri-rotor ou les mêmes gains seront utilisés pour la poursuite.

### 2.3.2 Stabilisation de l'erreur de poursuite

L'approche a été introduit par [16]

Etant donnée un vecteur d'état  $X_d$  qui évolue selon une trajectoire désirée, on définit l'espace d'état de l'erreur qui suit :

$$\dot{e} = \dot{X} - \dot{X}_d \quad (2.30)$$

On explicite

$$\dot{e} = \sum_{i=1}^r \mu_i \cdot (A_i \cdot X + B_i \cdot U) - \dot{X}_d \quad (2.31)$$

Dans le coté du  $X$  on ajoute et on retranche  $X_d$ ,

$$\dot{e} = \sum_{i=1}^r \mu_i \cdot (A_i \cdot e + A_i \cdot X_d + B_i \cdot U) - \dot{X}_d \quad \text{avec} \quad e = X - X_d \quad (2.32)$$

Et donc

$$\dot{e} = \sum_{i=1}^r \mu_i \cdot A_i \cdot e + \sum_{i=1}^r \mu_i \cdot A_i \cdot X_d + \sum_{i=1}^r \mu_i \cdot B_i \cdot U - \dot{X}_d \quad (2.33)$$

On établit le changement de variable

$$\sum_{i=1}^r \mu_i \cdot B_i \cdot w = \sum_{i=1}^r \mu_i \cdot A_i \cdot X_d + \sum_{i=1}^r \mu_i \cdot B_i \cdot U - \dot{X}_d \quad (2.34)$$

Ce qui permet d'écrire

$$\dot{e} = \sum_{i=1}^r \mu_i \cdot (A_i \cdot e + B_i \cdot w) \quad (2.35)$$

L'espace d'état de l'erreur s'est transformé vers une représentation TS usuelle et ayant les mêmes caractéristiques dynamiques  $A_i$  et  $B_i$  sauf ici une transformation inverse sera nécessaire afin de retrouver le signal de commande naturel.

On met

$$\sum_{i=1}^r \mu_i \cdot B_i = Bts \quad \text{et} \quad \sum_{i=1}^r \mu_i \cdot A_i = Ats \quad (2.36)$$

$$Bts \cdot w = Ats \cdot X_d + Bts \cdot U - \dot{X}_d \quad (2.37)$$

On déduit que

$$U = w + Bts^{-1} \cdot (\dot{X}d - Ats \cdot Xd) \quad (2.38)$$

À la fin

$$U = w + \left( \sum_{i=1}^r \mu_i \cdot Bi \right)^{-1} \cdot (\dot{X}d - \sum_{i=1}^r \mu_i \cdot Ai \cdot Xd) \quad (2.39)$$

Dans (2.35) puisque il s'agit du meme système TS la meme loi de commande utilisée dans la stabilisation de l'état vers l'origine sera utilisée sauf ici on opte à remplacer l'état du système par l'état de l'erreur et on garde les meme gains précédents,

$$w = - \sum_{j=1}^r \mu_j \cdot Kj \cdot e \quad (2.40)$$

La dérivée de la fonction de Lyapounov résultante obéit à la meme forme précédente dans (2.8)

Si

$$V = e^T P e \quad (2.41)$$

Alors

$$\dot{V} = e^T \cdot \left( \sum_{i=1}^r \sum_{j=1}^r \mu_i \cdot \mu_j \cdot (Gij^T \cdot P + P \cdot Gij) \right) \cdot e < 0 \quad (2.42)$$

$$\sum_{i=1}^r \sum_{j=1}^r (Gij^T \cdot P + P \cdot Gij) < 0 \quad (2.43)$$

L'espace d'état de l'erreur entre l'état naturel et celui désiré arrive à se stabiliser vers l'origine asymptotiquement en résolvant le problème des LMIs (2.43).

Il est à noter que les deux approches traitées pour la poursuite se diffèrent quasiment malgré la ressemblance, en effet la première méthode (2.4.1) comporte uniquement un retour d'état d'erreur ,pour certaines consignes qui évoluent à hautes fréquences ca risque que le système n'arrive pas à poursuivre cette trajectoire car il n'est pas renseigné sur l'état de la dérivée pour cela la deuxième méthode (2.4.2) survient à résoudre ce problème en introduisant une expression de commande comportant les différents termes des dérivées qui une propriété impératif dans la synthèse des loi de commande car la connaissance de l'objectif actuel du système ne demeure pas toujours suffisant pour décider sur l'action à faire prochainement parfois il faudrait connaitre le comportement que devrait le système faire au préalable pour que ce dernier suive la trajectoire d'état la plus adéquate possible faces aux exigences et c'est grâce à l'information que porte la dérivée.

Par contre pour certains processus simple à grandes inerties la méthode (2.4.1) est plus recommandée vu sa simplicité d'implémentation par rapport à la deuxième qui nécessite des inversions matricielles et parfois les matrices Bi ne sont pas carrées il faudrait passer par le calcul au niveau scalaire plus bas une problématique qui sera traitée prochainement dans le modèle du quadri-rotor dont les deux sous-systèmes ont des matrices Bi non carrées.

## 2.4 Poursuite de trajectoire du quadri-rotor

Nous allons procéder par une régulation en cascade dont le régulateur de position calcule des signaux de commande qui se transforment par les contraintes non-holonomes sous formes des deux consignes d'angles  $\theta r$  et  $\phi r$ .

Il est important d'indiquer que la boucle esclave étant le composant primaire qui décide sur la performance du système global, le régulateur des angles doit assurer une convergence précise des angles et en temps minimum vers ces consignes.

### 2.4.1 Première approche

La loi de commande prend la forme classique :

Pour le sous-système des angles on garde les mêmes contraintes sur les régions et donc les memes gains seront employés :

$$U = - \sum_{j=1}^4 \mu_j . K_j . e \quad \text{avec} \quad e = X_{angle} - Xd_{angle} \quad (2.44)$$

Pour le sous-système de position le calcul direct des gains par l'algorithme des LMI sur « Matlab » donne les memes problèmes vécus comme le cas des angles (placement aléatoire des pôles), on opte à utiliser le placement de pôles à travers deux contraintes seulement ayant les paramètres suivants :

$$\alpha^{xyz} = -0.5 \quad \text{et} \quad \beta^{xyz} = 0 \quad (2.45)$$

et puis on s'appuie sur l'expression de commande en position suivante :

$$U_{xyz} = - \sum_{j=1}^2 \mu_j^{xyz} . K_j^{xyz} . e^{xyz} \quad \text{avec} \quad e^{xyz} = X^{xyz} - Xd^{xyz} \quad (2.46)$$

Le calcul des gains par restriction de la région des LMI résulte les valeurs des gains suivants :

$$K1^{xyz} = K2^{xyz} = \begin{bmatrix} 1.0679 & 0.96 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1.0679 & 0.96 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1.068 & 0.96 \end{bmatrix} \quad (2.47)$$

Les gains trouvés sont égaux car les deux multi-modèles possèdent la meme matrice  $Ai$ .

On s'intéresse dans un premier lieu pour des références en échelons.

On prend,

Les états initiaux sont tous nuls.

Avec,

$$U1 = m \cdot g \quad \text{et} \quad U2 = U3 = U4 = Ux = Uy = 0 \quad (2.48)$$

$$Y_r^{xyz} = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} \quad X3_r = \psi_r = 0 \quad (2.49)$$

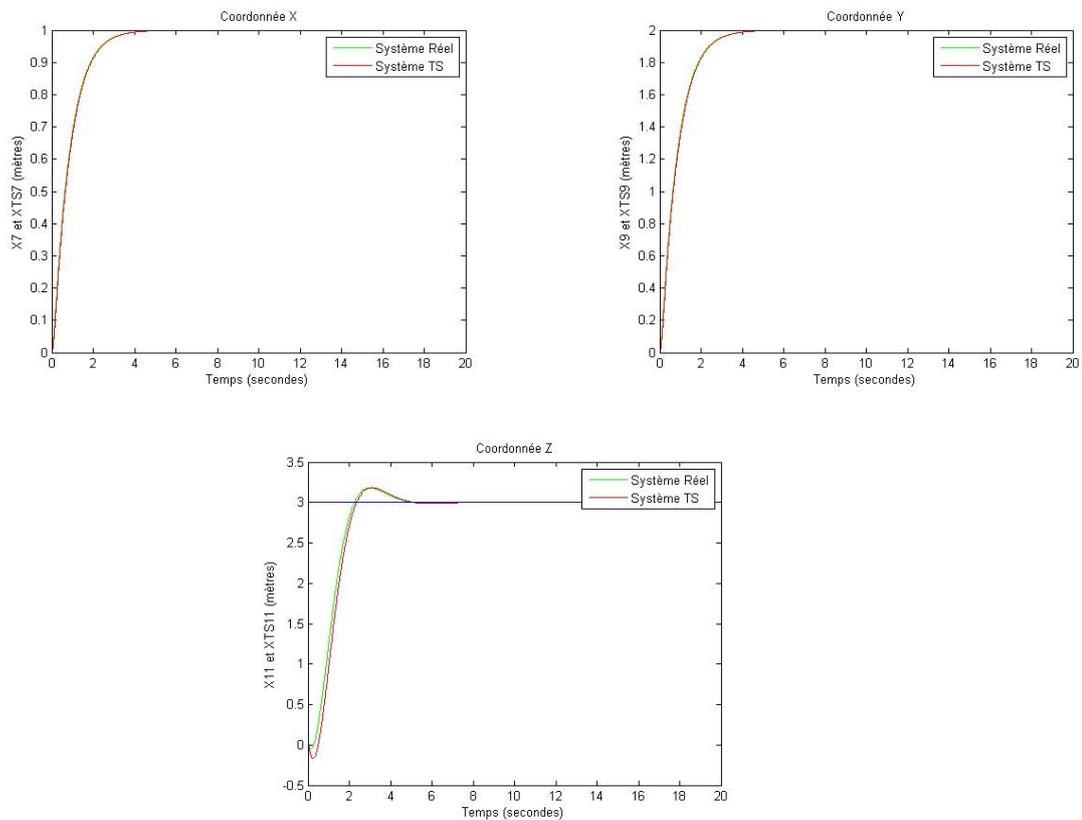


FIG. 2.6 : La sortie du système réel et TS

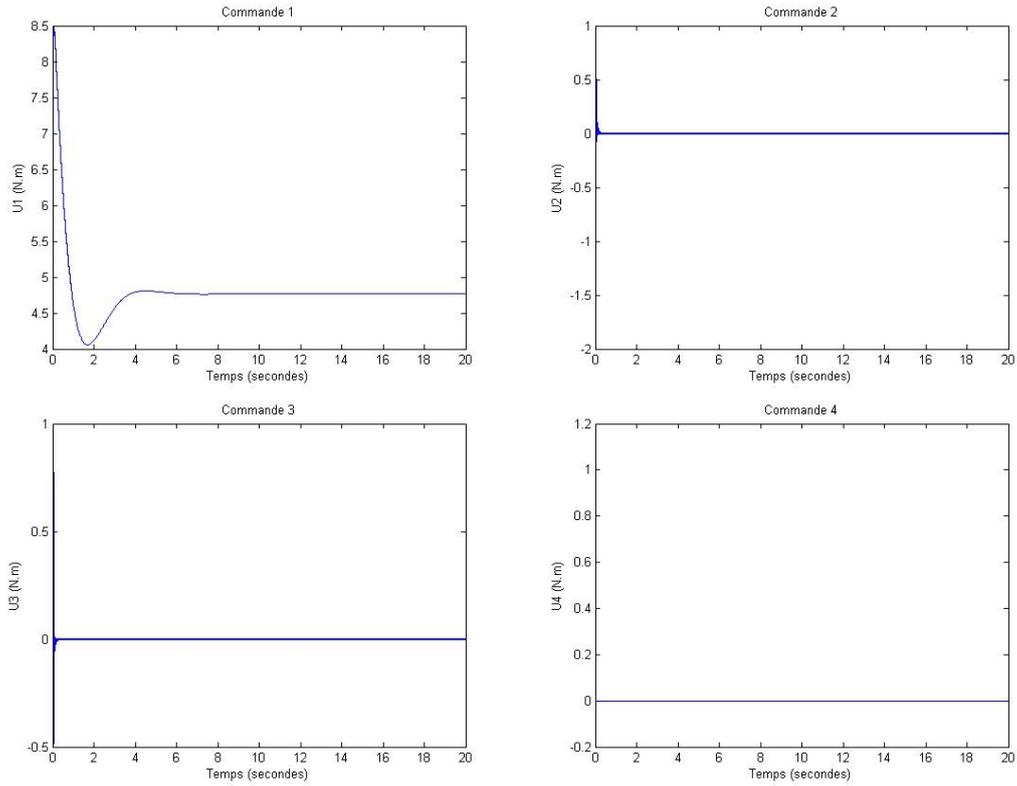


FIG. 2.7 : Les commandes  $U_1$ ,  $U_2$ ,  $U_3$  et  $U_4$

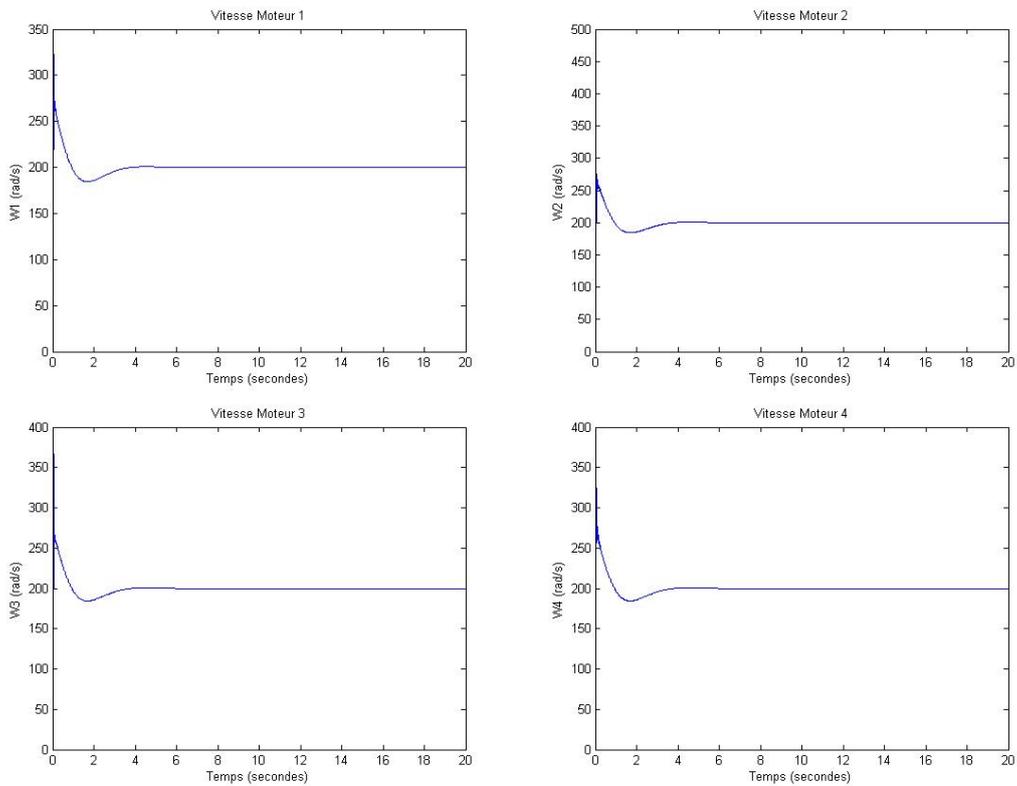


FIG. 2.8 : Les vitesses des moteurs

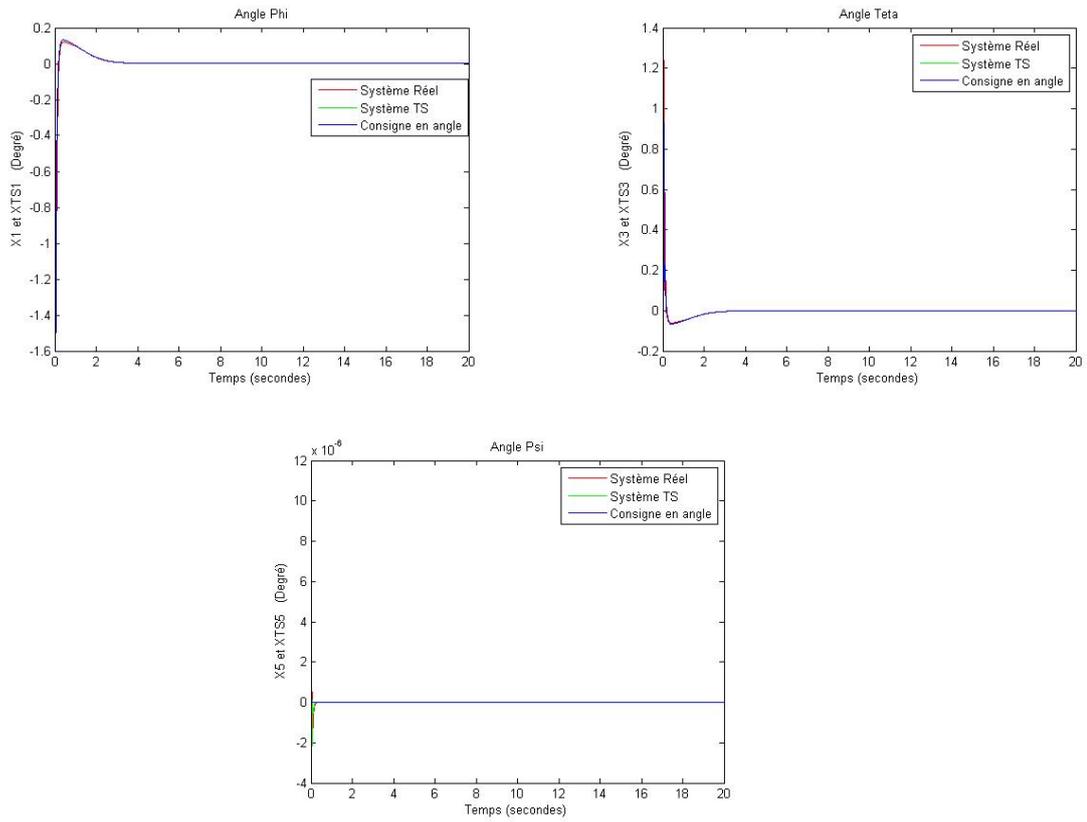


FIG. 2.9 : Les angles

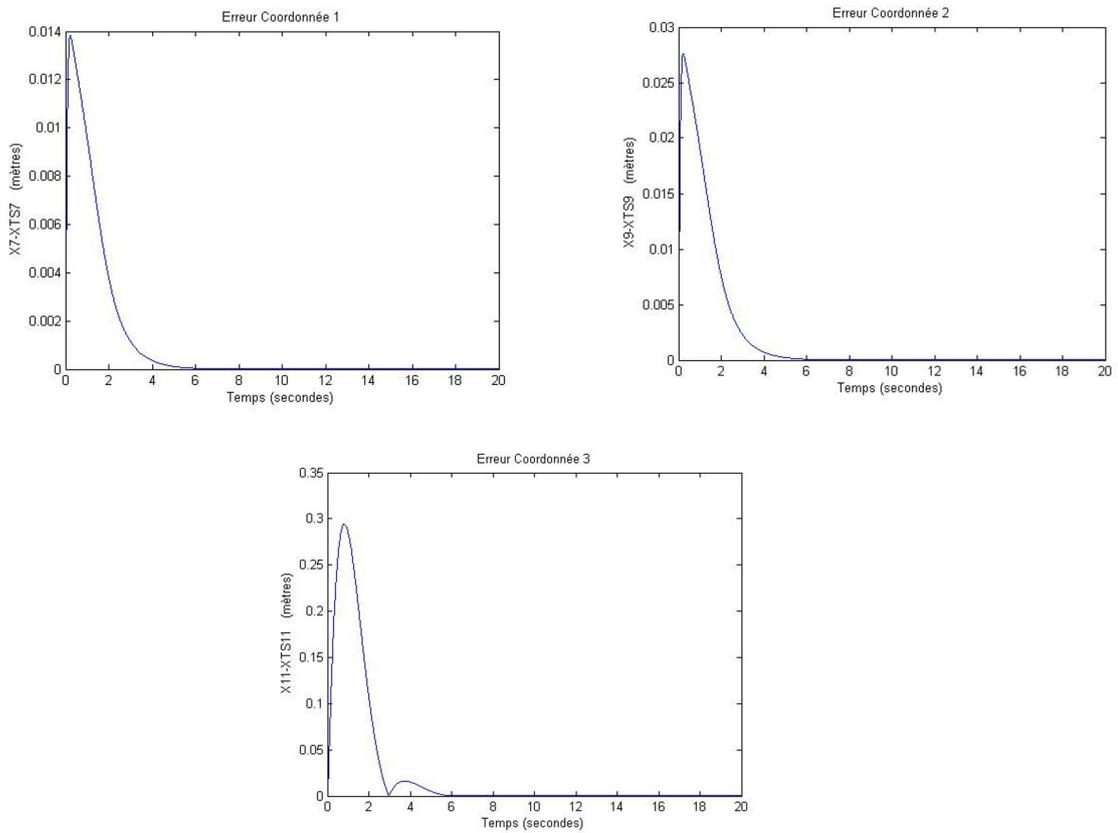


FIG. 2.10 : Les erreurs

Les deux systèmes performant une poursuite vers leurs consignes au bout de 5 secondes selon la meme dynamique et la meme trajectoire.

On impose maintenant une référence variable :

$$Y_r^{xyz} = \begin{bmatrix} \cos(0.5t) \\ \sin(0.5t) \\ t \end{bmatrix} \quad X3_r = \psi_r = 0 \quad (2.50)$$

On doit changer les paramètres de la région des LMI pour le régulateur de position :

$$\alpha^{xyz} = -0.1 \quad \text{et} \quad \beta^{xyz} = 0 \quad (2.51)$$

Ce qui en résulte :

$$K1^{xyz} = K2^{xyz} = \begin{bmatrix} 0.7383 & 0.1458 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.7383 & 0.1458 & 0 & 0 \\ 0 & 0 & 0 & 0 & -0.7383 & -0.1458 \end{bmatrix} \quad (2.52)$$

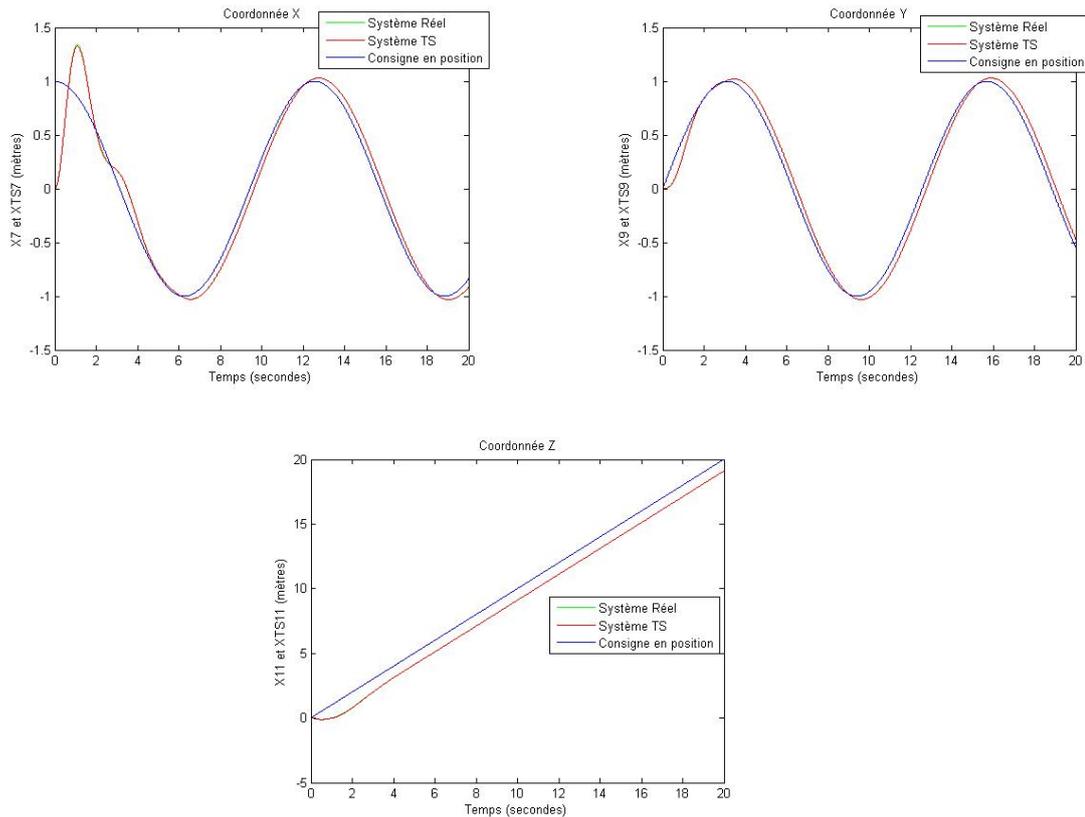


FIG. 2.11 : La sortie du système réel et TS

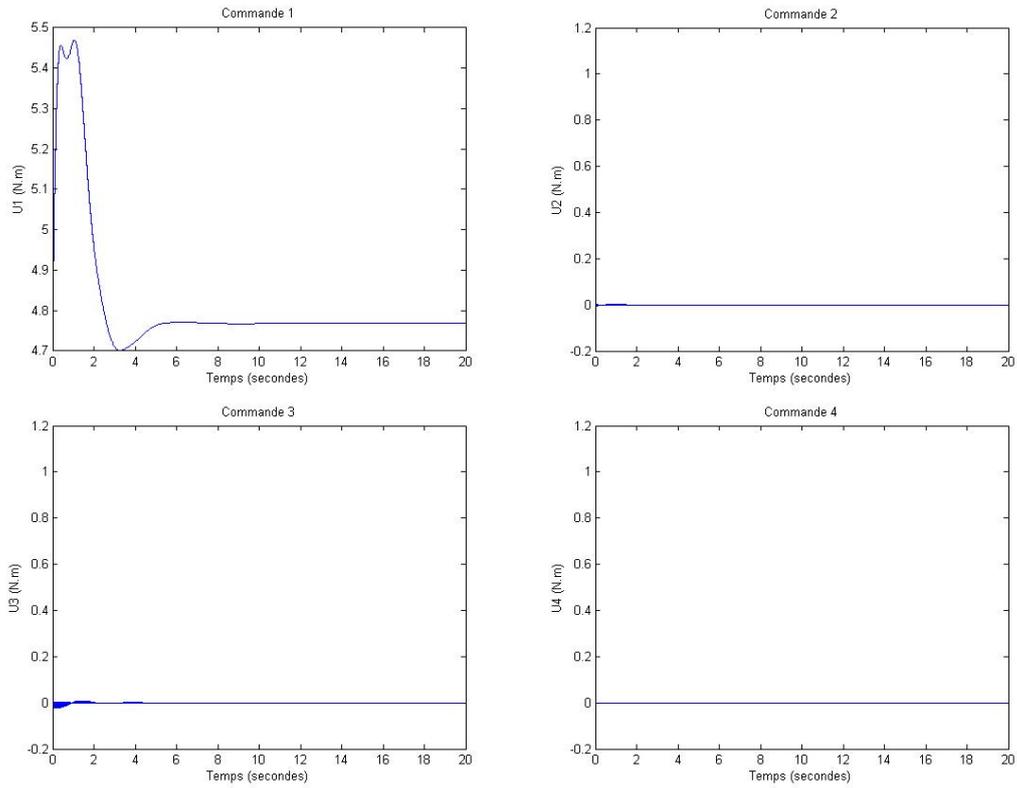


FIG. 2.12 : Les commandes  $U_1$ ,  $U_2$ ,  $U_3$  et  $U_4$

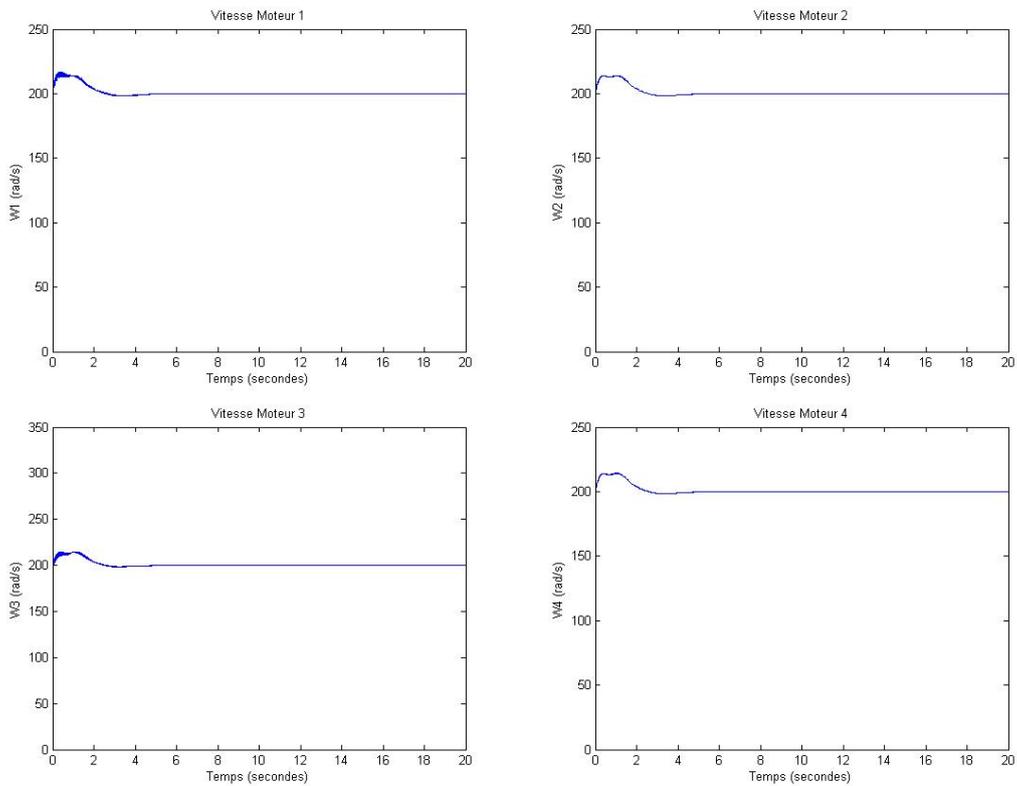


FIG. 2.13 : Les vitesses des moteurs

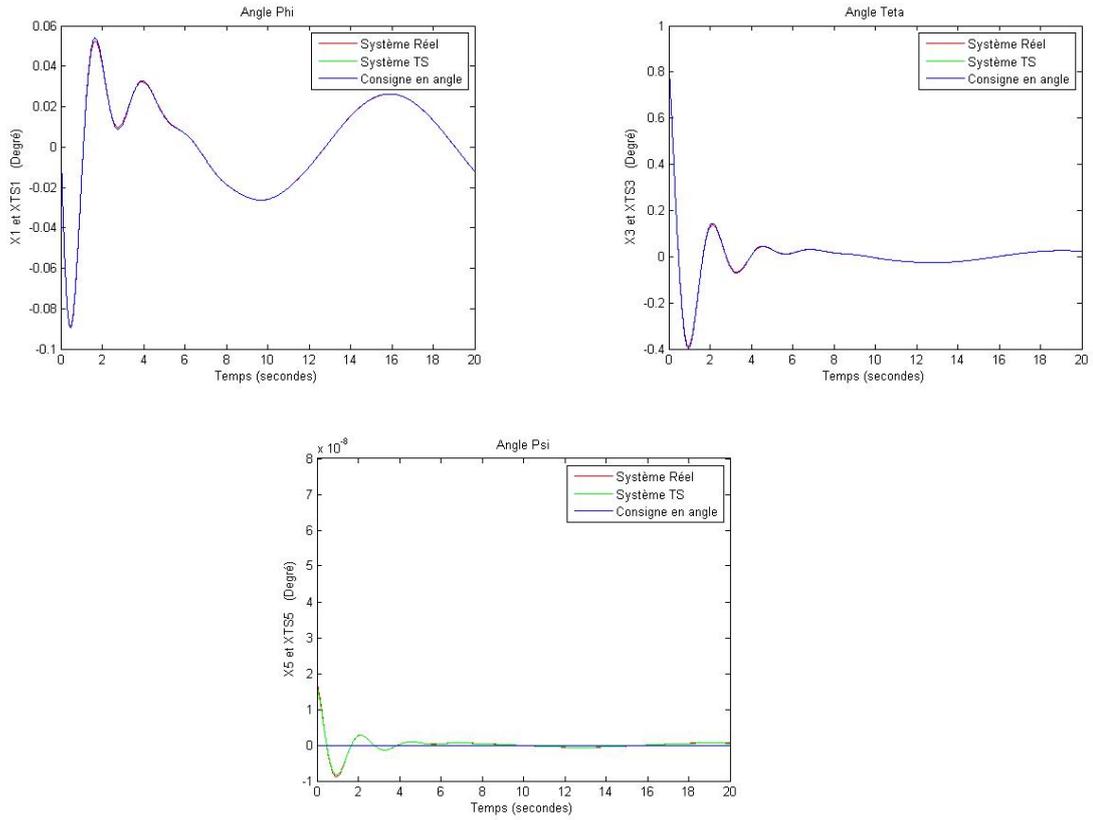


FIG. 2.14 : Les angles

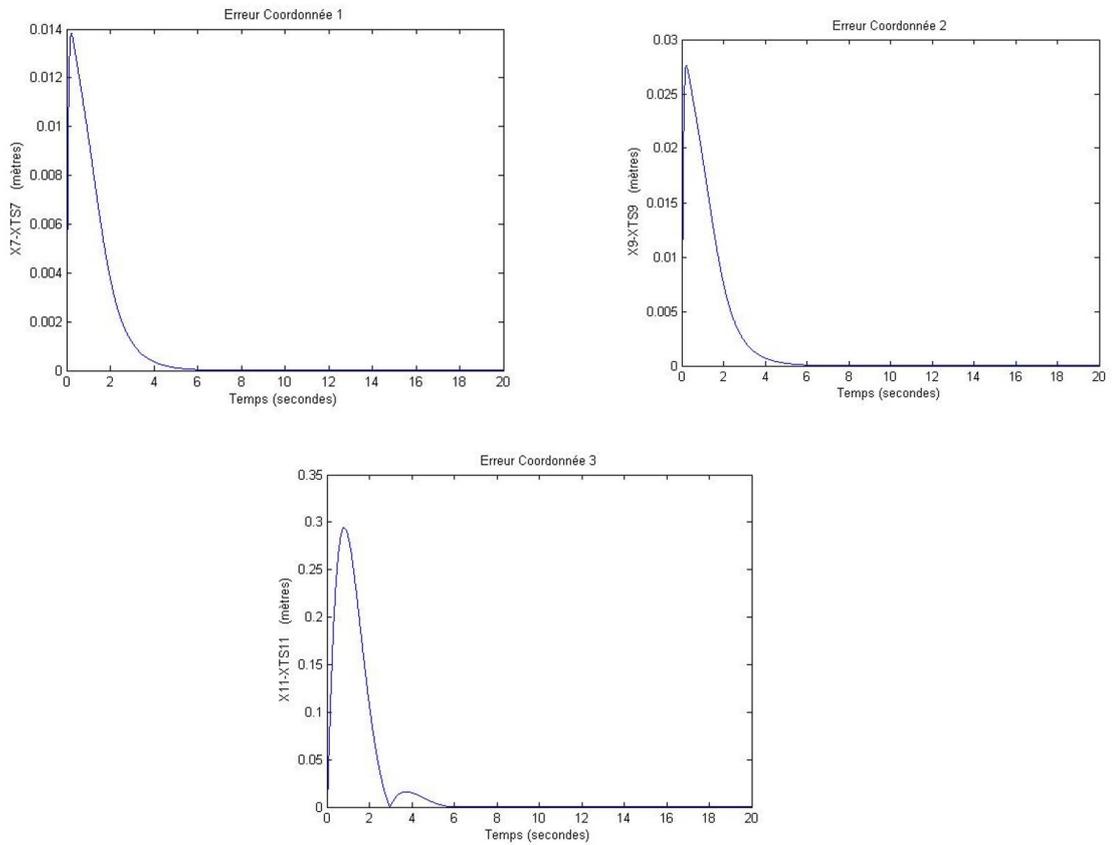


FIG. 2.15 : Les erreurs

On observe que les deux systèmes coïncident l'un sur l'autre en établissant une bonne qualité de poursuite vers la trajectoires hélicoïdales, ca montre bien la validité du modèle TS obtenu et le potentiel de l'approche PDC dont les signaux présentent une poursuite satisfaisante par rapport aux deux commandes utilisées dans les chapitres précédents un compromis entre la précision et le temps de réponse devrait chaque commande comprendre et la théorie du TS flou valide bien ces critères.

Les signaux de commandes évoluent progressivement selon une dynamique lisse ce qui est avantageux par rapport aux commandes non linéaires classiques dont les actionneurs sont toujours sollicités brusquement.

On procède par suite par la deuxième méthode de poursuite des trajectoires

## 2.4.2 Deuxième approche

La loi de commande en position prend la forme suivante :

$$U^{xyz} = w^{xyz} + \left( \sum_{i=1}^2 \mu_i^{xyz} \cdot B_i^{xyz} \right)^{-1} \cdot (\dot{X}d^{xyz} - \sum_{i=1}^2 \mu_i^{xyz} \cdot A_i^{xyz} \cdot Xd^{xyz}) \quad (2.53)$$

avec

$$w^{xyz} = - \sum_{j=1}^2 \mu_j^{xyz} \cdot K_j^{xyz} \cdot e^{xyz} = \begin{bmatrix} w1^{xyz} \\ w2^{xyz} \\ w3^{xyz} \end{bmatrix} \quad (2.54)$$

Les matrices  $B_i^{xyz}$  ne sont pas carrées donc non-inversibles, il faut donc passer au niveau scalaire pour pouvoir calculer chaque signal de commande individuellement.

On reformule l'équation de commande selon celle-ci,

$$\sum_{i=1}^2 \mu_i^{xyz} \cdot B_i^{xyz} \cdot U^{xyz} = \sum_{i=1}^2 \mu_i^{xyz} \cdot B_i^{xyz} \cdot w^{xyz} + (\dot{X}d^{xyz} - \sum_{i=1}^2 \mu_i^{xyz} \cdot A_i^{xyz} \cdot Xd^{xyz}) \quad (2.55)$$

Il est clair que le terme  $\dot{X}d^{xyz} - \sum_{i=1}^2 \mu_i^{xyz} \cdot A_i^{xyz} \cdot Xd^{xyz}$  est un vecteur de dimension 6x1,

On le note par :

$$\tilde{X} = \begin{bmatrix} \tilde{X}1 \\ \tilde{X}2 \\ \tilde{X}3 \\ \tilde{X}4 \\ \tilde{X}5 \\ \tilde{X}6 \end{bmatrix}$$

$$\sum_{i=1}^2 \mu_i^{xyz} \begin{bmatrix} 0 \\ \frac{U_x}{m} \\ 0 \\ \frac{U_y}{m} \\ 0 \\ \frac{U_z}{m} b_i \end{bmatrix} = \sum_{i=1}^2 \mu_i^{xyz} \begin{bmatrix} 0 \\ \frac{w_1^{xyz}}{m} \\ 0 \\ \frac{w_2^{xyz}}{m} \\ 0 \\ \frac{w_3^{xyz}}{m} b_i \end{bmatrix} + \tilde{X} \quad (2.56)$$

Ce qui permet de déduire les signaux de commande

$$U_x = w_1^{xyz} + m.\tilde{X}2 \quad (2.57)$$

$$U_y = w_2^{xyz} + m.\tilde{X}4 \quad (2.58)$$

$$U_z = U_1 = w_3^{xyz} + m.\left(\sum_{i=1}^2 \mu_i^{xyz} .b_i\right)^{-1} \tilde{X}6 \quad (2.59)$$

Selon la meme manière pour le sous-système des angles on a l'équation de commande

$$U = w + \left(\sum_{i=1}^4 \mu_i B_i\right)^{-1} (\dot{X}d - \sum_{i=1}^4 \mu_i A_i Xd) = \begin{bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \end{bmatrix}$$

$$w = - \sum_{j=1}^r \mu_j K_j e \begin{bmatrix} w_1 \\ w_2 \\ w_3 \\ w_4 \end{bmatrix} \quad (2.60)$$

Et donc

$$\sum_{i=1}^4 \mu_i B_i U = \sum_{i=1}^4 \mu_i B_i w + (\dot{X}d - \sum_{i=1}^4 \mu_i A_i Xd) \quad (2.61)$$

avec

$$\tilde{X} = \begin{bmatrix} \tilde{X}1 \\ \tilde{X}2 \\ \tilde{X}3 \\ \tilde{X}4 \\ \tilde{X}5 \\ \tilde{X}6 \end{bmatrix} = \dot{X}d - \sum_{i=1}^4 \mu_i A_i Xd \quad (2.62)$$

Ce qui donne

$$\begin{bmatrix} 0 \\ U_2 b_1 \\ 0 \\ U_3 b_2 \\ 0 \\ U_4 b_3 \end{bmatrix} = \begin{bmatrix} 0 \\ w_2 b_1 \\ 0 \\ w_3 b_2 \\ 0 \\ w_4 b_3 \end{bmatrix} + \begin{bmatrix} \tilde{X}1 \\ \tilde{X}2 \\ \tilde{X}3 \\ \tilde{X}4 \\ \tilde{X}5 \\ \tilde{X}6 \end{bmatrix} \quad (2.63)$$

Par conséquent

$$U_2 = w_2 + \frac{\tilde{X}2}{b_1} \quad (2.64)$$

$$U_3 = w_3 + \frac{\tilde{X}4}{b_2} \quad (2.65)$$

$$U_4 = w_4 + \frac{\tilde{X}6}{b_3} \quad (2.66)$$

On implémente les signaux obtenus selon la meme configuration des paramètres et des conditions initiales utilisés dans la première méthode (2.48).

On reprend la consigne échelon précédente

$$Y_r^{xyz} = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} \quad X3_r = \psi_r = 0 \quad (2.67)$$

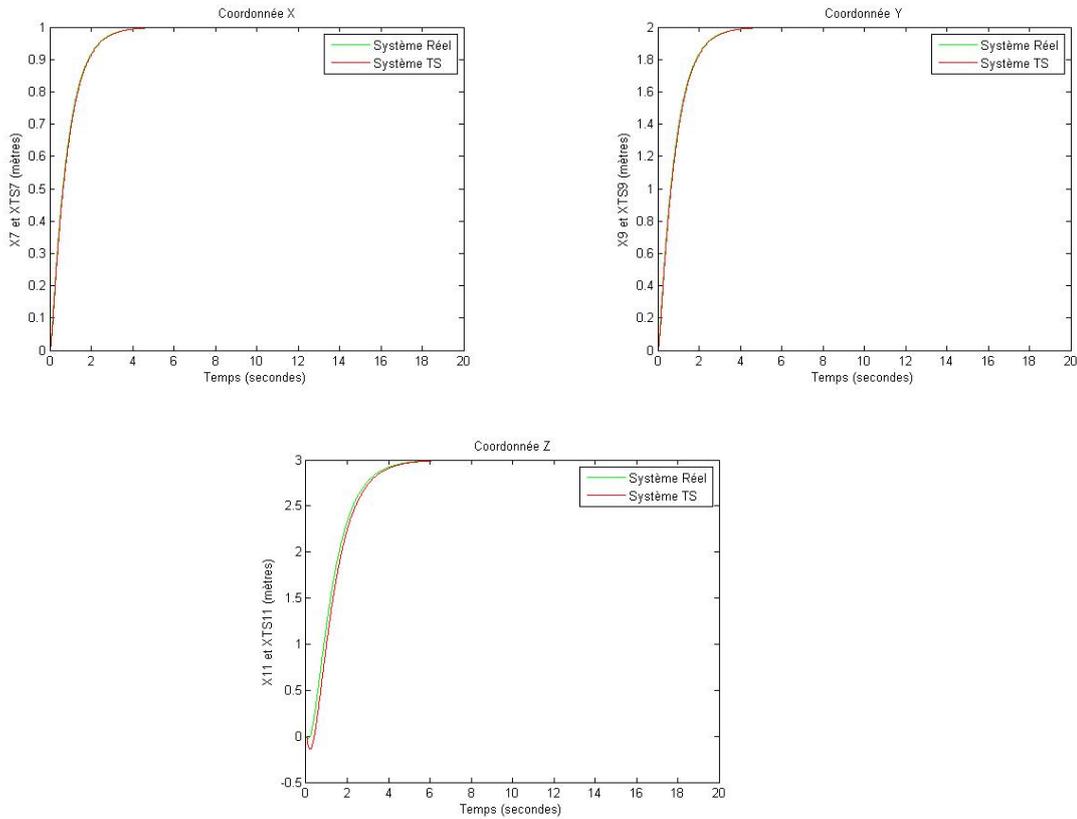


FIG. 2.16 : La sortie du système réel et TS

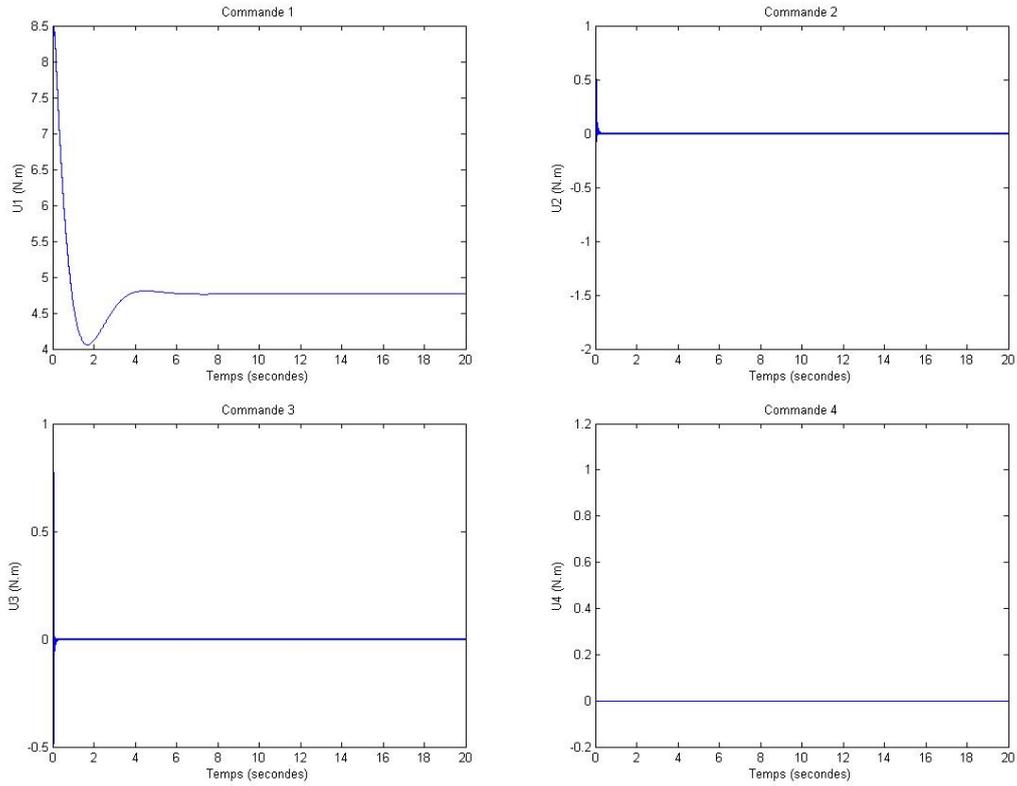


FIG. 2.17 : Les commandes U1, U2, U3 et U4

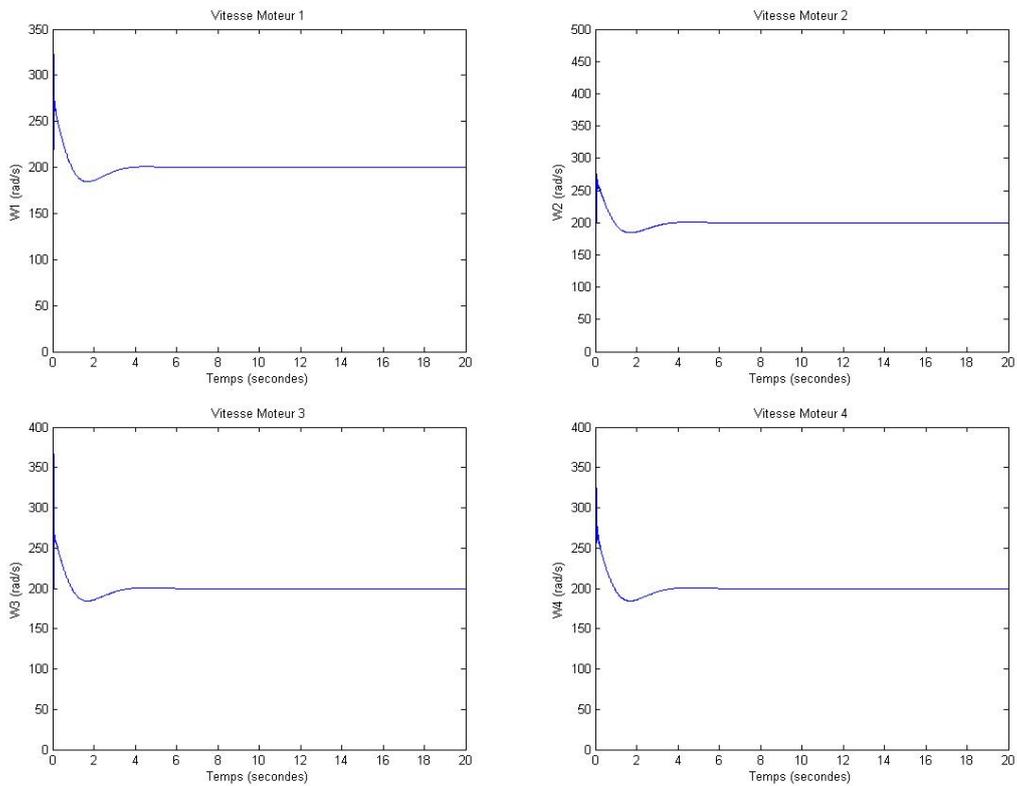


FIG. 2.18 : Les vitesses des moteurs

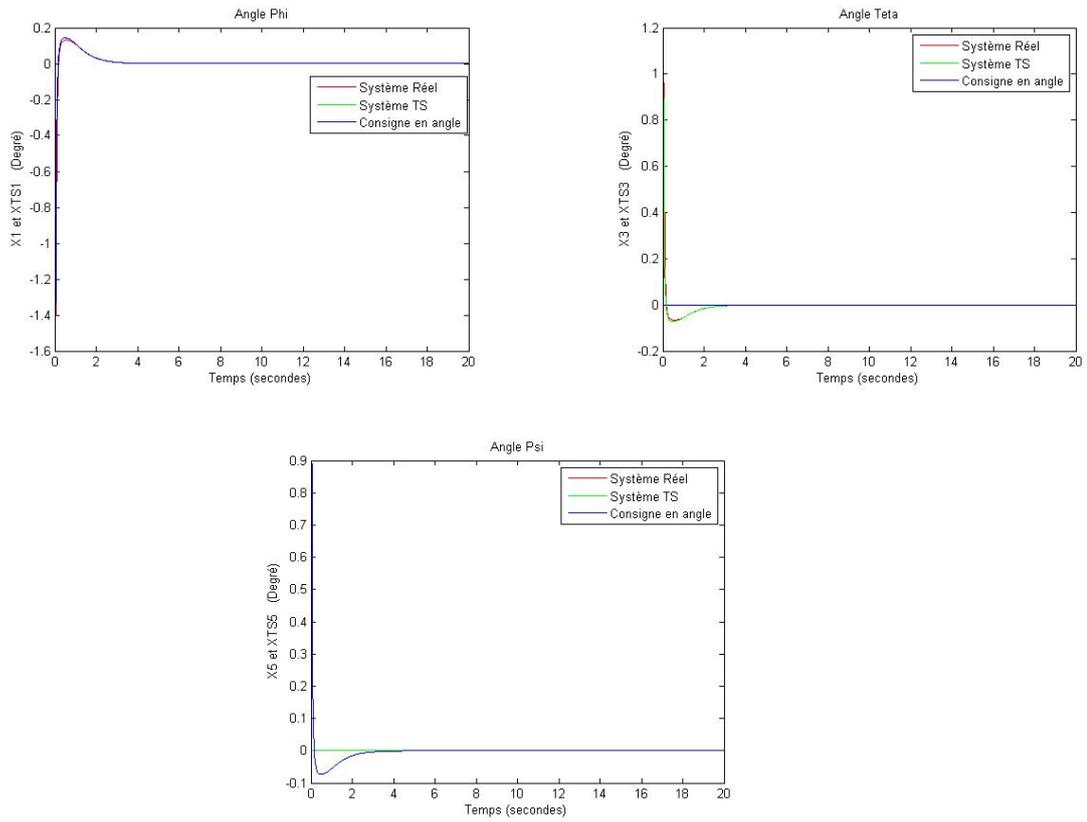


FIG. 2.19 : Les angles

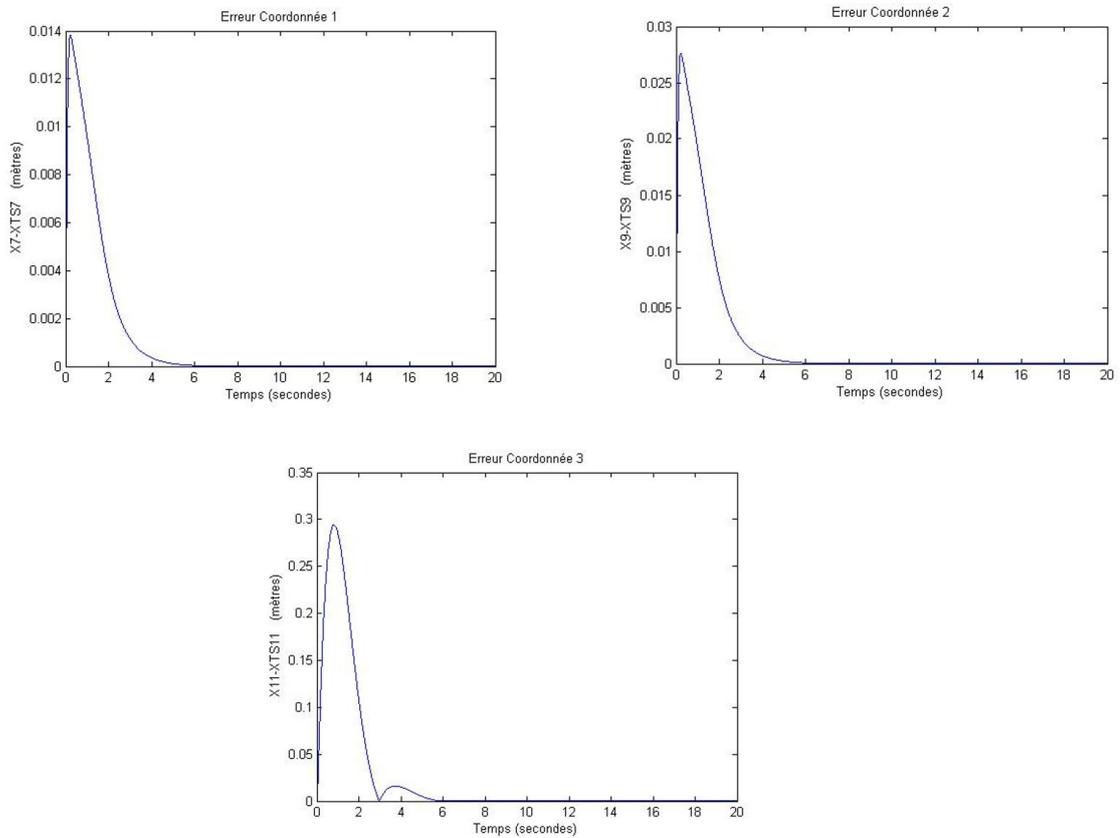


FIG. 2.20 : Les erreurs

La deuxième méthode donne des résultats exacts que ceux de la première.

On passe à la référence variable la meme qui a été employée précédemment.

$$Y_r^{xyz} = \begin{bmatrix} \cos(0.5t) \\ \sin(0.5t) \\ t \end{bmatrix} \quad X3_r = \psi_r = 0 \quad (2.68)$$

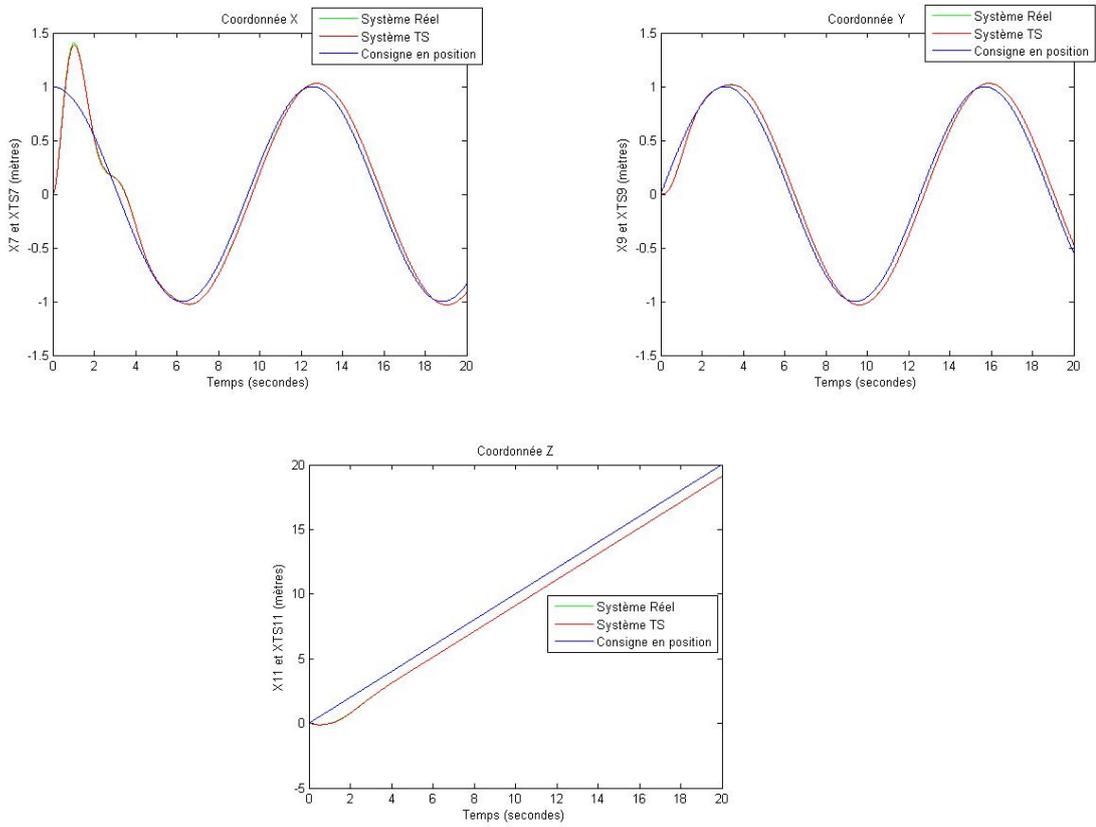


FIG. 2.21 : La sortie du système réel et TS

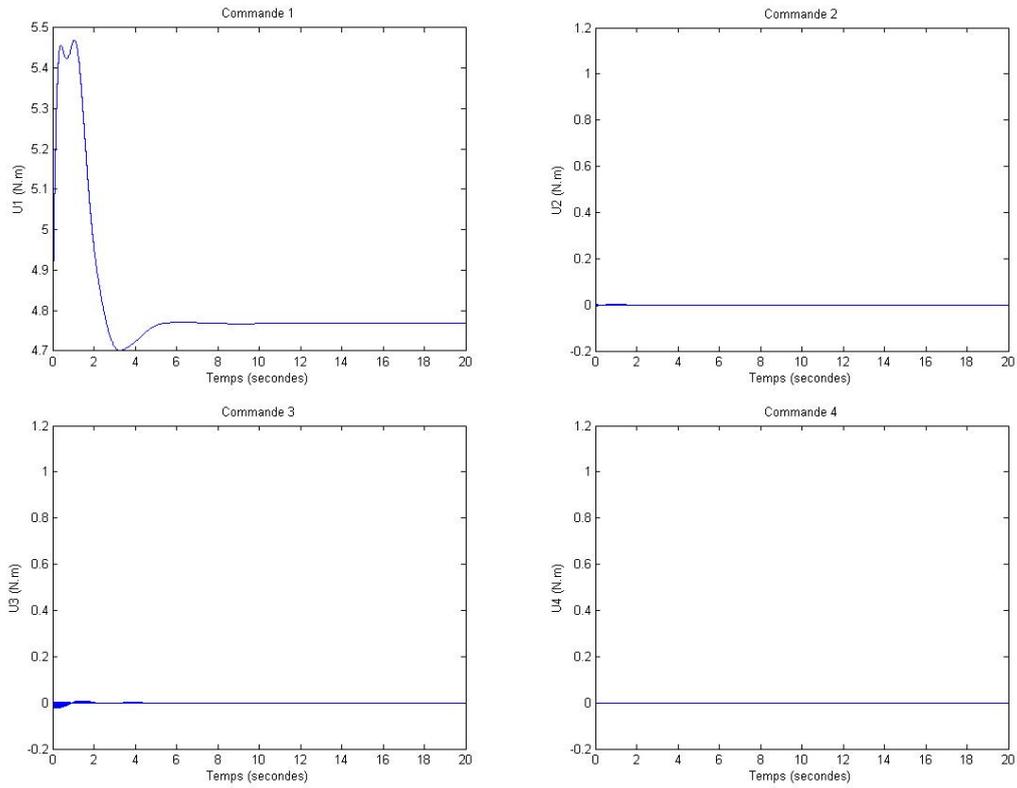


FIG. 2.22 : Les commandes U1, U2, U3 et U4

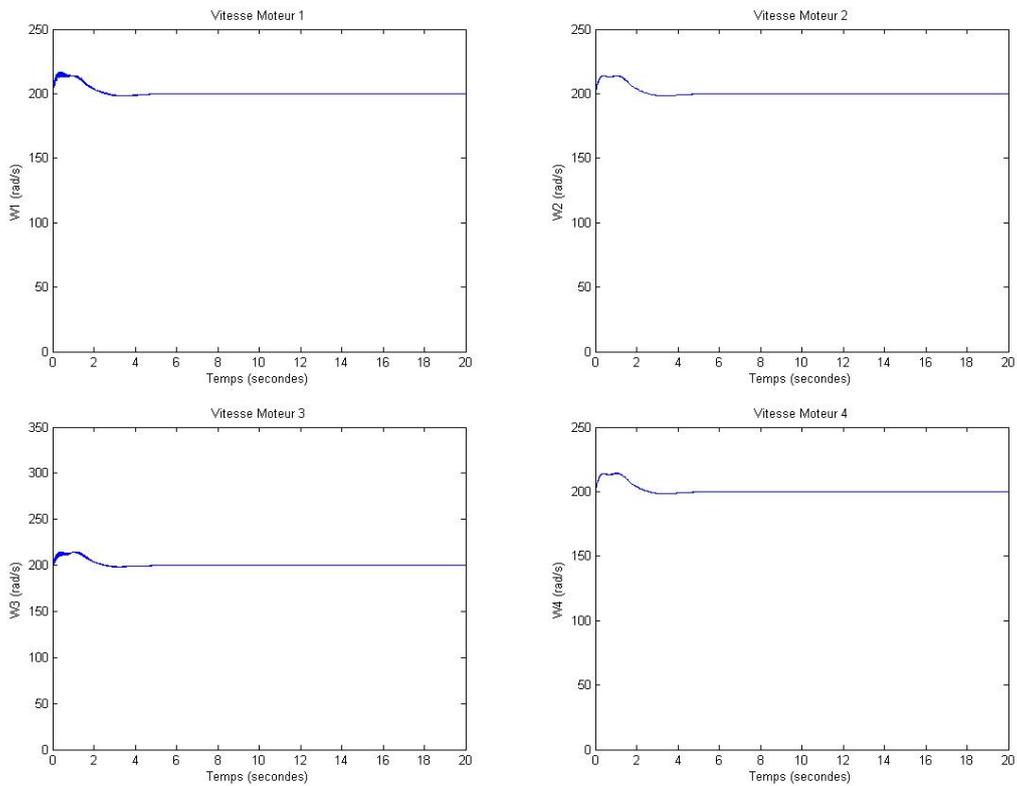


FIG. 2.23 : Les vitesses des moteurs

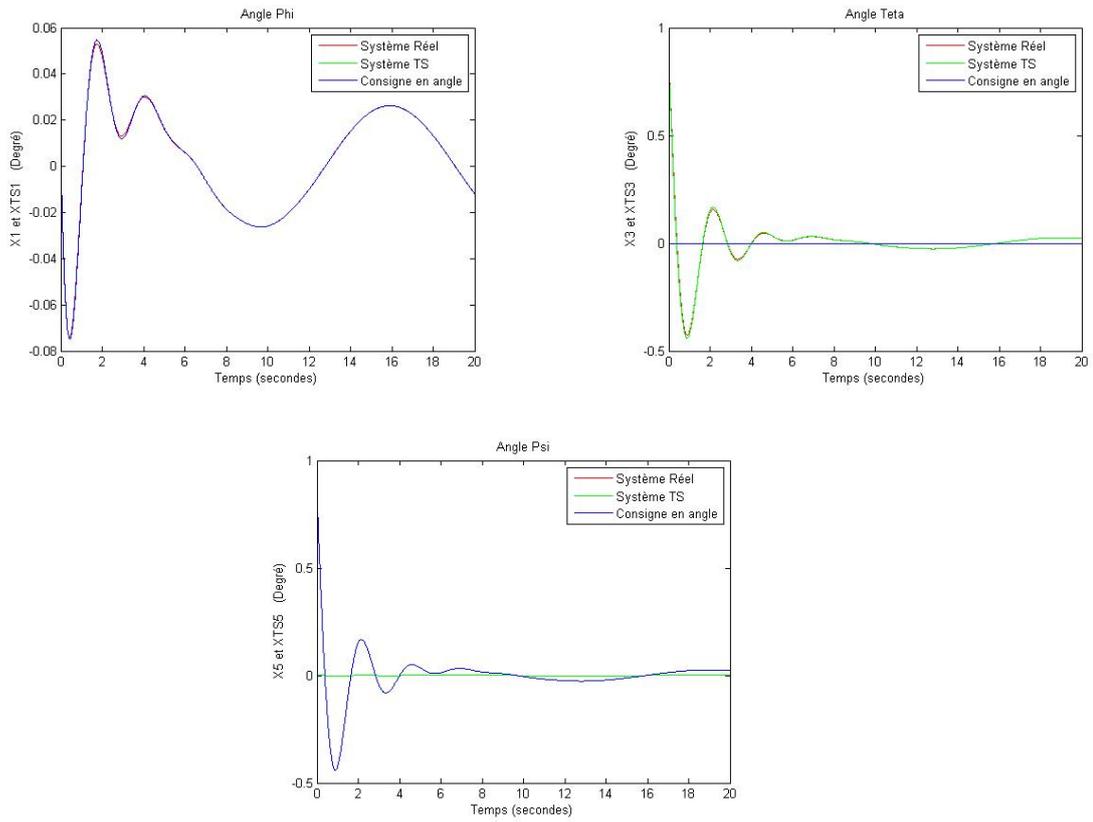


FIG. 2.24 : Les angles

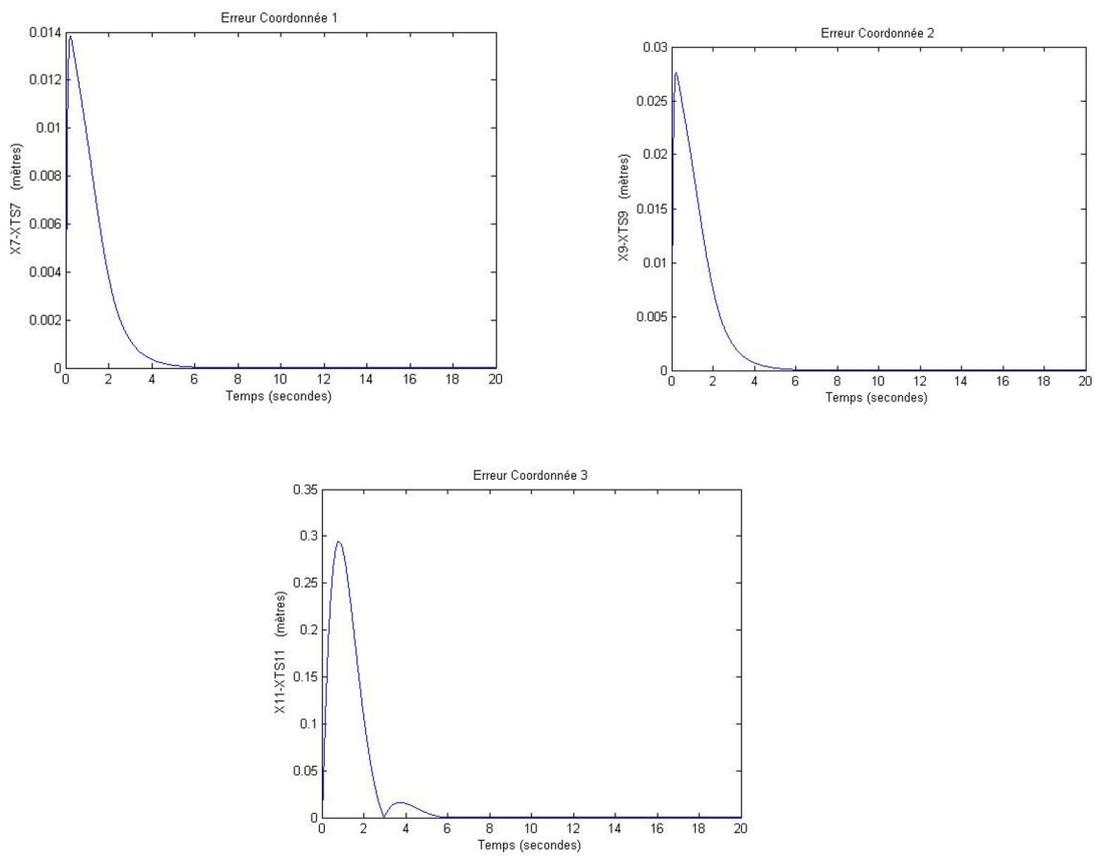


FIG. 2.25 : Les erreurs

Les résultats obtenus coïncident avec la méthode déployée précédemment, cependant il est préférable d'utiliser cela vu que c'est une méthode qui fait appel aux caractéristiques de la dérivée de la consigne une information importante pouvant faire la différence pour les hautes fréquences ce qui est bien validé dans les littératures des commandes non linéaires mais parfois la première méthode est plus recommandée vu sa simplicité d'implémentation par rapport à celle là.

### 2.4.3 Comparaison avec la commande par mode glissant

On introduit la consigne suivante :

$$Y_r^{xyz} = \begin{bmatrix} \cos(0.5t) \\ \sin(0.5t) \\ 3 \end{bmatrix} \quad X3_r = \psi_r = 0 \quad (2.69)$$

On garde les mêmes configurations paramétriques précédentes des régulateurs pour les deux méthodes, cette fois-ci nous utiliserons la deuxième approche de poursuite TS (2.4.2) pour la faire comparer avec l'approche de glissement.

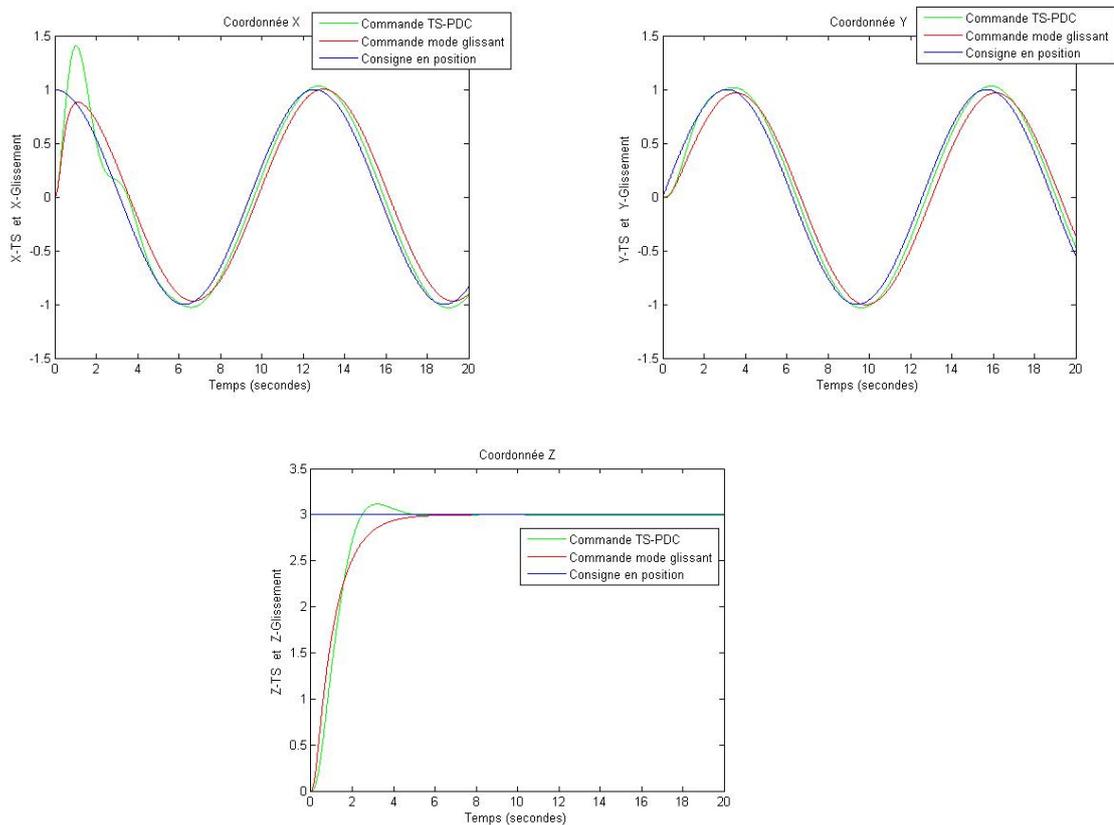


FIG. 2.26 : La sortie du système réel et TS

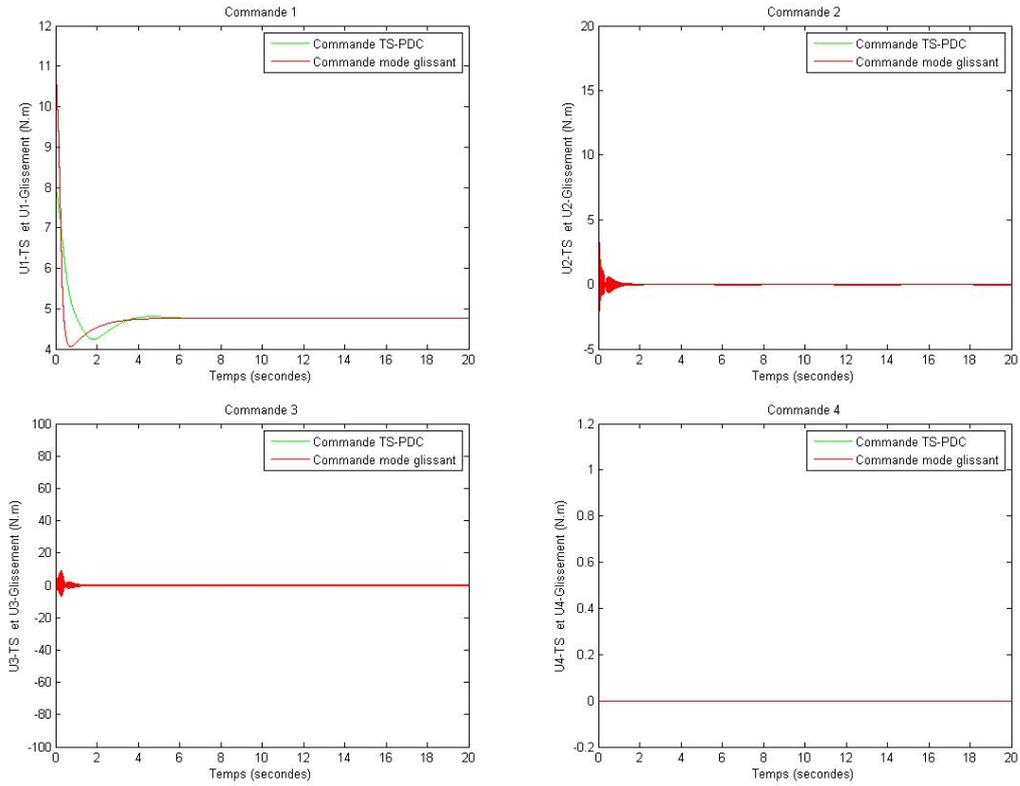


FIG. 2.27 : Les commandes U1, U2, U3 et U4

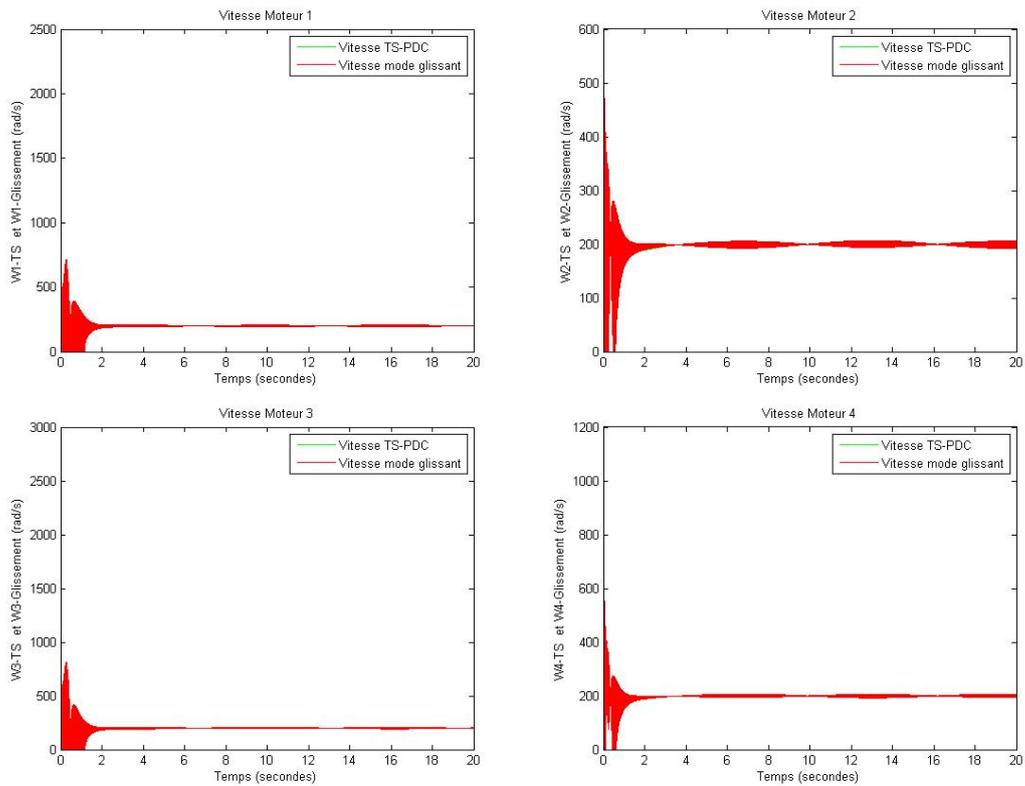


FIG. 2.28 : Les vitesses des moteurs

D'une part, on aperçoit que les deux régulateurs présentent une vitesse de poursuite proche, mais la méthode du TS introduit une meilleure précision durant le régime permanent, si on observe les signaux de commande en couples et en vitesses on trouve que les signaux du mode glissant sont plus brusques alors que la poursuite de la trajectoire selon TS est plus souple et lisse cela est important lorsque on désire choisir le bon régulateur dans une implémentation pratique car cette légère différence peut conduire vers des conséquences importantes en terme d'économie d'énergie.

La poursuite de trajectoire par approche PDC introduit un bon compromis entre les performances de poursuite et le coût d'énergie consommée par les propulseurs malgré, la théorie du TS reste valide donc.

Dans les chapitres venant, on se limite uniquement avec la commande TS-PDC pendant la comparaison puisque on arrivait à prouver l'amélioration nette par rapport à l'aspect du glissement

## 2.5 Conclusion

Le présent chapitre était une illustration du potentiel de la théorie du PDC, nous nous sommes intéressés dans un premier lieu par la stabilisation des angles du quadrirotor moyennant le retour d'état flou calculé par les algorithmes des LMI, et puis nous avons abordé la poursuite des trajectoires selon deux manières différentes la généralisation de la stabilisation et puis la méthode de l'espace d'état de l'erreur qui ont conduit toutes vers des bons résultats de poursuite et à la fin nous avons clôturé ce chapitre par la comparaison avec une procédure de commande non linéaire classique qui est le mode glissant dont laquelle nous nous sommes retrouvés avec des résultats presque proches avec une légère amélioration en performance pour le PDC. Nous avons conclu donc que la procédure du TS présente un potentiel important dans la théorie du contrôle ou on arrive à synthétiser des asservissements linéaires qui se basent sur des représentations non linéaires tout en préservant une bonne qualité de performance par rapport aux approches classiques.

# Chapitre 3

## Commande par poursuite d'un modèle de référence

## 3.1 Introduction

Le modèle de référence est largement employé dans la littérature, il est clair que dans les précédentes sections on avait vu que chaque commande possède ces limites de performances chacune nécessite des tests et de modification des différents paramètres que suggère le régulateur mais parfois lorsque on désire imposer ou bien améliorer un comportement dynamique donné les commandes classiques ne pourront pas satisfaire le cahier de charge désiré.

Le modèle de référence est l'un des fameuses méthodes qui arrive à remédier ces exigences et contraintes, ça consiste à développer un système autonome de même nature que celui du système réel et ayant le même ordre, le système choisi doit évoluer selon une trajectoire d'état et selon des performances dynamiques voulues.

Après avoir choisi le comportement désiré, il convient par suite d'établir l'espace d'état d'erreur entre le procédé et le système autonome et après on choisit une des commandes pour stabiliser cet espace d'état vers l'origine de l'erreur.

Dans la majorité des cas les systèmes autonomes sont choisis de nature linéaire pure pour simplifier la procédure de synthèse mais parfois la complexité des non linéarités que stipule le procédé nécessite à choisir un modèle non linéaire bien adapté à l'espace d'état réel.

Dans le présent chapitre, nous allons procéder via cet enchaînement en se basant sur la commande PDC afin de stabiliser l'erreur de poursuite.

## 3.2 Approche

La méthode a été employée par [4].

On considère un système TS proche d'un modèle non linéaire quelconque affine en commande :

$$\dot{X} = \sum_{i=1}^r \mu_i (A_i X + B_i U) \approx f(x) + g(x)U \quad (3.1)$$

Soit le modèle de référence qui performe une poursuite vers une consigne :

$$\dot{X}_r = A_r X_r + B_r r(t) \quad (3.2)$$

Tel que  $\dim(X) = \dim(X_r)$  et  $r(t)$  : Signal de référence

L'espace d'état de l'erreur entre le modèle TS et la référence se définit par :

$$\dot{e} = \dot{X} - \dot{X}_r \quad (3.3)$$

Ca permet d'écrire,

$$\dot{e} = \sum_{i=1}^r \mu_i (A_i X + B_i U) - A_r X_r - B_r r(t) \quad (3.4)$$

On ajoute et on retranche au coté de  $X$  le terme  $X_r$  pour faire apparaître l'erreur,

$$\dot{e} = \sum_{i=1}^r \mu_i \cdot (A_i \cdot e + A_i \cdot X_r + B_i \cdot U) - A_r \cdot X_r - B_r \cdot r(t) \quad (3.5)$$

Et donc,

$$\dot{e} = \sum_{i=1}^r \mu_i \cdot (A_i \cdot e + B_i \cdot U) + \left( \sum_{i=1}^r \mu_i \cdot A_i - A_r \right) \cdot X_r - B_r \cdot r(t) \quad (3.6)$$

On note,

$$\sum_{i=1}^r \mu_i \cdot B_i = B_{ts} \quad \text{et} \quad \sum_{i=1}^r \mu_i \cdot A_i = A_{ts} \quad \text{avec} \quad A_{ts} - A_r = \Delta A \quad (3.7)$$

Pour avoir,

$$\dot{e} = A_{ts} \cdot e + B_{ts} \cdot U + \Delta A X_r - B_r \cdot r(t) \quad (3.8)$$

On effectue un changement de variable de la forme :

$$\sum_{i=1}^r \mu_i \cdot B_i w = \sum_{i=1}^r \mu_i \cdot B_i \cdot U + \left( \sum_{i=1}^r \mu_i \cdot A_i - A_r \right) X_r - B_r \cdot r(t) \quad (3.9)$$

Par conséquent,

$$\dot{e} = \sum_{i=1}^r \mu_i \cdot (A_i \cdot e + B_i \cdot w) \quad (3.10)$$

Nous obtenons un espace d'état d'erreur ayant les mêmes caractéristiques matricielles du système TS un résultat qui est attendu puisque la méthode ressemble à celle qui a été utilisée dans la poursuite des trajectoires sauf ici on s'est proposé d'établir l'erreur entre deux espaces d'états et pas entre un modèle d'état et un état désiré.

Selon la même manière que la poursuite des trajectoires la loi de commande stabilisante par approche PDC est :

$$w = - \sum_{j=1}^r \mu_j \cdot K_j \cdot e \quad (3.11)$$

Les gains qui seront utilisés demeurent les mêmes gains qui ont été indiqués dans la stabilisation de l'état naturel par suite, on fait la transformation inverse afin de déduire l'expression de la commande réelle :

$$B_{ts} \cdot w = B_{ts} \cdot U + \Delta A \cdot X_r - B_r \cdot r(t) \quad (3.12)$$

On trouve que,

$$U = w + B_{ts}^{-1} \cdot (B_r \cdot r(t) - \Delta A \cdot X_r) \quad (3.13)$$

D'où,

$$U = - \sum_{j=1}^r \mu_j \cdot K_j \cdot e + \left( \sum_{i=1}^r \mu_i \cdot B_i \right)^{-1} \cdot (B_r \cdot r(t) - \left( \sum_{i=1}^r \mu_i \cdot A_i \right) - A_r) \cdot X_r \quad (3.14)$$

La loi de commande est constituée par une commande classique stabilisante par approche PDC et un terme comprenant les caractéristiques dynamiques du modèle de référence et la consigne que suit ce dernier, ca permet d'assurer une bonne performance de poursuite pas en référence mais en trajectoire d'état  $X \Rightarrow X_r$  tout en gardant la performance que stipule le système autonome ceci est trop fort dans les cas ou un régulateur classique comme un *PID* [12] ou un  $H^\infty$  [11] n'arrive pas à satisfaire un cahier de décharge de hautes performances, l'approche est très pratique pour imposer un comportement dynamique approprié sur le procédé réel sans avoir besoin de réajuster et d'optimiser les paramètres du régulateur.

Il est impératif de choisir un modèle de référence qui ne dépasse pas les limites physiques du système en rapidité et en puissance car si ce dernier évolue selon une fréquence qui dépasse la bande passante intrinsèque du système en boucle fermé l'écart entre ce dernier et le référentiel risque de devenir important ce qui peut provoquer des signaux de commandes non réalistes et donc des dégâts matérielles dans l'actionneur ou meme dans le procédé.

Dans la section qui vient, une implémentation sur le modèle du quadri-rotor sera traitée dans laquelle on va discuter toute les sections précédentes afin de satisfaire l'objet de ce mémoire et puis on clôture par une comparaison récapitulative pour montrer le potentiel et les inconvénients de chaque approches.

### 3.3 Implémentation de la loi de commande

Le modèle de référence choisi doit présenter un compromis entre la précision et la vitesse de poursuite tout en respectant les contraintes que le procédé réel porte sur.

L'avantage majeur de la poursuite du modèle de référence étant l'aptitude de rendre un quelconque système à se comporter comme un système parfait, il y a pas mal de références qui utilisent des systèmes de référence avec commande optimale dont laquelle on pourra optimiser les performances dynamiques tout en préservant les ressources matérielles comme la puissance et le vieillissement de l'équipement.

Dans le présent mémoire on se limite avec un système de référence qui performe une poursuite vers la trajectoire désirée par une action intégral d'un retour d'état [18].

#### 3.3.1 Sous-système des angles

On effectue la poursuite par sous-système et puis on aborde le cas du système global .

On se propose donc d'établir une loi de commande poursuivant le modèle suivant :

$$\dot{X}_r = A_r \cdot X_r + B_r \cdot r(t) \quad (3.15)$$

$$\dot{X}_r = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ -100.05 & -20.005 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & -100.09 & -20.009 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & -100.01 & -20.001 \end{bmatrix} X_r + B.K_i \int e.dt \quad (3.16)$$

Tel que  $e = y - Y_r$

Le modèle choisi étant exponentiellement stable dans  $\mathbb{R}^6$ , purement linéaire et possédant des valeurs propres rapides :

$$\gamma_i = (-10 \quad -10.001 \quad -10.002 \quad -10.003 \quad -10.004 \quad -10.005) \quad (3.17)$$

$B$  : La matrice naturelle du sous-système des angles

$K_i$  : le gain de l'action intégral

Le gain  $K_i$  se calcul par un retour d'état de l'espace augmenté de l'erreur comme ca été introduit dans [18], en imposant les valeurs propres :

$$\gamma_i^e = (-10 \quad -10.001 \quad -10.002 \quad -10.003 \quad -10.004 \quad -10.005 \quad -0.02 \quad -0.02001 \quad -0.02002) \quad (3.18)$$

On trouve le gain qui vaut :

$$K_i = \begin{bmatrix} 0 & 0 & 0 \\ 0.0307 & 0 & 0 \\ 0 & 0.0307 & 0 \\ 0 & 0 & 0.0613 \end{bmatrix} \quad (3.19)$$

**Remarque :** Les matrices  $A_r$  des deux sous-systèmes sont élaborées par un simple retour d'état de la forme ( $A_r = A_i - b * K$ ) tel que  $K$  est retrouvés par la fonction "place" au niveau de "Matlab" en introduisant les valeurs propres voulues.

L'implémentation de telle loi de commande permet au système de référence d'effectuer une poursuite vers la consigne selon une vitesse et une précision de poursuite assez satisfaisantes.

On s'intéresse maintenant par la loi de commande naturelle que doit le régulateur généré pour le suivie de la référence.

On a d'après la description de la méthode, l'expression de la commande sous la forme :

$$U = w + \left( \sum_{i=1}^4 \mu_i B_i \right)^{-1} \cdot (B \cdot r(t) - \left( \sum_{i=1}^4 (\mu_i A_i) - A_r \right) X_r) = \begin{bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \end{bmatrix} \quad (3.20)$$

avec,

$$w = - \sum_{j=1}^r \mu_j K_j e \begin{bmatrix} w_1 \\ w_2 \\ w_3 \\ w_4 \end{bmatrix} \quad (3.21)$$

L'expression ressemble à celle trouvée dans la poursuite de trajectoire et donc on doit passer par le calcul au niveau scalaire comme ça été fait précédemment.

On trouve,

$$U_2 = w_2 + \frac{\tilde{X}2}{b1} \quad (3.22)$$

$$U_3 = w_3 + \frac{\tilde{X}4}{b2} \quad (3.23)$$

$$U_4 = w_4 + \frac{\tilde{X}6}{b3} \quad (3.24)$$

avec,

$$\tilde{X} = B \cdot r(t) - \left( \sum_{i=1}^r (\mu_i A_i) - A_r \right) \cdot X_r = \begin{bmatrix} \tilde{X}1 \\ \tilde{X}2 \\ \tilde{X}3 \\ \tilde{X}4 \\ \tilde{X}5 \\ \tilde{X}6 \end{bmatrix} \quad (3.25)$$

On garde les mêmes gains des LMI comme cité précédemment.

On introduit au modèle de référence la consigne :

$$Y_r = \begin{bmatrix} \cos(t) + \cos(4t) \\ \sin(t) + \sin(4t) \\ 0.1 \sin(t) \end{bmatrix} \quad (3.26)$$

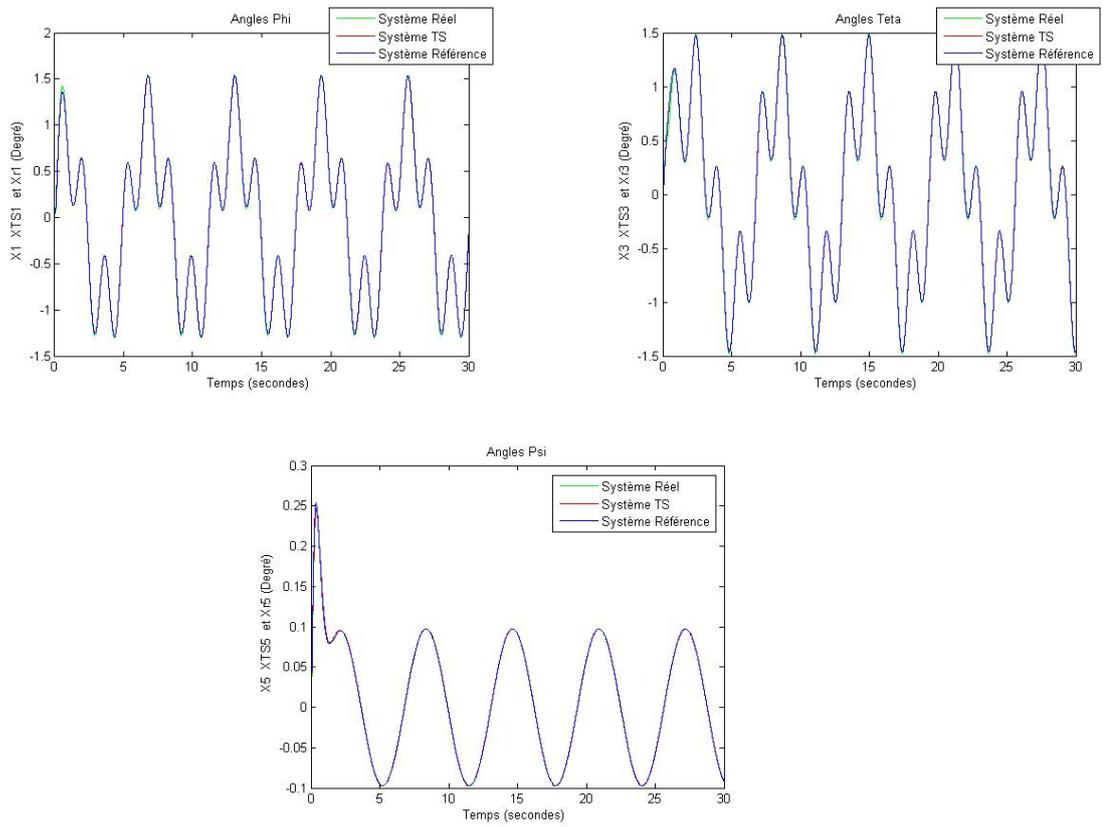


FIG. 3.1 : La sortie du système réel et TS

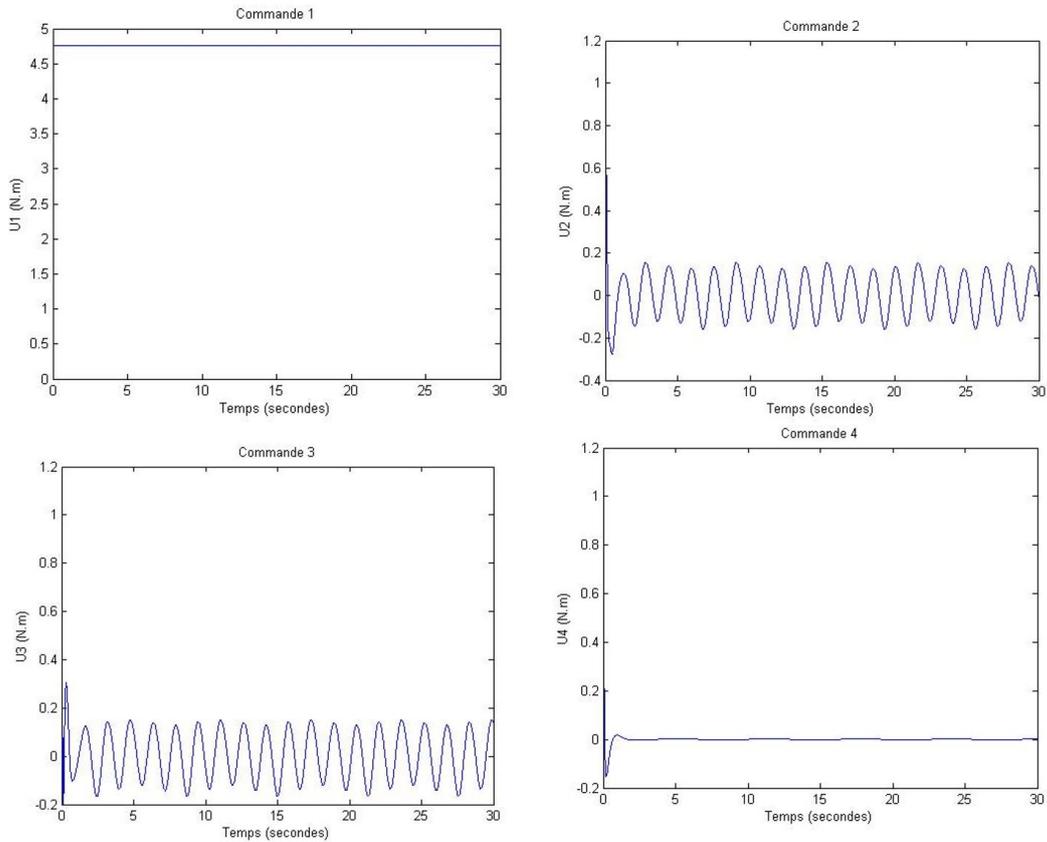


FIG. 3.2 : Les commandes U1, U2, U3 et U4

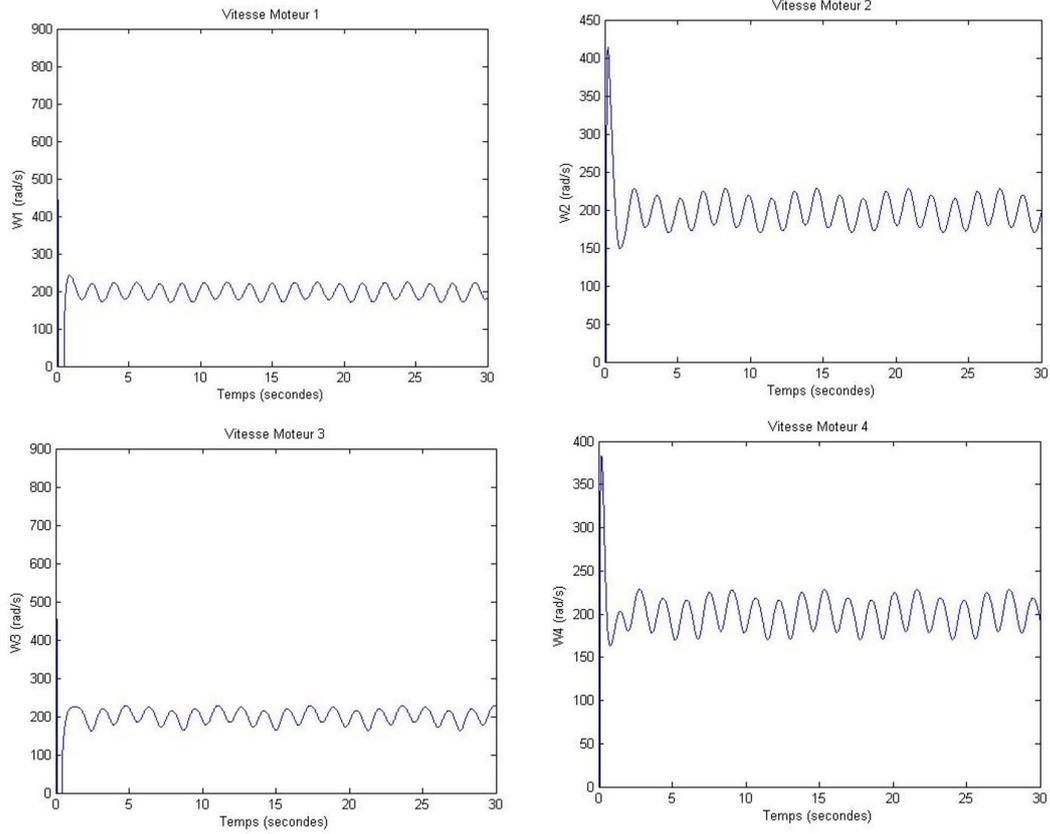


FIG. 3.3 : Les vitesses des moteurs

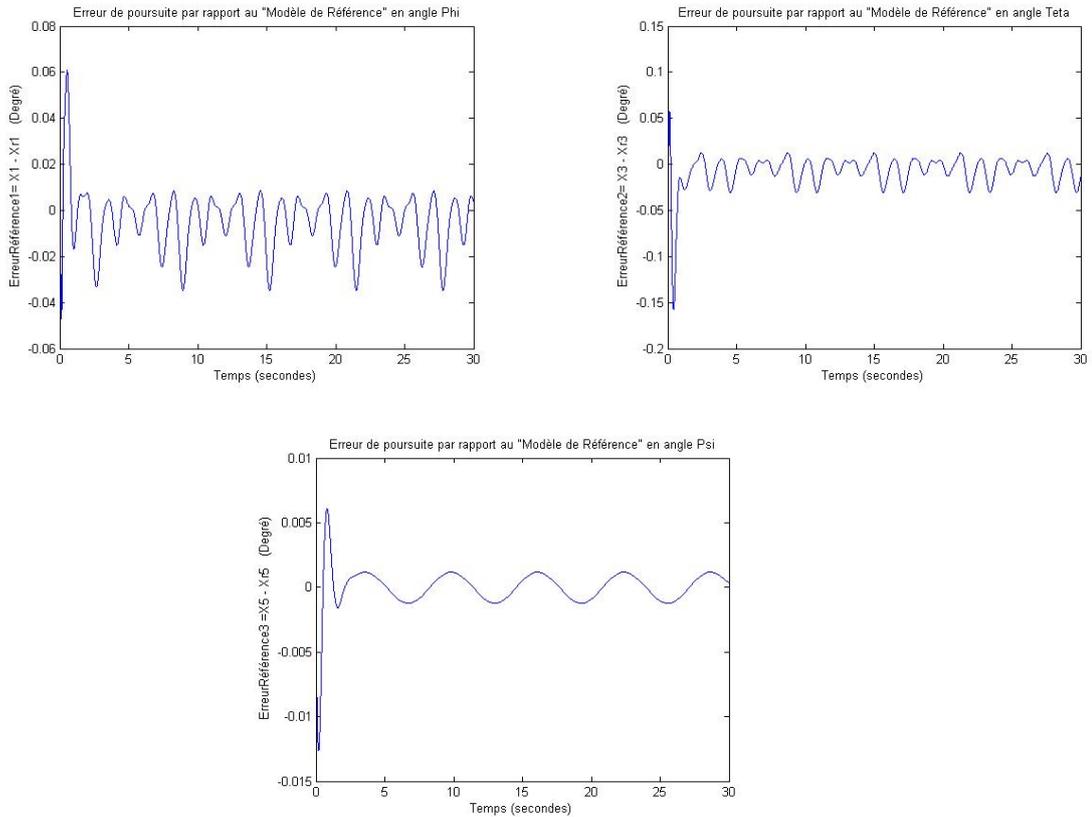


FIG. 3.4 : Erreur Modèle Référence

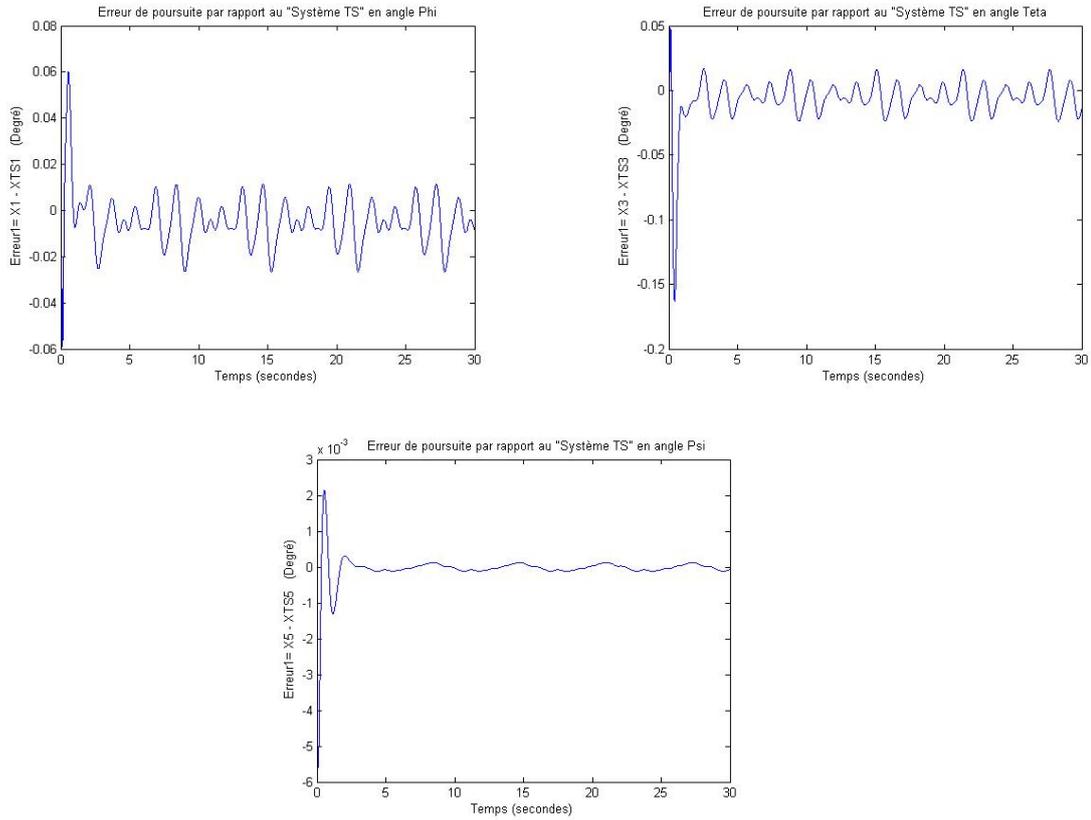


FIG. 3.5 : Les erreurs

Le sous-système des angles performe une poursuite vers la consigne imposée selon la trajectoire d'état que parcourt le modèle de référence ceci montre bien la validité de la méthode et son potentiel par rapport aux autres approches, on est dans le cas où on arrive à exploiter toutes les limites physiques du système pour poursuivre des consignes hyper rapides.

### 3.3.2 Sous-système de position

Selon la même manière précédente, on a la loi de commande suivante :

$$U = w^{xyz} + \left( \sum_{i=1}^2 \mu_i^{xyz} B_i^{xyz} \right)^{-1} (B^{xyz} \cdot r^{xyz}(t) - \left( \sum_{i=1}^2 (\mu_i^{xyz} A_i^{xyz}) - A_r^{xyz} \right) X_r^{xyz}) = \begin{bmatrix} U_x \\ U_y \\ U_z \end{bmatrix} \quad (3.27)$$

avec,

$$w^{xyz} = - \sum_{j=1}^2 \mu_j^{xyz} \cdot K_j^{xyz} \cdot e^{xyz} = \begin{bmatrix} w1^{xyz} \\ w2^{xyz} \\ w3^{xyz} \end{bmatrix} \quad (3.28)$$

On trouve,

$$U_x = w_1^{xyz} + m \cdot \tilde{X}_2^{xyz} \quad (3.29)$$

$$U_y = w_2^{xyz} + m \cdot \tilde{X}_4^{xyz} \quad (3.30)$$

$$U_z = U_1 = w_3^{xyz} + m. \left( \sum_{i=1}^r \mu_i^{xyz} . b_i \right)^{-1} \tilde{X}6^{xyz} \quad (3.31)$$

avec,

$$\tilde{X}^{xyz} = B^{xyz} . r^{xyz}(t) - \left( \left( \sum_{i=1}^2 (\mu_i^{xyz} A_i^{xyz}) - A_r^{xyz} \right) . X_r^{xyz} \right) = \begin{bmatrix} \tilde{X}1^{xyz} \\ \tilde{X}2^{xyz} \\ \tilde{X}3^{xyz} \\ \tilde{X}4^{xyz} \\ \tilde{X}5^{xyz} \\ \tilde{X}6^{xyz} \end{bmatrix} \quad (3.32)$$

### 3.3.3 Poursuite par modèle de référence en position (système global en cascade)

Après avoir validé le régulateur des angles et avoir trouvé celui de la position on se propose maintenant de poursuivre le modèle de position suivant :

$$\dot{X}_r^{xyz} = A_r^{xyz} . X_r^{xyz} + B_r^{xyz} . r^{xyz}(t) \quad (3.33)$$

$$\dot{X}_r^{xyz} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ -25.025 & -10.005 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & -25.045 & -10.009 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & -25.005 & -10.001 \end{bmatrix} X_r^{xyz} + B^{xyz} . K_i^{xyz} \int e^{xyz} . dt \quad (3.34)$$

avec,

$$e^{xyz} = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} - \begin{bmatrix} X_d \\ Y_d \\ Z_d \end{bmatrix} \quad (3.35)$$

La matrice  $A_r^{xyz}$  est exponentiellement stable et ayant des valeurs propres rapides :

$$\gamma_i^{xyz} = (-5 \quad -5.001 \quad -5.002 \quad -5.003 \quad -5.004 \quad -5.005) \quad (3.36)$$

Pour pouvoir retrouver le gain de l'action intégral, on impose les valeurs propres suivantes :

$$\gamma_i^{xyz} = (-5 \quad -5.001 \quad -5.002 \quad -5.003 \quad -5.004 \quad -5.005 \quad -0.02)$$

$$-0.02001 \quad -0.02002) \quad (3.37)$$

On trouve,

$$K_i^{xyz} = \begin{bmatrix} 0.2437 & 0 & 0 \\ 0 & 0.2433 & 0 \\ 0 & 0 & 0.2431 \end{bmatrix} \quad (3.38)$$

On connecte l'ensemble des sous-systèmes en cascade et on injecte les lois de commandes trouvées avec le système de référence en temps réel. Pour le sous-modèle des angles on garde les mêmes paramètres de la région LMI, par contre pour le sous-modèle de la position on choisi les valeurs suivantes :  $\alpha^{xyz} = -0.7$ ,  $\beta^{xyz} = 0$  nous donnant les valeurs des gains :

$$K1^{xyz} = K2^{xyz} = \begin{bmatrix} 1.3982 & 1.8287 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1.3982 & 1.8287 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1.3983 & -1.8289 \end{bmatrix} \quad (3.39)$$

La consigne est :

$$Y_r^{xyz} = \begin{bmatrix} \cos(0.5t) \\ \sin(0.5t) \\ t \end{bmatrix} \quad (3.40)$$

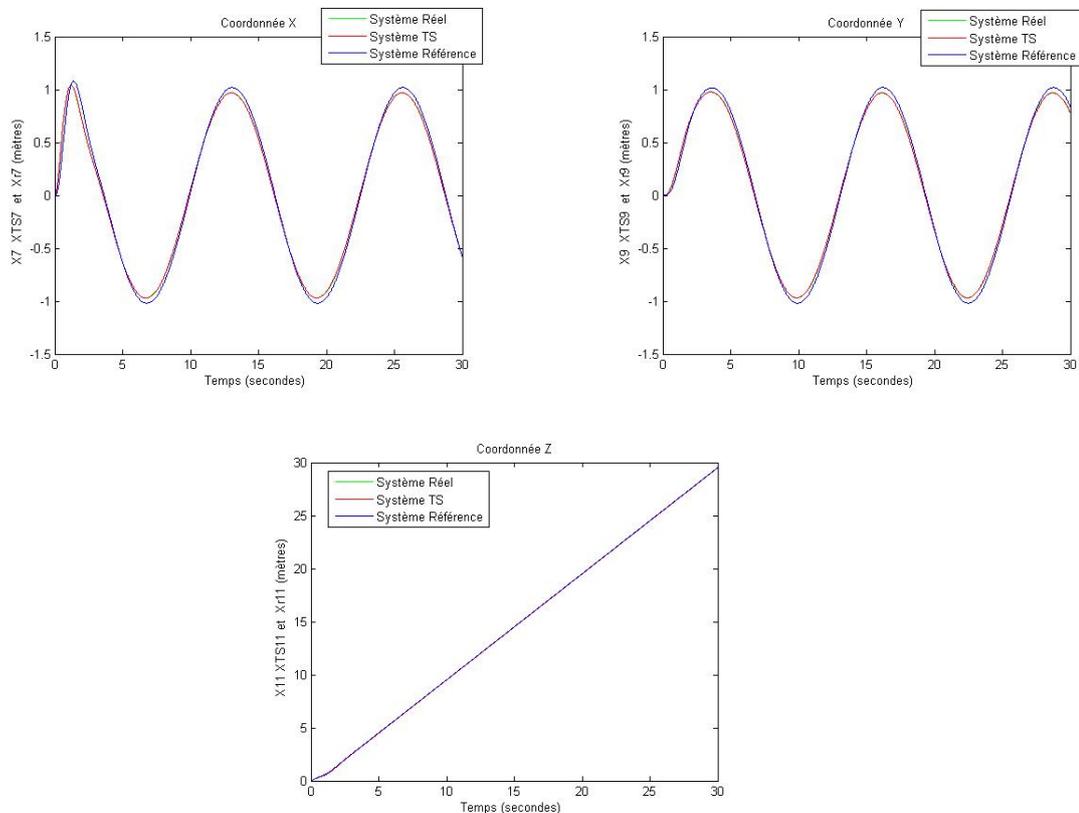


FIG. 3.6 : La sortie du système réel et TS

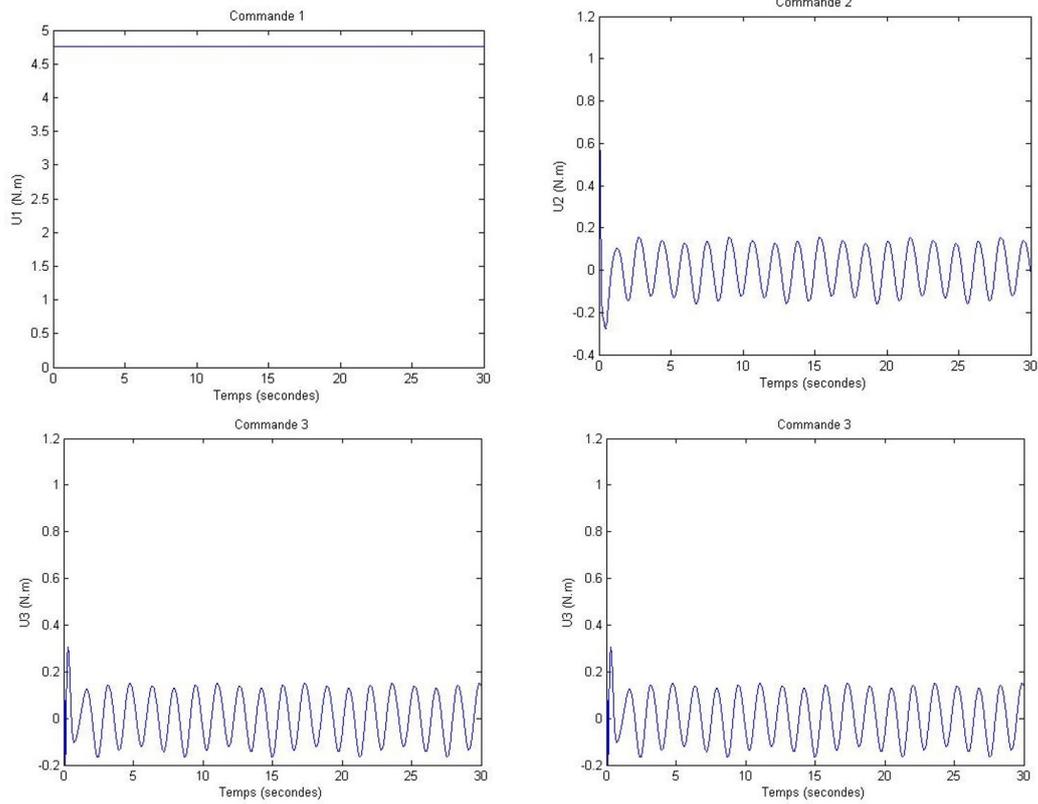


FIG. 3.7 : Les commandes  $U1$ ,  $U2$ ,  $U3$  et  $U4$

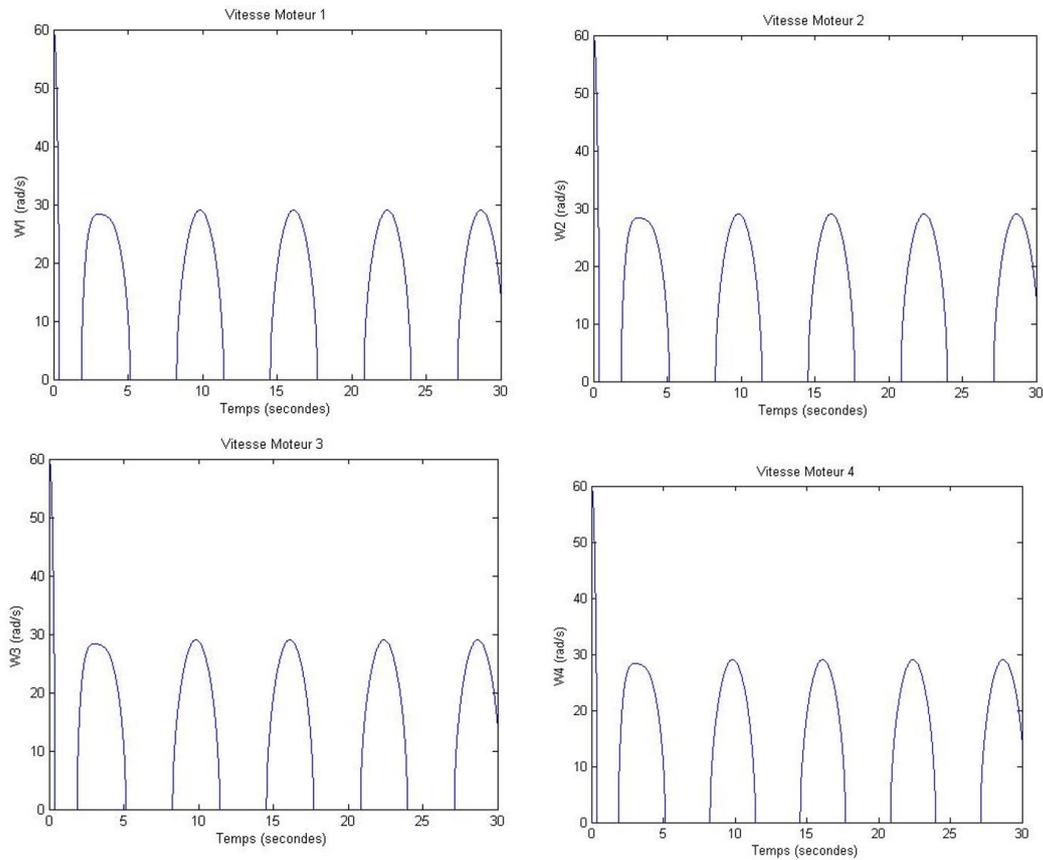


FIG. 3.8 : Les vitesses des moteurs

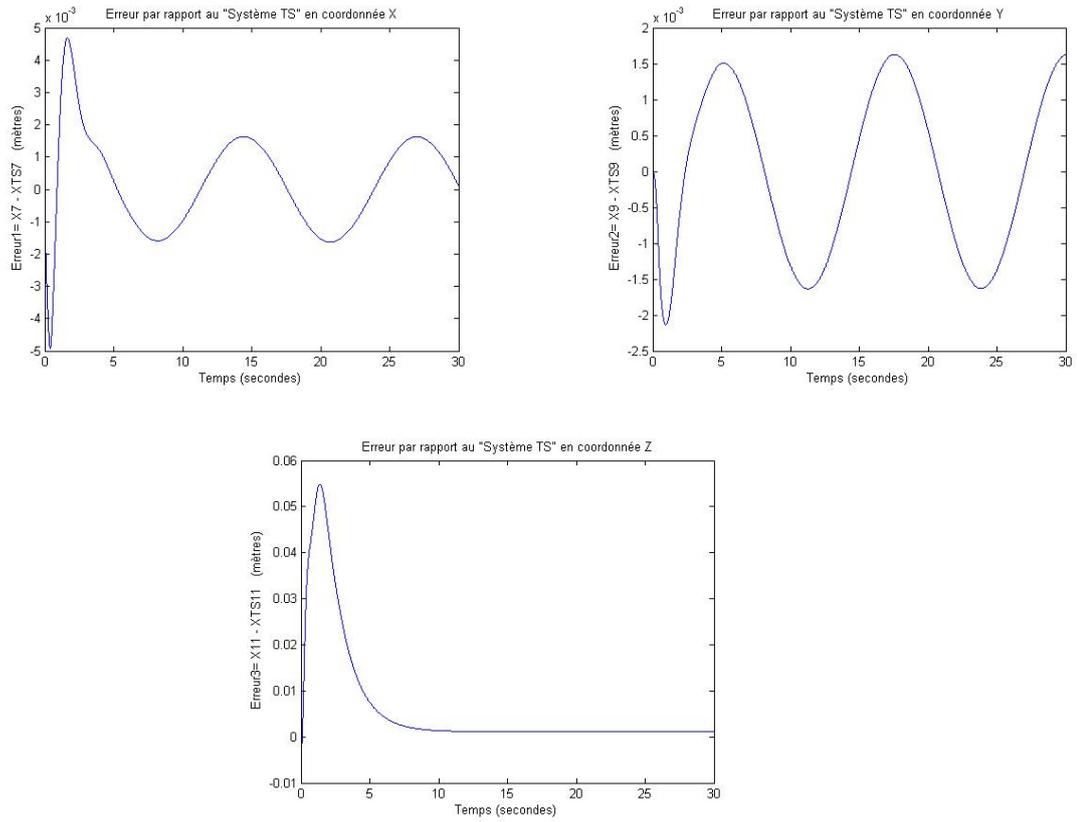


FIG. 3.9 : Les erreurs

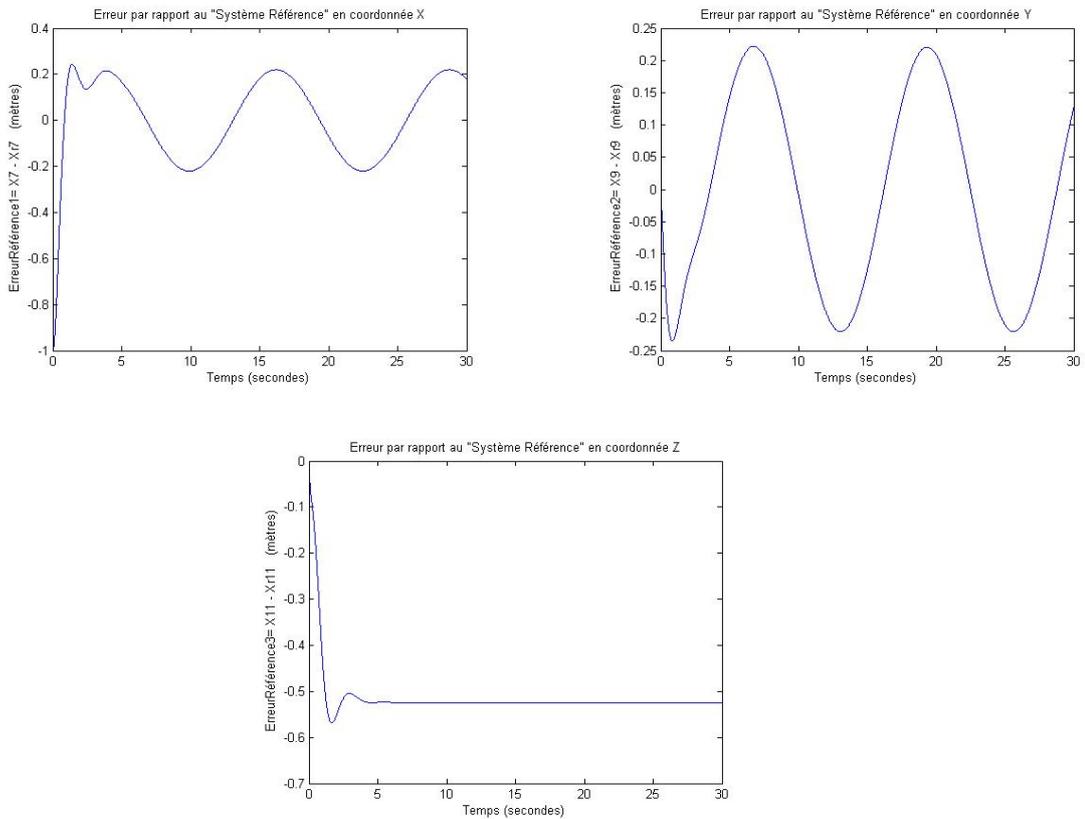


FIG. 3.10 : Erreur Modele de Référence

Les résultats présentent une coïncidence entre le TS et le réel, les deux états performent une poursuite quasi-égale vers la trajectoire des états que parcourt l'état du modèle de référence avec une légère erreur de poursuite due à la rapidité d'évolution de la consigne par rapport aux limites physique du quadri-rotor.

La commande implémentée arrive à donner de bonnes performances et meilleures que celles des sections précédentes grâce à la logique floue que porte la théorie du TS dont on peut rendre le système à se comporter comme un modèle parfait.

### 3.4 Comparaison avec l'approche de poursuite de trajectoire

Il vient que la méthode du modèle de référence a donné de meilleurs résultats par rapport à ceux de la poursuite de trajectoire précédente (2.4.2), il est plus judicieux maintenant d'essayer de valider l'approche en faisant une comparaison récapitulative avec les méthodes utilisées précédemment, pour se faire on opte à utiliser les mêmes paramètres des régulateurs PDC et le même modèle de référence.

On reprend donc la référence :

$$Y_r^{xyz} = \begin{bmatrix} \cos(0.5t) \\ \sin(0.5t) \\ t \end{bmatrix} \quad (3.41)$$

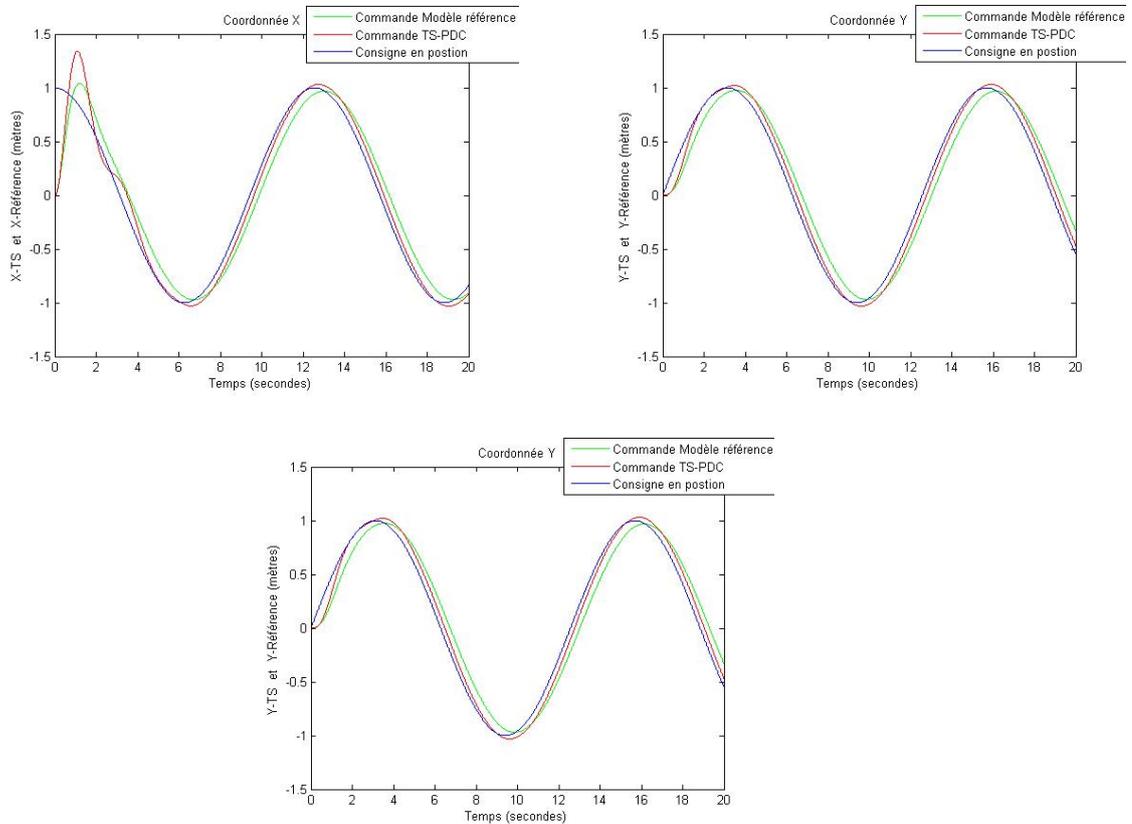


FIG. 3.11 : La sortie du système réel et TS

L'approche du modèle de référence arrive à poursuivre la consigne selon une manière plus précise que celle de la poursuite de trajectoire par PDC, ceci est une conséquence immédiate du fait que le système réel parcourt la même trajectoire d'état d'un système linéaire parfait ayant une dynamique améliorée que celle du système TS en boucle fermée par PDC. Le modèle de référence demeure donc une approche valide dans un cas plus général où on se retrouve avec des systèmes non linéaires plus compliqués que le quadri-rotor dont la synthèse d'une loi de commande classique n'est pas si simple vu les non linéarités qui s'opposent. La procédure satisfait donc le cahier de charge des performances, d'évolution des signaux de commande et également dans la l'implémentation de l'approche (simplicité de l'aspect linéaire).

### 3.5 Conclusion

Le présent chapitre était une alternative pratique aux techniques de commande des systèmes non linéaires complexes, la théorie du TS vient de montrer son avantage, cette aptitude nous offre la possibilité de réaliser des commandes plus facilement qui était un défi bien illustré lors l'implémentation dans le quadrirotor selon une navigation en trajectoire hélicoïdale. Pour finaliser une loi de commande par modèle de référence a été employée pour poursuivre les consignes selon la dynamique désirée, une étape qui a pu montrer le potentiel majeur par rapport aux techniques classiques et qui a pu exploiter toutes les caractéristiques physiques du quadri-rotor. Pour finaliser une comparaison récapitulative a été faite, nous avons conclu sur le tau de précision de la procédure du modèle de référence par rapport aux commandes précédentes. Dans le prochain chapitre, une commande optimale floue sera employée sur le modèle TS linéaire.

# Chapitre 4

## Commande prédictive floue optimale

## 4.1 Introduction

Les procédés industriels ont connu une large progression technique et c'est du aux besoins et exigences que stipule l'industrie d'aujourd'hui, cette progression nécessite éventuellement des techniques d'asservissement plus compacts, la régulation des processus exige maintenant des contraintes sur le comportement dynamique et également sur les ressources économiques.

La commande prédictive « Model Predictive Control » est l'une des approches rentrant dans l'axe des commandes optimales qui permettent de satisfaire les exigences sur la boucle fermée, le choix de cette technique est du à ses performances spectaculaires illustrées dans [7] et [9].

L'objet est de faire une hybridation entre les avantages que nous fournit la modélisation floue TS et les performances que stipule la commande prédictive afin de pouvoir les comparer avec tous ce qu'on vient de détailler au long de ce mémoire , pour ceci nous optons dans ce chapitre d'implémenter la commande MPC sur la représentation TS équivalente au quadri-rotor.

La méthode se diffère selon deux versions non linéaire et linéaire, pour ce présent cas nous nous intéresserons par le coté linéaire puisque la représentation TS étant purement linéaire. Il est impératif dans un premier lieu de démontrer la source du MPC afin de pouvoir l'implémenter dans le modèle réel.

Comme toute les approches optimales, ca se base sur la minimisation d'un critère le plus souvent quadratique dont lequel on décrit des contraintes sur les grandeurs physique du système ainsi sur les actionneurs, parmi les commandes optimales qui interprètent rigoureusement cette aspect est la commande « LQR » [19] ou le critère quadratique dans un système continu donné se formule comme ceci :

$$J = X^T Q X + U^T R U \quad (4.1)$$

Avec  $Q$  et  $R$  sont des matrices diagonales ayant des pondérations qui permettent d'ajuster le tau d'économie d'énergie sur l'état.

La minimisation du critère  $J$  étant un problème quadratique qui engendre la déduction d'une commande satisfaisant les exigences décrites par le critère.

La commande MPC est formulée selon la meme manière que celle-ci sauf ici nous nous intéressons sur le compromis entre « Performances/Economie d'énergie de l'actionneur » .

La commande prédictive qui sera utilisée étant discrète et qui nécessite discrétisation de la représentation du système continu.

Ca se fait sur « Matlab » par la fonction « c2dm » qui prend comme paramètres les matrices de l'espace d'état linéaire et le temps d'échantillonnage désiré, pour notre cas de figure la simulation sera établie par l'ODE « Koutta Runge » qu'on vient de l'utiliser au bout des chapitres précédents, pour cela on opte à utiliser le meme temps de discrétisation qui est  $t=0.01$  s.

## 4.2 Elaboration de l'espace d'état augmenté

L'approche se base sur la création d'un espace d'état augmenté qui possède les variations des états et des sorties comme étant les variables d'états :

Soit un espace d'état donné dans  $\mathbb{R}^n$  et ayant  $p \leq n$  sorties et qui a subit une discrétisation

selon un temps d'échantillonnage donné  $t$  :

$$\begin{cases} X_m(k+1) &= A_m \cdot X_m + B_m \cdot U_m(k) \\ Y_m(k+1) &= C_m \cdot X_m(k+1) \end{cases} \quad (4.2)$$

On établit la différence entre deux états successifs :

$$X_m(k+1) - X_m(k) = A_m \cdot (X(k) - X_m(k-1)) + B_m \cdot (U(k) - U(k-1)) \quad (4.3)$$

On note donc,

$$\Delta X_m(k+1) = X_m(k+1) - X_m(k) \quad \text{et} \quad \Delta U(k+1) = U(k) - U(k-1) \quad (4.4)$$

Ce qui permet d'écrire,

$$\Delta X_m(k+1) = A_m \cdot \Delta X_m(k) + B_m \cdot \Delta U(k) \quad (4.5)$$

De même pour la sortie,

$$y_m(k+1) - y_m(k) = C_m(X_m(k+1) - X(k)) = C_m \cdot \Delta X_m(k+1) \quad (4.6)$$

On explicite,

$$y_m(k+1) - y_m(k) = C_m \cdot A_m \cdot \Delta X_m(k) + C_m \cdot B_m \cdot \Delta U(k) \quad (4.7)$$

Afin de générer l'espace d'état augmenté on prend  $X = \begin{bmatrix} \Delta X_m \\ y_m \end{bmatrix}$  comme étant la nouvelle variable d'état, et par conséquent :

$$X(k+1) = \begin{bmatrix} A_m & O_m^T \\ C_m \cdot A_m & I_p \end{bmatrix} \cdot \begin{bmatrix} \Delta X_m \\ y_m \end{bmatrix} + \begin{bmatrix} B_m \\ B_m \cdot C_m \end{bmatrix} \cdot \Delta U(k) \quad (4.8)$$

On note aussi la sortie augmentée de la forme :

$$y(k+1) = [O_m \quad I_p] \cdot \begin{bmatrix} \Delta X_m(k+1) \\ y_m(k+1) \end{bmatrix} \quad (4.9)$$

Avec  $O_m$  est un vecteur colonne des zéros de dimension  $p$  alors que  $I_p$  étant une matrice diagonale unitaire de dimension  $p \times p$ .

En posant,

$$A = \begin{bmatrix} A_m & O_m^T \\ C_m \cdot A_m & I_p \end{bmatrix}, \quad B = \begin{bmatrix} B_m \\ C_m \cdot B_m \end{bmatrix} \quad \text{et} \quad C = [O_m \quad I_p] \quad (4.10)$$

On se retrouve avec le nouvel espace d'état augmenté :

$$\begin{cases} X(k+1) = A \cdot X(k) + B \cdot \Delta U(k) \\ Y(k+1) = C \cdot X(k+1) \end{cases} \quad (4.11)$$

Le système augmenté prend en compte la variation entre deux états successifs ainsi l'évolution instantanée de la sortie engendrée par la variation en signal de commande ceci est indispensable pour pouvoir établir l'horizon de prédiction une question qui sera traitée dans la prochaine section.

### 4.3 Prédiction des états futurs

Au moyen de l'information sur l'état  $X(k_i)$ ,  $i = 1, 2, \dots$  on s'intéresse à prédire le comportement dynamique qui se caractérise par le nombre  $N_p$  désignant le nombre des instants d'échantillonnage qui s'enchainent dans un horizon de prédiction donné pour savoir l'évolution de l'état augmenté.

$$X(k_i), X(k_i + 1 / k_i), X(k_i + 2 / k_i), \dots \dots X(k_i + N_p - 1 / k_i) \quad (4.12)$$

De même l'optimisation dynamique s'effectue en tenant compte un horizon de control qui sert à prédire la trajectoire de control dont les signaux de commandes varient, cet horizon de control se caractérise par un nombre  $N_c$  indiquant le nombre des instants d'échantillonnage qui se succèdent afin d'avoir l'information sur les variations futures :

$$\Delta U(k_i), \Delta U(k_i + 1), \Delta U(k_i + 2), \dots \dots \Delta U(k_i + N_c - 1) \quad (4.13)$$

En se basant sur l'état actuel, les états futurs se calculent séquentiellement en prenant :

$$\begin{aligned} X(k_i + 1/k_i) &= AX(k_i) + B\Delta U(k_i) \\ X(k_i + 2/k_i) &= AX(k_i + 1) + B\Delta U(k_i + 1) \\ X(k_i + 3/k_i) &= A^2X(k_i) + AB\Delta U(k_i) + B\Delta U(k_i + 1) \\ &\dots \\ &\dots \\ &\dots \\ X(k_i + N_p/k_i) &= A^{N_p}X(k_i) + A^{N_p-1}B\Delta U(k_i) + A^{N_p-2} \cdot B \cdot \Delta U(k_i + 1) + \dots \end{aligned}$$

$$\dots + A^{Np-Nc}B\Delta U(k_i + Nc - 1) \quad (4.14)$$

Les sorties futures sont donc :

$$\begin{aligned} y(k_i + 1/k_i) &= CAX(k_i) + CB\Delta U(k_i) \\ y(k_i + 2/k_i) &= CAX(k_i + 1) + CB\Delta U(k_i + 1) \\ y(k_i + 3/k_i) &= CA^2X(k_i) + CAB\Delta U(k_i) + CB\Delta U(k_i + 1) \\ &\dots \\ &\dots \\ &\dots \\ y(k_i + Np/k_i) &= CA^{Np}X(k_i) + CA^{Np-1}B\Delta U(k_i) + CA^{Np-2} \cdot B \cdot \Delta U(k_i + 1) + \\ &\dots + CA^{Np-Nc}B\Delta U(k_i + Nc - 1) \end{aligned} \quad (4.15)$$

Les sorties sont toutes formulées en s'appuyant sur la mesure de l'état actuel  $X(k_i)$  engendré par la variation instantanée du signal  $\Delta U(k_i)$ .

Afin d'élargir encore le concept de la prédiction on introduit les deux vecteurs qu'on utilise pour élaborer la loi de commande instantanée par MPC.

Soit :

$$\Upsilon = \begin{bmatrix} y(k_i + 1/k_i) \\ y(k_i + 2/k_i) \\ \vdots \\ y(k_i + Np/k_i) \end{bmatrix} \quad (4.16)$$

$$\Delta\mu = \begin{bmatrix} \Delta U(k_i) \\ \Delta U(k_i + 1) \\ \vdots \\ \Delta U(k_i + Nc - 1) \end{bmatrix} \quad (4.17)$$

Avec  $\Upsilon$  étant un vecteur ligne de dimension  $pxN_p$  et  $\Delta\mu$  est un vecteur ligne ayant  $mxN_c$  comme dimension ou  $m$  désignant le nombre de commandes du système réel.

Si on éclate le vecteur  $\Upsilon$  ca nous résulte un espace d'état de prédiction ayant les variations séquentielles des espaces d'états augmentés prenant la structure suivante :

$$\Upsilon = F \cdot X(k_i) + \phi \cdot \Delta\mu \quad (4.18)$$

avec,

$$F = \begin{bmatrix} CA \\ CA^2 \\ \vdots \\ CA^{Np} \end{bmatrix} \quad (4.19)$$

$$\phi = \begin{bmatrix} C \cdot B & 0 & \dots & 0 \\ C \cdot A \cdot B & C \cdot B & \dots & 0 \\ C \cdot A^2 \cdot B & C \cdot A \cdot B & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ C \cdot A^{Np-1} \cdot B & C \cdot A^{Np-2} \cdot B & \dots & C \cdot A^{Np-Nc} \cdot B \end{bmatrix} \quad (4.20)$$

L'espace d'état de prédiction obtenu obéit aux variations instantanées des états futures par l'influence de la variation des signaux de commandes futures en se basant sur la connaissance de l'état actuel, l'horizon de prédiction fourni une information sur l'évolution des signaux dans  $Np$  instants ce qui permet de décider sur l'action que doit le signal  $\Delta U$  produire afin de satisfaire un objet de commande donné.

Dans la prochaine section, on s'intéresse à exploiter cette prédiction afin d'optimiser le critère et puis d'élaborer l'expression finale de la commande prédictive.

## 4.4 Optimisation dynamique dans l'horizon de prédiction

Le concept de la commande optimale repose sur la satisfaction d'un cahier de charge renseignant sur les contraintes qu'on désire avoir sur le système afin de produire un comportement donné ou bien d'atténuer la consommation des ressources données, la satisfaction de telles exigences nécessitent l'optimisation d'un critère, pour le présent cas on se donne une optimisation dynamique qui se fait à travers un horizon de prédiction sur  $Np$  instants.

Le critère de la prédiction prend l'expression suivante :

$$J_{mpc} = (R_s - \Upsilon)^T \cdot (R_s - \Upsilon) + \Delta\mu^T \cdot R \cdot \Delta\mu \quad (4.21)$$

Ou  $R_s$  étant le vecteur ligne du signal de référence sur  $Np$  instants d'échantillonnage ayant comme dimension  $p \times Np$  qui prend la forme

$$R_s = \begin{pmatrix} \bar{R}s_1 \\ \bar{R}s_2 \\ \vdots \\ \bar{R}s_{Np} \end{pmatrix} \text{ avec } \bar{R}_{si} = \begin{pmatrix} Yr_1 \\ Yr_2 \\ \vdots \\ Yr_p \end{pmatrix}, \quad i = 1, 2 \dots Np \quad (4.22)$$

Et  $R$  étant une matrice carrée diagonale de dimension  $(m \times N_c) \times (m \times N_c)$  possédant les pondérations sur les signaux de commande et permettant de contraindre leur évolution.

Le critère  $J_{mpc}$  comporte l'erreur quadratique entre la consigne et la sortie ainsi l'énergie des signaux de commande pondérés au bout de tout l'horizon de prédiction.

La minimisation de ce critère engendre la déduction d'une loi de commande  $\Delta\mu$  qui fait annuler l'erreur quadratique tout en préservant les ressources en fonction des paramètres de  $R$ .

La minimisation s'établi en prenant la dérivée nulle par rapport à la commande :

$$\frac{dJ_{mpc}}{d\Delta\mu} = 0 \quad (4.23)$$

Afin de calculer le  $\Delta\mu$  on utilise l'expression du critère explicite :

$$\begin{aligned} J_{mpc} = & (R_s - F \cdot X(k_i))^T \cdot (R_s - F \cdot X(k_i)) - 2 \cdot \Delta\mu^T \cdot \phi^T \cdot (R_s - F \cdot X(k_i)) \\ & + \Delta\mu^T \cdot (\phi^T \cdot \phi + R) \cdot \Delta\mu \end{aligned} \quad (4.24)$$

On dérive donc,

$$\frac{dJ_{mpc}}{d\Delta\mu} = -2 \cdot \phi^T \cdot (R_s - F \cdot X(k_i)) + 2 \cdot (\phi^T \cdot \phi + R) \cdot \Delta\mu \quad (4.25)$$

On déduit par (4.23) l'expression du vecteur des commandes optimales,

$$\Delta\mu = (\phi^T \cdot \phi + R)^{-1} \cdot \phi^T \cdot (R_s - F \cdot X(k_i)) \quad (4.26)$$

L'expression du vecteur des variations de commande est issu par l'erreur entre le vecteur de consigne et le produit  $F \cdot X(k_i)$  qui indique les sorties séquentielles pondérées par la matrice  $R$  ce qui qualifie bien le caractère de la poursuite.

Le vecteur  $\Delta\mu$  contient les variations des commandes tout au long de l'horizon de control :

$$\Delta U(k_i + j), \quad j = 0, 1, 2, \dots, N_c - 1 \quad (4.27)$$

Par le principe de l'horizon résident dans [21], on nous informe que seulement les variations premières  $k_i + 1$  seront prises en compte.

Il est important de mentionner qu'on vient de retrouver l'expression des variations séquentielles de commandes qui sont issues par la transition entre un état réel vers l'état qui le suit, en réalité et c'est ce qui va être abordé dans l'implémentation, on utilise la superposition instantanée de chaque commande comme suit :

$$U(k_{i+1}) = U(k_i) + \Delta U(k_{i+1}) \quad (4.28)$$

A noter aussi, que le choix des pondérations influe sur les performances, à chaque fois on utilise des paramètres importants on est en train d'imposer un caractère plus économique sur les actionneurs les performances de poursuite ont tendance donc de diminuer et vis-versa, il faut donc prendre en considération des valeurs qui satisfassent un compromis entre ces deux exigences.

## 4.5 Stabilité du système en boucle fermée

On s'intéresse d'expliciter la nature de la commande MPC en analysant l'impact que fait sur le système en boucle fermée.

Selon le principe de l'horizon résident, seulement les commandes à l'instant  $k_i + 1$  seront prises en comptes ce qui en résulte,

$$\Delta U(k_i + 1) = \left( [I_m \quad 0_m \quad \cdots \quad 0_m]^{Nc} \cdot (\phi^T \cdot \phi + R) \right)^{-1} \cdot \phi^T \cdot (R_s - F \cdot X(k_i)) \quad (4.29)$$

Avec  $I_m$  est une matrice d'identité carrée de dimension  $m \times m$  et  $0_m$  étant une matrice carrée des zéros de même dimension. Le symbole  $Nc$  indique que le vecteur possède  $Nc$  éléments.

Ce qui donne,

$$\Delta U(k_i + 1) = (\phi^T \cdot \phi + R)^{-1} \cdot \phi^T \cdot ((\overline{R}_{s1} - F \cdot X(k_i))) \quad (4.30)$$

On cherche maintenant à expliciter cette expression, on distribue donc,

$$\Delta U(k_i + 1) = (\phi^T \cdot \phi + R)^{-1} \cdot \phi^T \cdot \overline{R}_{s1} - (\phi^T \cdot \phi + R)^{-1} \cdot \phi^T \cdot F \cdot X(k_i) \quad (4.31)$$

Le terme  $(\phi^T \cdot \phi + R)^{-1}$  est purement constant ce qui permet de désigner,

$$\Delta U(k_i + 1) = K_{\text{poursuite}} \cdot \overline{R}_{s1} - K_{\text{stab}} \cdot X(k_i) \quad (4.32)$$

On aperçoit que l'expression de la commande repose sur deux principaux termes un terme qui se manifeste par un retour d'état classique stabilisant et un terme servant à faire la poursuite.

D'après ce qu'on a détaillé dans les chapitres de la poursuite par PDC ainsi dans [18], la commande MPC étant une approche rationnelle qui explique bien son comportement en s'appuyant sur la stabilisation et la poursuite à travers des gains  $K_{\text{poursuite}}$  et  $K_{\text{stab}}$  qui se résultent par la minimisation du critère décrivant le compromis de la poursuite et l'économie de puissance.

Les performances de poursuite et de stabilisation vont dépendre éventuellement aux pondérations choisies dans  $R$  et qui impactent directement sur les gains.

Pour un espace d'état augmenté donné,

$$X(k_i + 1) = A \cdot X(k_i) + B \cdot \Delta U(k_i + 1) \quad (4.33)$$

On a le système en boucle fermée,

$$X(k_i + 1) = A \cdot X(k_i) - B \cdot K_{\text{stab}} \cdot X(k_i) + B \cdot K_{\text{poursuite}} \cdot \overline{R}_{s1} \quad (4.34)$$

Par conséquent,

$$X(k_{i+1}) = (A - B \cdot K_{stab}) \cdot X(k_i) + B \cdot K_{poursuite} \cdot \overline{R_{s1}} \quad (4.35)$$

Le comportement de la commande prédictive ressemble nettement à l'approche du retour d'état avec action compensation dans [3].

La dynamique du système en boucle fermée sera régulée donc par le gain  $K_{stab}$  selon les valeurs propres du polynôme caractéristique de la boucle fermée, tandis que la précision de la poursuite sera ajustée par le gain  $K_{poursuite}$ . Le seul inconvénient de la méthode réside sur le fait qu'on est en train de calculer ces commandes à chaque instant d'échantillonnage  $k_i + 1, k_i + 2, \dots, k_i + Np$ , ce qui nécessite un calculateur numérique ayant une puissance de calcul importante afin d'achever ces calculs matricielles dans un temps inférieur ou égale au temps d'échantillonnage.

Dans la section finale venant, on s'oriente d'implémenter la commande MPC sur le modèle équivalent TS au modèle non linéaire du quadri-rotor afin d'illustrer et de comparer les performances qui en résultent avec les procédures précédentes.

## 4.6 Commande prédictive floue T-S du quadri-rotor

La démarche repose sur l'application d'une commande prédictive à chaque tope d'horloge au modèle TS obtenue par la méthode des secteurs non linéaire TS, ca permet de calculer les gains nécessaires pour la stabilisation et la poursuite et puis au moyens des variations réelles on fournit les commandes naturelles calculées par le principe de l'horizon résident selon cette structure :

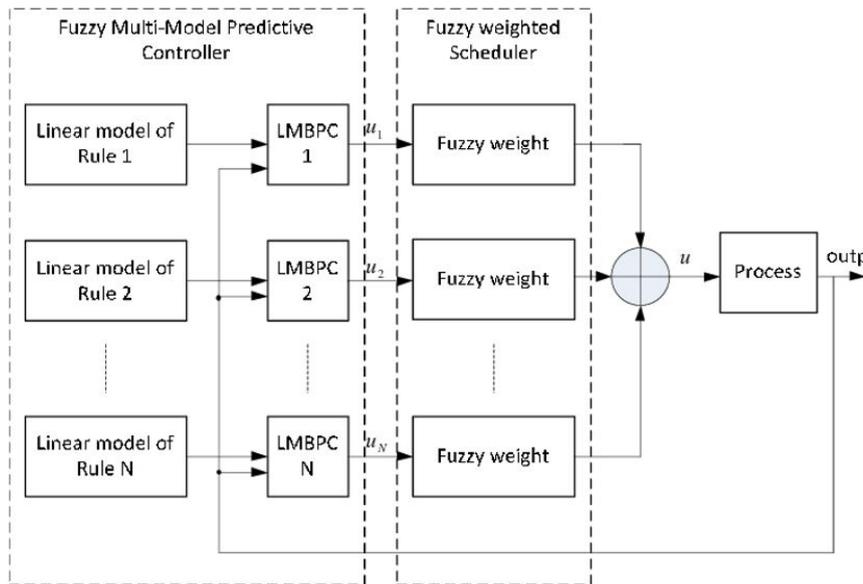


FIG. 4.1 : Structure d'asservissement par commande TS-MPC linéaire

Lorsqu'on tient en compte le compromis entre la performance et la consommation d'énergie, et après avoir tenté plusieurs paramètres, les résultats les plus satisfaisants ont été engendrés

par le moyen de ces valeurs :

$$Np = 50, \quad Nc = 15, \quad T_{\text{échantillonnage}} = 0.01 \text{ s}, \quad R = 0.1 \quad (4.36)$$

Les paramètres seront utilisés pour les régulateurs des sous-systèmes.

On se propose donc de poursuivre la référence dans l'espace  $3D$  :

$$Y_r = \begin{pmatrix} \cos(0.5t) \\ \sin(0.5t) \\ t \end{pmatrix} \quad (4.37)$$

Les conditions initiales en états et en commandes sont toutes nulles.

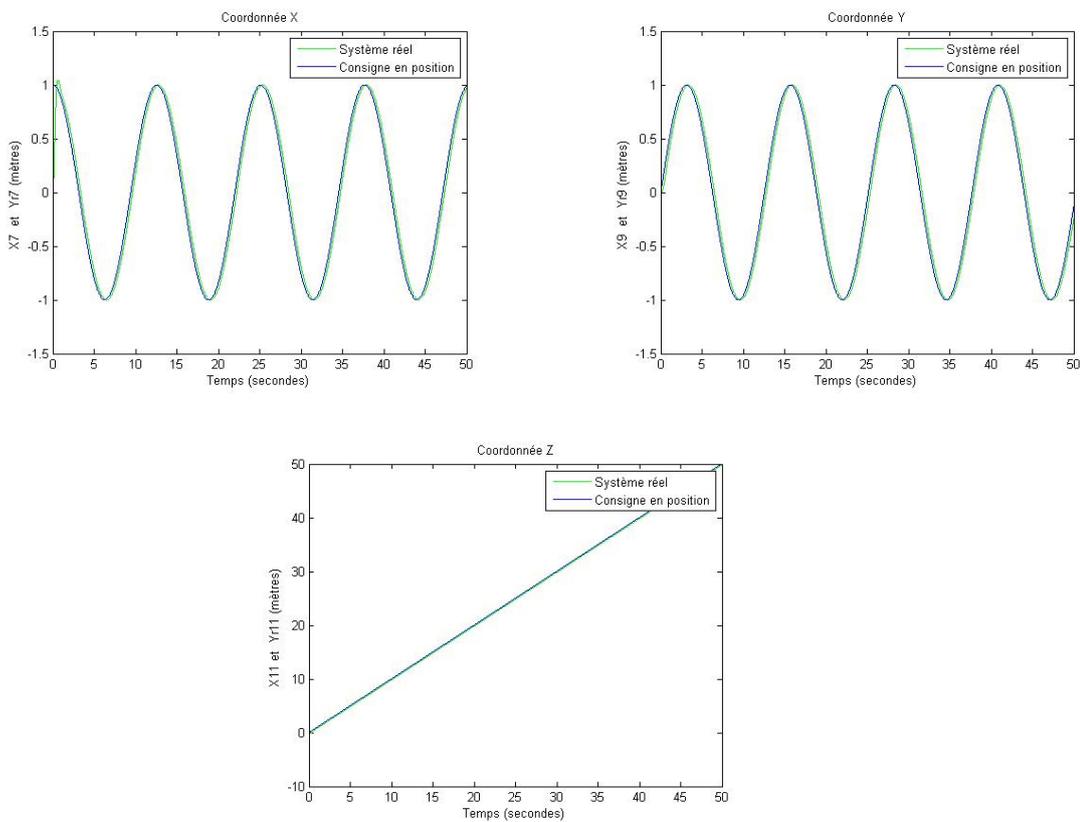


FIG. 4.2 : La sortie du système réel et TS

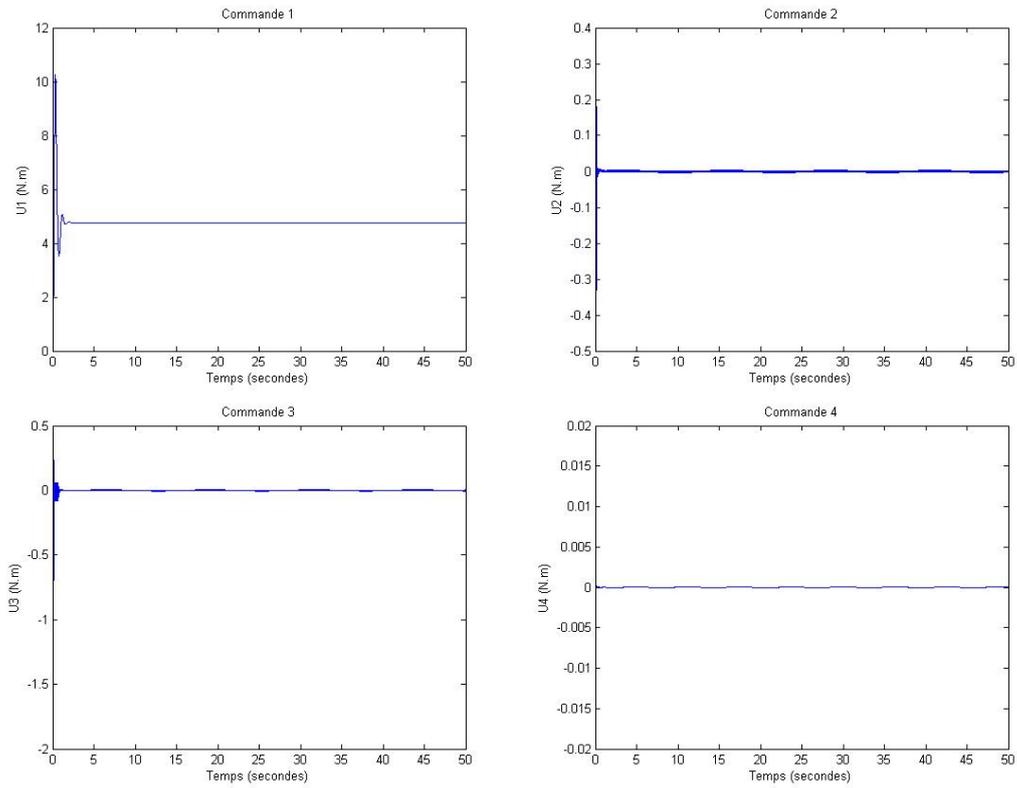


FIG. 4.3 : Les commandes U1, U2, U3 et U4

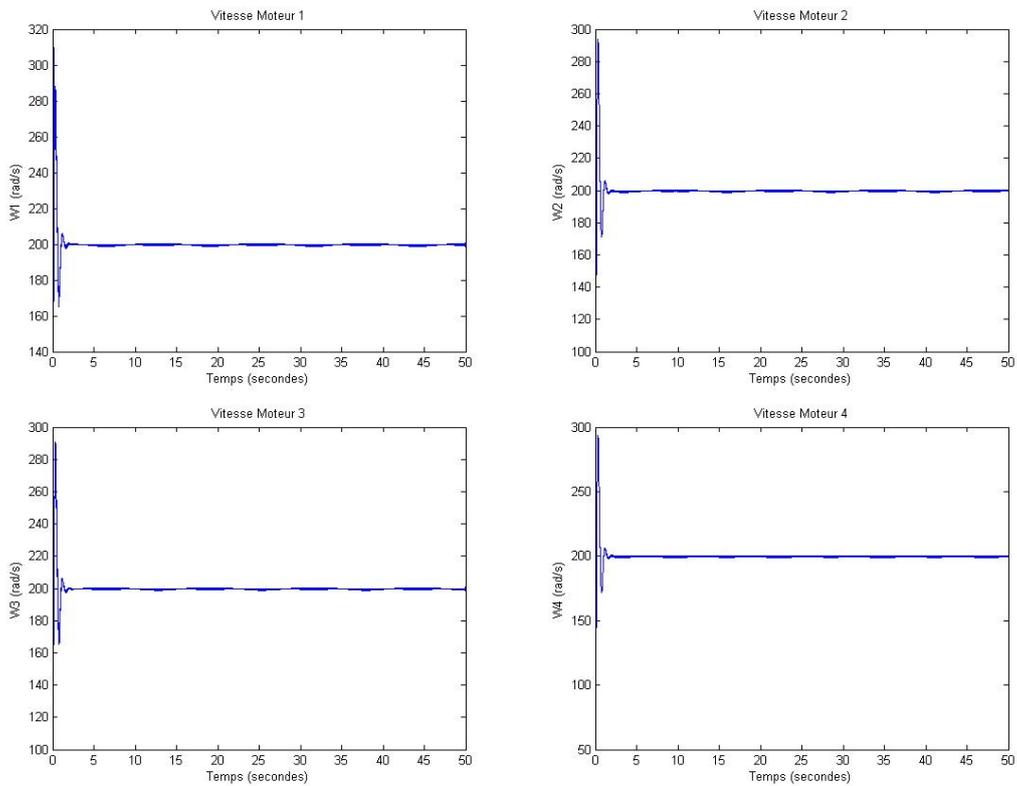


FIG. 4.4 : Les vitesses

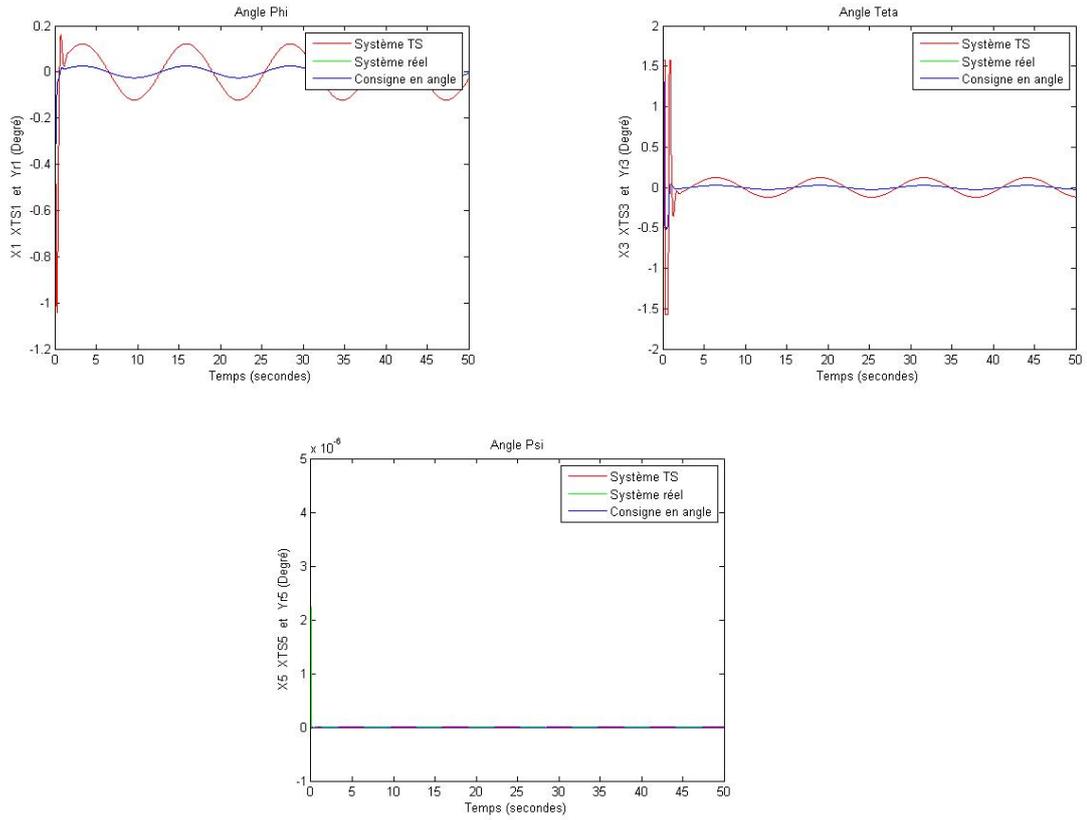


FIG. 4.5 : Les angles

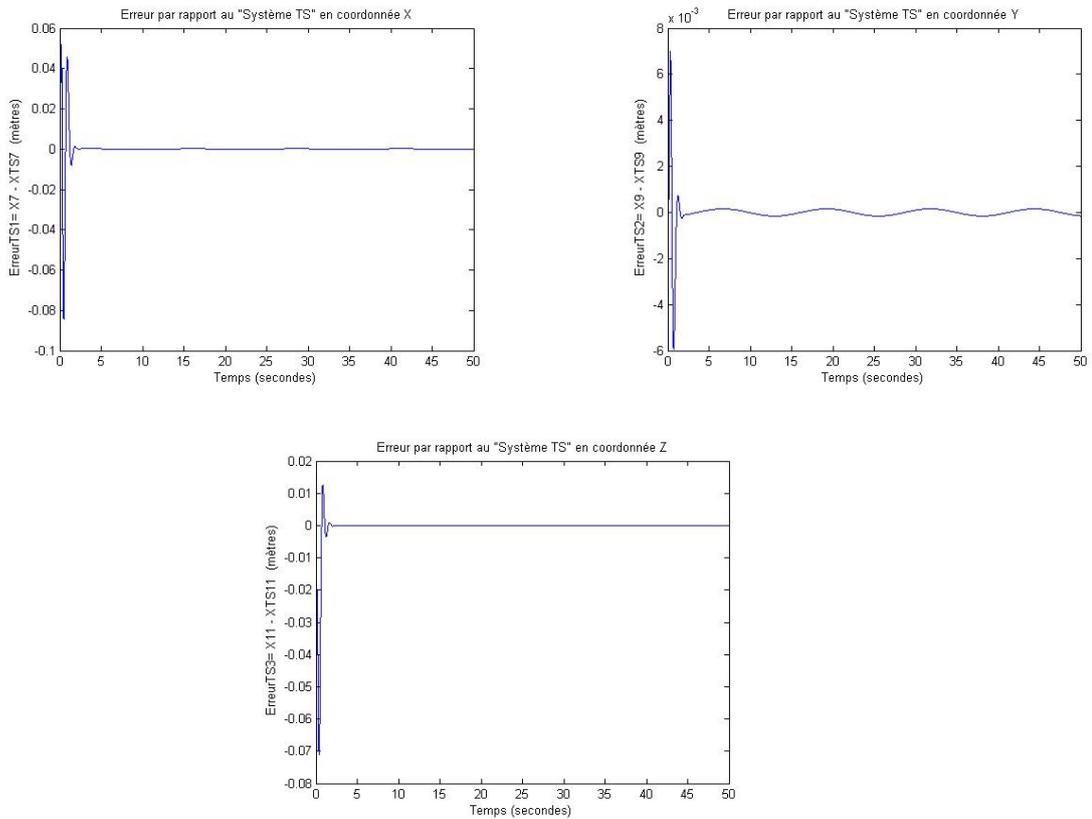


FIG. 4.6 : Erreur

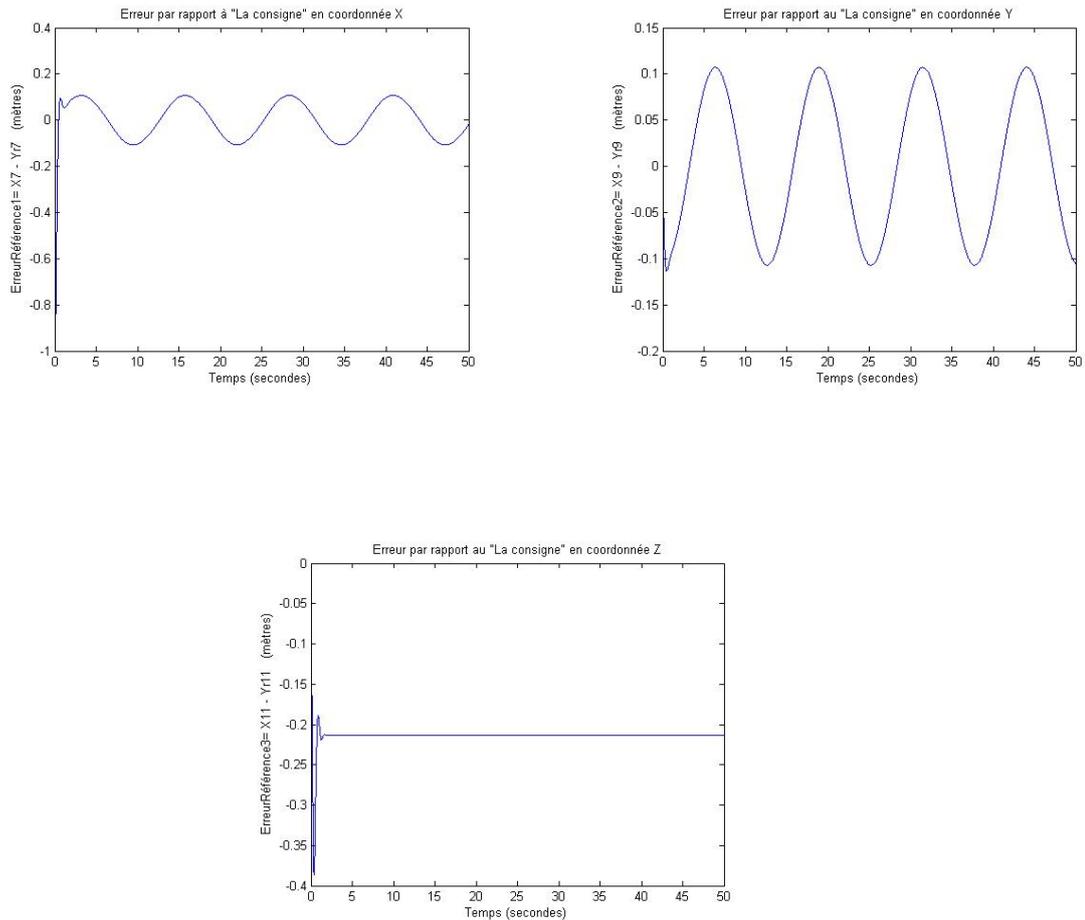


FIG. 4.7 : Les erreurs du Pousuite

L'implémentation de telle commande optimale a produit une poursuite précise et rapide vers les consignes ayant une bande fréquentielles assez large, ce qui montre bien le potentiel de l'aspect d'optimalité.

La commande hybride TS-MPC nous a donné des résultats plus satisfaisants que les approches étudiées précédemment notamment en ce qui concerne la précision de la poursuite, on aperçoit que l'erreur de poursuite suivant l'axe  $X$  et  $Y$  varie dans le voisinage de 0.1 mètre tandis que la poursuite à travers l'axe  $Z$  se stabilise dans 0.2 mètre qui est une amélioration visible par rapport à la commande PDC par modèle de référence dans la figure (3.6) ainsi par rapport au mode glissant dans la figure (2.26).

Les résultats obtenus sont engendré par le fait que le régulateur établi une optimisation dynamique à chaque instant d'échantillonnage en prenant en considération le tau d'évolution énergétique des ressources qui sont les puissances des propulseurs ainsi l'énergie que présente l'erreur par rapport à la consigne désirée, cette optimisation est faite rigoureusement selon un horizon de prédiction  $Np=50$  instants qui assure la connaissance instantanée de l'évolution des états futurs du quadri-rotor par le moyen des informations provenant des capteurs gyroscopique et de localisation. Cette prédiction permettra de se décider sur le signal de commande que doit le régulateur prédictif générer dans le prochain instant ce qui permettra d'avoir un comportement dynamique plus compact par rapport aux approches précédentes qui se basent uniquement sur la connaissance de l'états actuel.

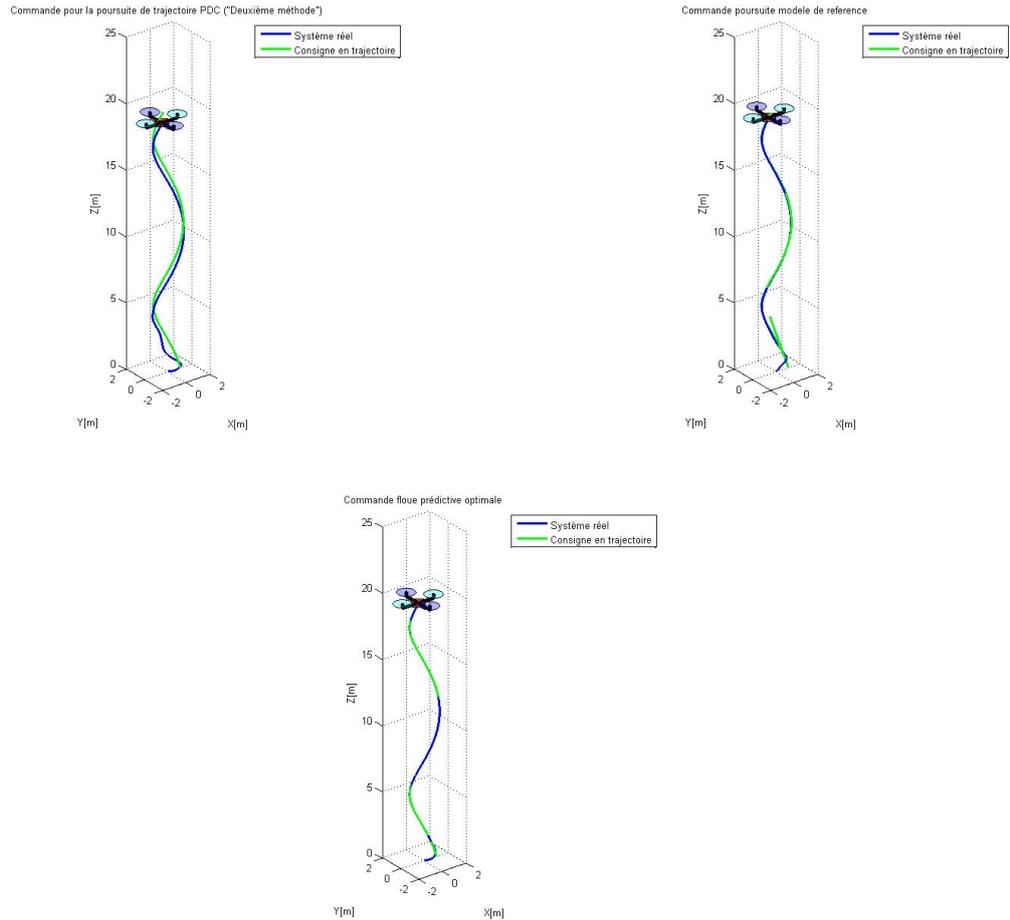


FIG. 4.8 : Les animations en 3d

Les trois animations correspondent à leurs propres approches de commande, l'évolution dans l'espace 3d dépend aux performances fournies par chacune ceci est bien aperçu. Les résultats obtenus reflètent les courbes générées dans chaque méthode on remarque que la commande prédictive introduit une meilleure précision, tandis que l'approche du modèle de référence donne de meilleur résultat que celle de la commande de poursuite de trajectoire PDC . L'animation de tels systèmes permet d'interpréter physiquement la performance et les avantages que porte chaque méthode

## 4.7 Conclusion

Le présent chapitre était une illustration sur l'avantage que présente la modélisation floue TS, on s'est intéressé par l'implémentation d'une commande optimale prédictive sur ce dernier qui porte sur la minimisation de l'énergie pondérée de la commande ainsi par l'erreur quadratique de la poursuite, la résolution de problème quadratique nous a permis de retrouver une loi de commande optimale qui satisfait le cahier de charge que décrit ce critère dans le cadre des limites physiques du quadri-rotor. Les résultats obtenus ont montré une amélioration satisfaisante en performances et ont permis aussi d'exploiter les limites physiques de notre système.

# Chapitre 5

## Implémentation pratique

### 5.1 Introduction

Le présent chapitre est dédié l'implémentation pratique d'une loi de commande PDC stabilisante sur la structure d'un quadri-rotor de type Dji-F450.

On s'intéresse dans un premier lieu à présenter l'équipement utilisé, et puis on passe vers la partie synthèse pour laquelle les gains multimodèles sont calculés sous « Matlab » et puis injectés dans la carte de commande.

### 5.2 Structure générale du quadri-rotor

Le robot utilisé est doté de deux manières possibles pour le vol aérien, la structure en + ou bien la structure en X.

Selon la même manière utilisée dans la partie modélisation dans le chapitre 1, la structure en + est adoptée donc l'inclinaison en angle est assurée uniquement par deux moteurs opposés.



FIG. 5.1 : Aperçu de la plateforme des essais

Le châssis comporte les composantes suivantes :

#### 5.2.1 Moteurs A2212

C'est des moteurs à courant continu à trois fils, sans balais, de puissance 120 W avec un rendement de 75% et qui fonctionnent en tension, l'évolution de leurs vitesses est garantie par la variation de la tension tandis que le couple généré par chacun est en fonction du courant de la ligne.

Les propulseurs peuvent atteindre une vitesse maximale de 11000 tour/minute. Ce type d'actionneurs est largement employé dans ce domaine d'application vu leur robustesse et leur couple important par rapport aux autres types de machine.



FIG. 5.2 : Le moteur A2212

### 5.2.2 Les variateurs de vitesse ESC (Electronic Speed Controller)

Ils représentent les pré-actionneurs qu'ils reçoivent des signaux de commande PWM (Pulse Width Modulation) de la part de la carte de commande, et puis en fonction de ce signal les variateurs font la conversion de la puissance depuis l'alimentation vers la partie opérative qui sont les moteurs avec une puissance maximale d'environ 300W.

La structure du ESC repose sur le principe de fonctionnement d'un hacheur à quatre quadrant lequel qu'il fait tourner les moteurs à courant continu selon les quatre axes en couple et en vitesse avec la possibilité de la récupération de l'énergie dissipée comme illustré dans la figure :

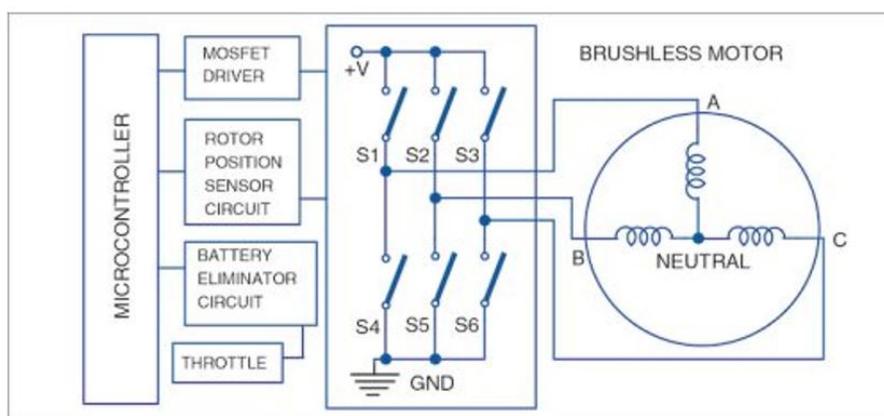


FIG. 5.3 : Stratégie de commande des ESC [5]

Les interrupteurs sont de type « Mosfet » lesquels recevant les excitations PWM au sein de leurs « Gate » et puis par le moyen de la stratégie de commande préprogrammée au niveau du microcontrôleur du variateur les interrupteurs font les commutations nécessaires afin de générer le signal de puissance qui fait exciter les bobines du moteurs pour l'entraîner. Le régulateur du ESC doit connaître impérativement la position du rotor de celui du moteur ceci est mesuré par un capteur incrémentale rotatif, pour notre présent cas les moteurs ne sont pas dotés par ce retour en position, en effet les bobines excitées génèrent un champs magnétique excitant les bobines inactives ce qui permet de générer un champs magnétique inverse dedans, cette inversion en champs produit une tension inverse par les bobines inactives vers les commutateurs du variateur, la valeur de la tension inverse générée est variable à la position du moteur ce qui

permet de créer une sorte de retour, le ESC a tendance donc d'estimer la position du rotor mais selon une manière moins précise de celle faite par le capteur réel.

La figure suivante montre le type du variateur utilisé dans le montage réel :



FIG. 5.4 : Aperçu du ESC

### 5.2.3 Le capteur MPU6050

C'est l'instrument de mesure principal doté d'un gyroscope pour l'inclinaison ainsi d'un capteur accéléromètre et ayant six degré de liberté lui permettant de mesurer précisément les inclinaisons que fait le quadri-rotor en mouvement.

Le calcul de l'inclinaison est estimé par le capteur gyroscopique tandis que l'amélioration de l'information est faite par la combinaison avec le capteur accéléromètre en faisant la double intégration de la valeur de l'accélération, cette information va être fusionnée avec celle provenant de la part de l'inclinaison réelle.

Les données sont codées numériquement et puis transmises à travers un bus de donnée de type I2C vers la carte de commande.

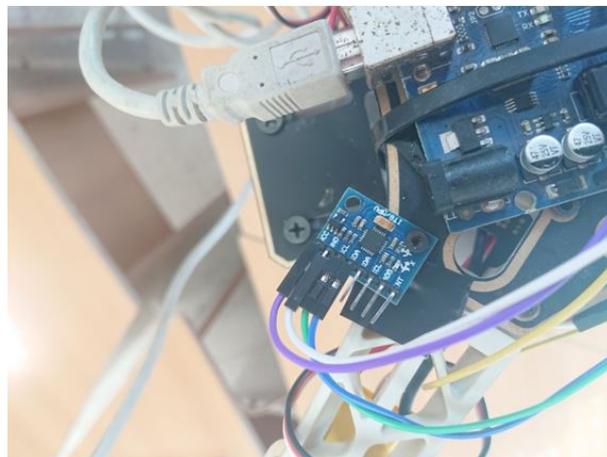


FIG. 5.5 : Aperçu du capteur d'inclinaison

### 5.2.4 La carte de commande

Le calcul en temps réel des signaux de la loi de commande floue PDC est établies par une carte numérique de type « Arduino Uno » à 10 Bits et ayant une fréquence de traitement de 16Mhz, la carte à l'aptitude de traiter des données entrées/sorties numériques et analogiques à 8Bits. Pour notre présente application les données qui proviennent du gyroscope sont transmises par un protocole de communication I2C.

La programmation du régulateur PDC est faite sous le langage spéci Arduino C++, pour générer les signaux PWM vers les ESC en utilisant la bibliothèque « Servo.h ». Les traitements matriciels sont décrits par le moyen de la bibliothèque de l'algèbre linéaire « BasicLinearAlgebra.h ».

Les informations fournies par le capteur des inclinaisons sont manipulées par la bibliothèque « Wire.h ».

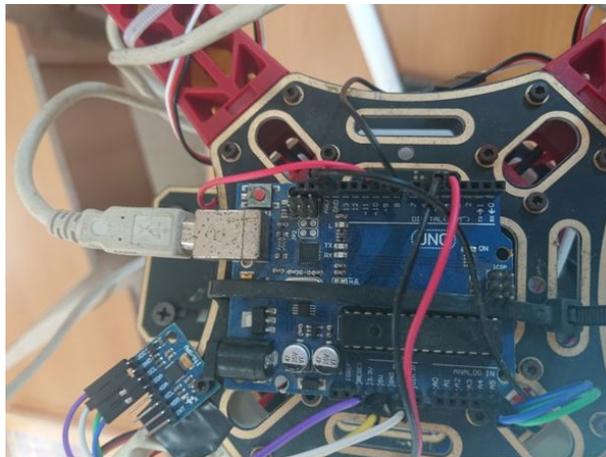


FIG. 5.6 : Aperçu de la carte de commande

## 5.3 Synthèse de commande

Les paramètres du modèle F450 sont différents de ceux du modèle utilisé précédemment en simulation. Pour pouvoir stabiliser le système réel, il convient donc de retrouver la bonne combinaison des régions LMI assurant les performances désirées.

Les paramètres du modèle F450 [14] :

Paramètre	Valeur
$m$	2.5 kg
$g$	9,806 m/s <sup>2</sup>
$l$	0,225 m
$b$	9.810 <sup>-6</sup> N/rad/s
$d$	1.610 <sup>-7</sup> N.m/rad/s
$J_r$	2.810 <sup>-6</sup> kg · m <sup>2</sup>
$I_x$	3,510 <sup>-3</sup> kg · m <sup>2</sup>
$I_y$	3,510 <sup>-3</sup> kg · m <sup>2</sup>
$I_z$	5,110 <sup>-3</sup> kg · m <sup>2</sup>

TAB. 5.1 : Paramètres du modèle du quadrirotor réel

Après avoir tenté plusieurs régions LMI, on s'est intéressé par un comportement moins rapide que celui fait dans la partie simulation. Puisque il s'agit d'une simple stabilisation et afin d'éviter la saturation des actionneurs, on s'est limité, un temps de réponse de 5 secondes. Les paramètres de placement de pôles sont :

$$\alpha = -0.5, \quad \beta = 1, \quad \theta = 1$$

Les gains trouvés sont :

$$K1 = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0.1618 & 0.0707 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.1618 & 0.0707 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.233 & 0.2776 \end{pmatrix}$$

$$K2 = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0.5302 & 0.4410 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.6794 & 0.8488 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.7747 & 0.8196 \end{pmatrix}$$

$$K3 = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0.6566 & 0.8256 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.5525 & 0.4653 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.7553 & 0.8002 \end{pmatrix}$$

$$K4 = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0.0166 & 0.1576 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.0166 & 0.1576 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.0387 & 0.0698 \end{pmatrix}$$

La loi de commande stabilisante de la première approche prend la forme :

$$U = - \sum_{j=1}^4 \mu_j \cdot K_j \cdot X \tag{5.5}$$

A partir de cette loi de commande et d'après des conditions initiales non nulles, les résultats suivants sont obtenues :

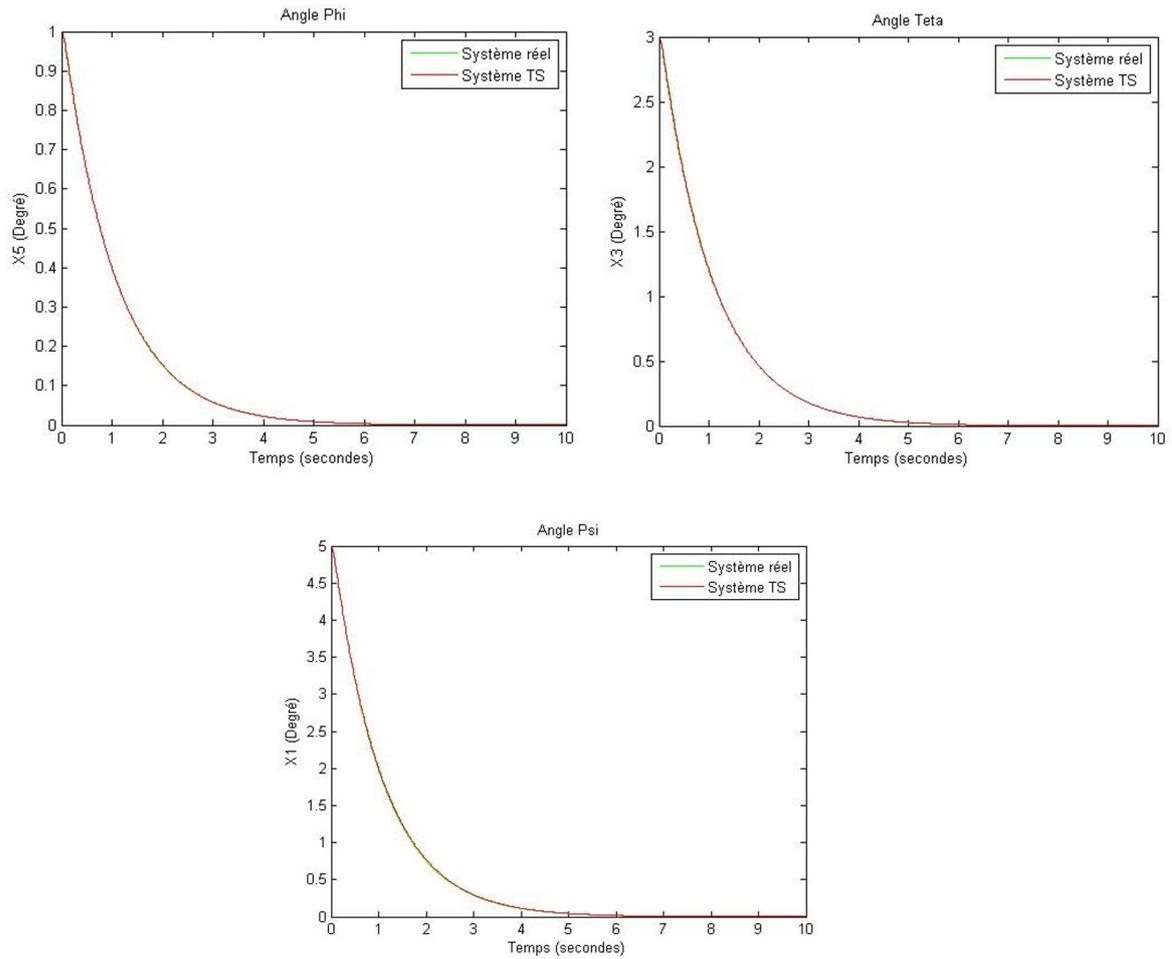


FIG. 5.7 : les angles

Ces résultats sont exploités par la suite dans l'implémentation réelle.

## 5.4 Implémentation

Le montage est basé sur deux fils, un fil pour l'alimentation ainsi que la programmation de la carte de commande et un fil de puissance à partir de la batterie LIPO ce qui empêche la liberté d'effectuer le mouvement de lacet, pour cette raison, on se limite uniquement avec les deux angles roulis et tangage.

Les signaux PWM sont calculé à partir d'une règle de trois, au niveau de la programmation le signal générer peut varier entre 0-255 (signal à 8bits) La valeur maximale PWM représente la valeur de la vitesse maximale, ce qui permet de déduire directement la valeur correspondante en PWM.

Les vitesses sont déduites par l'inversion matricielle :  $w = M^{-1}.U$

### Aperçu du système en fonctionnement :



FIG. 5.8 : Système en fonctionnement

Lorsque on met seulement deux moteurs opposés en fonctionnement (par sous modèle d'angle), l'état arrive à se stabiliser selon la performance correspondante à celle trouvée dans la simulation tandis que lorsque on utilise les quatre moteurs, les performances diminuent un peu à cause du moments réactif générés par les quatre moteurs de un et de deux est du au fait que les moteurs ne sont pas nettement identiques leurs moments d'inerties changent proportionnellement avec leurs vitesses (on a utilisé une valeur moyenne de  $J_r$ ).

Mais ca reste une démonstration indiquant le tau de faisabilité de la théorie du TS et sa simplicité d'implémentation dans ce genre d'application, le système réel malgré la dégradation en performance arrive à rejoindre l'origine des angles en s'appuyant sur une loi de commande PDC synthétisée sur la base des paramètres du procédé réel.

## 5.5 Conclusion

Le chapitre actuel porte sur une implémentation pratique d'une commande PDC stabilisante des angles vers l'origine, nous nous sommes initié avec la familiarisation préliminaire avec le matériel et puis à travers les paramètres du modèle F450 nous avons pu généré les gains multimodèles assurant le caractère de stabilisation via une régulation au niveau de la carte de commande Arduino qui recoit en temps réel les inclinaisons en angles et puis par le moyen des gains trouvés, ca produit une commande PWM qui en découle de la contribution de chaque multimodèle selon l'évolution dynamique du quadri-rotor F450.

# Conclusion générale

Le présent document était une illustration théorique sur les possibilités et les chemins qui peuvent être envisagés durant la commande du système non linéaire complexe et le modèle du quadri-rotor était un exemple adéquat.

La théorie floue au sens de « Takagi-Sugeno » a été employée afin de construire un modèle flou linéaire équivalent au modèle du quadri-rotor non linéaire et puis à travers les algorithmes des LMI, on a pu synthétiser des lois de commandes PDC afin de satisfaire le cahier de charge portant sur la stabilisation et la poursuite sur les grandeurs d'états du modèle selon deux approches.

Une commande par modèle de référence a été traitée aussi afin d'imposer un comportement dynamique judicieux durant la poursuite, des résultats satisfaisants ont été obtenus et cela a montré l'avantage de la représentation TS par rapport à la complexité que stipulent les approches classiques en faisant la comparaison avec le retour d'état et le mode glissant pour chaque méthode.

A la fin une commande optimale prédictive linéaire a été synthétisée et implémentée dans le modèle TS et cela a engendré des résultats beaucoup plus intéressants par rapport à ce qui précède, un seul inconvénient que porte la procédure et qui était sur le coût des calculs que doit le calculateur exécuter.

Dans le dernier chapitre, une tentative d'implémentation d'une commande PDC stabilisante des angles d'un quadri-rotor réel a été faite à laquelle les gains multimodèles ont été calculés et implémentés au sein de la carte de commande, les résultats ont montré le gain de simplicité de l'implémentation de la théorie TS d'une part et d'autre part sur la performance que porte. Les deux sous-modèles des angles tangage et roulis arrivent à rejoindre leurs origines selon la simulation mais lorsque on les met simultanément la qualité des résultats a diminué vu l'imprécision paramétrique que portent les moteurs.

# Bibliographie

- [1] Ademola ABDULKAREEM et al. "Modeling and Nonlinear Control of a Quadcopter for Stabilization and Trajectory Tracking". In : *SSRN Electronic Journal* (jan. 2022). DOI : [10.2139/ssrn.4036208](https://doi.org/10.2139/ssrn.4036208).
- [2] Ademola ABDULKAREEM et al. "Modeling and Nonlinear Control of a Quadcopter for Stabilization and Trajectory Tracking". In : *SSRN Electronic Journal* (jan. 2022). DOI : [10.2139/ssrn.4036208](https://doi.org/10.2139/ssrn.4036208).
- [3] H. S. BAE et al. "Digital state feedback control and feed-forward compensation for a parallel module DC-DC converter using the pole placement technique". In : *2008 Twenty-Third Annual IEEE Applied Power Electronics Conference and Exposition*. 2008, p. 1722-1727. DOI : [10.1109/APEC.2008.4522959](https://doi.org/10.1109/APEC.2008.4522959).
- [4] Souad BEZZAOUCHA REBAÏ et al. "Model Reference Tracking Control for Nonlinear Systems described by Takagi-Sugeno Structure". In : juill. 2013. DOI : [10.1109/FUZZ-IEEE.2013.6622398](https://doi.org/10.1109/FUZZ-IEEE.2013.6622398).
- [5] CIRCUIT. *How Brushless DC Motor Works ? BLDC and ESC Explained*. en. Mars 2019. URL : <https://howtomechatronics.com/how-it-works/how-brushless-motor-and-esc-work/> (visité le 18/02/2023).
- [6] Laloui DERAFA, Tarek MADANI et Abdelaziz BENALLEGUE. "Dynamic Modelling and Experimental Identification of Four Rotors Helicopter Parameters". In : jan. 2007, p. 1834 -1839. DOI : [10.1109/ICIT.2006.372515](https://doi.org/10.1109/ICIT.2006.372515).
- [7] mohamed HOSSNY, Ayman EL-BADAWY et Ragi HASSAN. "Fuzzy Model Predictive Control of a Quadrotor Unmanned Aerial Vehicle". In : sept. 2020, p. 1704-1713. DOI : [10.1109/ICUAS48674.2020.9213902](https://doi.org/10.1109/ICUAS48674.2020.9213902).
- [8] L HSU. "Nonlinear Control Systems : Techniques and Applications. Wiley." In : 2009. DOI : [10.1109/ICUAS48674.2020.9213902](https://doi.org/10.1109/ICUAS48674.2020.9213902).
- [9] Bin LI et Yaxin WANG. "An Enhanced Model Predictive Controller for Quadrotor Attitude Quick Adjustment with Input Constraints and Disturbances". In : *International Journal of Control, Automation and Systems* 20 (fév. 2022), p. 648-659. DOI : [10.1007/s12555-020-0815-9](https://doi.org/10.1007/s12555-020-0815-9).
- [10] John H. LILLY. "Parallel Distributed Control with Takagi-Sugeno Fuzzy Systems". In : *Fuzzy Control and Identification*. 2010, p. 106-120. DOI : [10.1002/9780470874240.ch7](https://doi.org/10.1002/9780470874240.ch7).
- [11] Johnson LOPEZ et al. "A Robust  $H_{\infty}$  Controller for an UAV Flight Control System". In : *The Scientific World Journal* 2015 (juin 2015), p. 1-11. DOI : [10.1155/2015/403236](https://doi.org/10.1155/2015/403236).

- [12] Shreya MAGGO et al. “Comparative Study of PID and FOPID Control Techniques for a Quadcopter”. In : *2022 3rd International Conference on Electrical Engineering and Informatics (ICon EEI)*. 2022, p. 59-63. DOI : [10.1109/IConEEI55709.2022.9972331](https://doi.org/10.1109/IConEEI55709.2022.9972331).
- [13] Luis MARTINS, C. CARDEIRA et Paulo OLIVEIRA. “Feedback Linearization with Zero Dynamics Stabilization for Quadrotor Control”. In : *Journal of Intelligent Robotic Systems* 101 (jan. 2021). DOI : [10.1007/s10846-020-01265-2](https://doi.org/10.1007/s10846-020-01265-2).
- [14] Ngoc Phi NGUYEN et Sung HONG. “Sliding Mode Thau Observer for Actuator Fault Diagnosis of Quadcopter UAVs”. In : *Applied Sciences* 8 (oct. 2018), p. 1893. DOI : [10.3390/app8101893](https://doi.org/10.3390/app8101893).
- [15] H. OHTAKE, K. TANAKA et H.O. WANG. “Fuzzy modeling via sector nonlinearity concept”. In : *Proceedings Joint 9th IFSA World Congress and 20th NAFIPS International Conference (Cat. No. 01TH8569)*. T. 1. 2001, p. 127-132. DOI : [10.1109/NAFIPS.2001.944239](https://doi.org/10.1109/NAFIPS.2001.944239).
- [16] Djamel OUNNAS et al. “A Fuzzy Tracking Control Design Strategy for Wind Energy Conversion System”. In : nov. 2015. DOI : [10.13140/RG.2.1.1262.9845](https://doi.org/10.13140/RG.2.1.1262.9845).
- [17] Danica ROSINOVÁ et Mária HYPÍUSOVÁ. “LMI Pole Regions for a Robust Discrete-Time Pole Placement Controller Design”. In : *Algorithms* 12.8 (2019). ISSN : 1999-4893.
- [18] Ahmed SAMIR et al. “Quadcopter Trajectory Tracking Control using State-Feedback Control with Integral Action”. In : *International Journal of Computer Applications* 168 (juin 2017), p. 1-7. DOI : [10.5120/ijca2017914539](https://doi.org/10.5120/ijca2017914539).
- [19] Gembong Edhi SETYAWAN, Wijaya KURNIAWAN et Amroy Casro Lumban GAOL. “Linear Quadratic Regulator Controller (LQR) for AR. Drone’s Safe Landing”. In : *2019 International Conference on Sustainable Information Engineering and Technology (SIET)*. 2019, p. 228-233. DOI : [10.1109/SIET48054.2019.8986078](https://doi.org/10.1109/SIET48054.2019.8986078).
- [20] Abdelhamid TAYEBI et S. MCGILVRAY. “Attitude stabilization of a four-rotor aerial robot”. In : t. 2. Jan. 2005, 1216 -1221 Vol.2. ISBN : 0-7803-8682-5. DOI : [10.1109/CDC.2004.1430207](https://doi.org/10.1109/CDC.2004.1430207).
- [21] Liuping WANG. *Model Predictive Control System Design and Implementation using MATLAB*. Mars 2009. ISBN : 978-1-84882-330-3.

# Webographie

- [5] CIRCUIT. *How Brushless DC Motor Works ? BLDC and ESC Explained*. en. Mars 2019. URL : <https://howtomechatronics.com/how-it-works/how-brushless-motor-and-esc-work/> (visité le 18/02/2023).

# Annexes

# Annexe A

## Commande par « Sliding Mode »

### A.1 Introduction

La commande non linéaire a reconnu un large emploi dans le plan pratique et également dans les travaux de recherches, certains procédés dynamiques notamment ceux de l'industrie présentent des non linéarités et des complexités qui empêchent l'emploi des techniques classiques du linéaire un cadre ce qui oblige de se baser sur la forme la plus générale possible de la commande des systèmes asservis.

Les travaux de « Lyapounov » ont donné un formalisme compacte et robuste de synthèse des lois de commandes pour une quelconque forme de modélisation des processus, elle est basée sur la proposition d'une fonction candidate représentant une trajectoire d'état qu'on désire l'amener vers l'origine, ceci permet de déterminer la loi de commande satisfaisant l'objectif de control désiré quoi que ce soit pour une stabilisation ou bien pour une poursuite.

En contrepartie, cette procédure nécessite le choix approprié de la fonction candidate, certains systèmes peuvent avoir des propositions évidentes et intuitives vu leurs caractères dynamique mais ca reste toujours limité pour d'autres cas de figure, les cas les plus généraux.

Plusieurs travaux ont vu le jour afin d'élaborer la généralisation la plus compacte de la commande non linéaire, parmi ces travaux des approches permettant de formuler la fonction candidate de Lyapounov en se basant sur des méthodes systématiques, le « mode glissant » ou bien le fameux « Sliding Mode » est l'un de ces approches ou il sera détaillé et implémenté dans ce chapitre.

### A.2 Aspect du glissement

La procédure repose sur l'attraction de l'ensemble des trajectoires d'états vers une fonction d'états appelée « Surface de glissement » dont laquelle des objectifs de commande se réalisent dedans.

La surface de glissement fait appel généralement aux états d'erreurs et ces dérivées successives afin de ramener les variables d'états vers des états désirés.

$$S = f(e, \dot{e}, \ddot{e}, \dots, e^n), \quad e = X - X_d \tag{A.1}$$

Avec  $S=0$ , si  $X_1=X_2=\dots=X_n=0$

La méthode a reconnu un large d'emploi dans la gamme des procédés industriels auxquels ils stipulent sur des aspects dynamiques dont les régulateurs linéaires classiques ne satisfont plus les performances adéquates.

Le mode glissement se devise en deux principaux termes Un terme comportant les propriétés intrinsèques du processus appelé le terme d'équilibre, et un autre terme possédant le caractère discontinu commutatif du mode glissant comme c'est mentionné dans le formule ci-dessous :

$$U = U_{\text{equilibre}} + U_{\text{glissant}} \quad (\text{A.2})$$

Avec

$$U_{\text{glissant}} = \begin{cases} U_{\text{max}}^+ & \text{si } S > 0 \\ U_{\text{max}}^- & \text{si } S < 0 \end{cases} \quad (\text{A.3})$$

En suite le terme d'équilibre sera calculé moyennant la première méthode de Lyapounov, pour ceci on introduit une brève explication sur le théorème.

- Premier théorème de « Lyapounov » :

Etant donné un système réel ayant la forme

$$\dot{X} = f(X, U) \quad (\text{A.4})$$

On propose la fonction candidate représentant le comportement dynamique en boucle fermée pendant un objectif de commande donné :

$$V = f(X, e, \dot{e}, \ddot{e}, \dots) \quad (\text{A.5})$$

Si  $V$  est une fonction définie positive dans  $\mathbb{R}^n$  et qui s'annule à l'origine des états, le système en boucle fermé est asymptotiquement stable au sens de « Lyapounov » si seulement si

$$\dot{V} < 0 \quad (\text{A.6})$$

La satisfaction de cette condition assure la convergence en régime permanent de la fonction candidate  $V$  vers l'origine.

Le développement de la loi de commande s'établi moyennant la deuxième condition dont laquelle l'expression peut être déduite analytiquement.

Ce théorème sera l'outil de la détermination du régime d'équilibre du mode glissement.

Il est important de mentionner que l'approche est dédiée vers des cas particuliers des systèmes, ou leurs représentation d'état possèdent une forme de cascade entre les états comme illustré ci-dessous :

$$\begin{aligned}\dot{X}_1 &= f(X_1, X_2, \dots, X_n) \\ \dot{X}_2 &= f(X_2, X_3, \dots, X_n) \\ &\vdots \\ \dot{X}_n &= f(X_n, U)\end{aligned}\tag{A.7}$$

### A.2.0.1 Notion du degré relatif

Pour les formes des systèmes illustrés précédemment, le degré relatif représente le nombre des dérivées successives qui peuvent s'effectuer au niveau de la sortie

$$y = h(X)\tag{A.8}$$

Afin d'y avoir l'occurrence du terme de commande

$$y^r = f(X_1, X_2, \dots, X_n, U)\tag{A.9}$$

Ce degré demeure indispensable pour formuler la surface de glissement afin de synthétiser la loi permettant d'envisager l'attractivité vers cette fonction.

La formulation de la surface de glissement a eu plusieurs manières dans le domaine de recherche, dans la présente synthèse on s'intéresse uniquement à celle qui est intitulée surface de « Slotine ».

L'expression prend la forme suivante :

$$S = \left(\gamma + \frac{d}{dt}\right)^{r-1} \cdot e \quad \text{avec} \quad e = y - Yr\tag{A.10}$$

$\gamma$  : Étant un paramètre positif servant à réguler la dynamique de la surface.

$Yr$  : le signal de référence

$r$  : le degré relatif

La fonction candidate de « Lyaponouv » est :

$$V = \frac{1}{2} \cdot S^2\tag{A.11}$$

Qui est une fonction définie positive dans  $\mathbb{R}^n$  et elle s'annule dans l'origine car  $S = 0$  à l'origine

On dérive

$$\dot{V} = S \cdot \dot{S}\tag{A.12}$$

Pour que la fonction s'attire vers l'origine il faut réaliser la stabilité asymptotique qui se satisfait par la condition

$$\dot{V} = S \cdot \dot{S} < 0 \quad (\text{A.13})$$

Pour se fait la proposition est :

$$\dot{S} = -K \cdot \text{sign}(S) \quad (\text{A.14})$$

Qui permet de déduire que :

$$(S) \quad \begin{cases} K \cdot t + \alpha, & \text{si } S > 0 \\ -K \cdot t + \alpha, & \text{si } S < 0 \end{cases} \quad \text{avec } \alpha = S(0) \quad (\text{A.15})$$

La courbe de convergence de la surface étant une droite linéaire qui s'oriente vers l'origine selon le signe de  $S$  dans un temps fini exprimé par

$$t_f = \frac{\alpha}{K} \quad (\text{A.16})$$

et donc

$$\dot{V} = S \cdot \dot{S} = -K \cdot \text{sign}(S) \cdot S \quad (\text{A.17})$$

La fonction étant toujours négative par conséquent la stabilité asymptotique au sens du premier théorème de « Lyapounov » est assurée si la condition (A.14) est satisfaite.

Alors l'extraction de la loi de commande en régime glissant se fait par le moyen de l'équation (A.14).

La commande obtenue prend la forme qui suit :

$$U = U_{\text{eq}} - K \cdot \text{sign}(S) \quad (\text{A.18})$$

Le signal de commande sert à maintenir l'attractivité de la trajectoire d'états vers la surface qui a de sa part la tendance de rejoindre l'origine dans un temps fini, cette convergence est du à l'action de commutation que présente la fonction discontinu « sign » d'un part et d'autre part au fait de la nature propre de cette fonction ayant l'avantage d'absorber toute les incertitudes aux niveaux des paramètres intrinsèques du système elle introduit une robustification fiable permettant d'assurer le suivi du signal de consigne vis-à-vis les variations dynamiques des paramètres.

Par contre, si on implémente directement cette approche les avantages de la procédure auront un prix couteux. En effet, les commutations progressives du signal de commandes entre

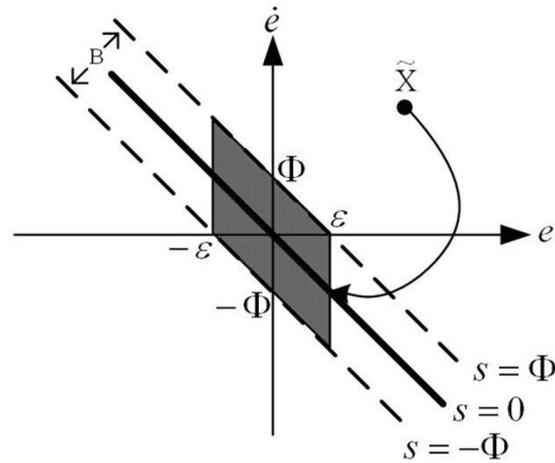


FIG. A.1 : Attractivité de la surface de glissement vers l'origine

les bornes maximales de la puissance de l'actionneur peuvent engendrer des dégâts matériels car ce comportement brusque pouvant être à hautes fréquences n'est pas toujours réalisable à cause de la nature technologique de l'équipement certains instrument stipulent sur les phénomènes réalistes qu'on on peut les éviter tels que l'hystérésis, le temps de réponse, la puissance consommée...

L'application directe de ce type de signaux de control peut causer :

- Un grand bruit au niveau de l'actionneur phénomène de broutement « Chattering »
- Dégradation de la durée de vie de l'équipement
- Génération de la chaleur
- Perte de puissance
- Diminution de la précision vu l'accumulation des bruits
- Excitation des dynamiques non modélisées

Pour remédier ce problème pleines de manières systématiques on été introduites pour le présent chapitre on envisage deux alternatives :

### A.2.1 Mode de glissement d'ordre supérieur (High Level Sliding Mode)

Ca consiste à mettre en évidence les dérivées supérieures de la surface en posant

$$\begin{aligned}
 S &= 0 \\
 \dot{S} &= 0 \\
 \ddot{S} &= 0 \\
 &\vdots \\
 S^n &= -K \cdot \text{sign}(S)
 \end{aligned}
 \tag{A.19}$$

La mise en œuvre de plusieurs conditions en régime permanent permet de restreindre la trajectoire des états du système avec la surface de « Slotine » ce qui limite le voisinage d'évolution des états et donc la déduction d'une loi de commande faisant appel aux expressions qui ne comportent plus la fonction « sign » d'où la dispersion du phénomène de broutement tout en gardant la robustesse de la méthode face aux erreurs de modélisations.

Dans la plus part des cas le mode de glissement d'ordre supérieur se limite avec uniquement la seconde dérivée.

### A.2.2 Fonction équivalente de « Sign »

Ca revient à remplacer la fonction « Sign » par une fonction proche continue de classe au moins  $C^1$  dans  $IR$  et qui se comporte d'une manière proche.

Exemples de fonctions :

- $\tanh(x)$
- $\frac{2}{\pi} \cdot \arctan(x)$
- $\text{Sat}(x)$

Dans ce qui suit, la fonction  $\tanh(x)$  sera utilisée afin d'y avoir de bonnes performances.

## A.3 Implémentation dans le modèle du quadri-rotor

### A.3.1 Sous système des angles

Le modèle est envisagé selon trois sous systèmes d'angles :

$$\begin{aligned}
 \dot{X}_1 &= X_2 \\
 \dot{X}_2 &= a_1 X_4 X_6 + a_2 X_2^2 + a_3 X_4 + b_1 U \quad (I) \\
 \dot{X}_3 &= X_4 \\
 \dot{X}_4 &= a_4 X_2 X_6 + a_5 X_4^2 + a_6 X_2 + b_2 U_3 \quad (II) \\
 \dot{X}_5 &= X_6 \\
 \dot{X}_6 &= a_7 X_2 X_4 + a_8 X_6^2 + b_3 U_4 \quad (III)
 \end{aligned} \tag{A.20}$$

On calcule pour chaque sous-modèle le degré relatif :

On a

$$y = \begin{pmatrix} X_1 \\ X_3 \\ X_5 \end{pmatrix} \tag{A.21}$$

Le degré relatif pour chacun vaut  $r = 2$  et donc les surfaces de « Slotine » sont :

$$S_1 = \left( \gamma_1 + \frac{d}{dt} \right) e_1 = \gamma_1 e_1 + \dot{e}_1, \quad e_1 = X_1 - Y_{r1} \tag{A.22}$$

$$S_2 = \left(\gamma_2 + \frac{d}{dt}\right)e_2 = \gamma_2 e_2 + \dot{e}_2, \quad e_2 = X_2 - Y_{r2} \quad (\text{A.23})$$

$$S_3 = \left(\gamma_3 + \frac{d}{dt}\right)e_3 = \gamma_3 e_3 + \dot{e}_3, \quad e_3 = X_3 - Y_{r3} \quad (\text{A.24})$$

$$\gamma_1, \gamma_2, \gamma_3 > 0 \quad (\text{A.25})$$

En se basant sur la condition de stabilité (I), la synthèse induit les lois de commandes par modes de glissement suivantes :

$$U_2 = \frac{1}{b_1} \left( -K_1 \tanh(S_1) - \gamma_1 \dot{e}_1 - a_1 X_4 X_6 - a_2 X_2^2 - a_3 X_4 + \ddot{Y}_{r1} \right) \quad (\text{A.26})$$

$$U_3 = \frac{1}{b_2} \left( -K_2 \tanh(S_2) - \gamma_2 \dot{e}_2 - a_4 X_2 X_6 - a_5 X_4^2 - a_6 X_2 + \ddot{Y}_{r2} \right) \quad (\text{A.27})$$

$$U_4 = \frac{1}{b_3} \left( -K_3 \tanh(S_3) - \gamma_3 \dot{e}_3 - a_7 X_2 X_4 - a_8 X_6^2 + \ddot{Y}_{r3} \right) \quad (\text{A.28})$$

$$K_1, K_2, K_3 > 0 \quad (\text{A.29})$$

Il est à noter que les paramètres  $K_i$  et  $\gamma_i$  font la régulation de la dynamique du système en boucle fermée : le paramètre  $\gamma_i$  influe sur la rapidité de la réponse par contre le  $K_i$  impacte sur la précision de la poursuite.

Le choix des paramètres devrait être judicieux car des valeurs males choisies peuvent engendrer des erreurs de simulations notamment les erreurs de singularité de la solution puisque les méthodes de résolution numérique « ODE » chacune se caractérise par une précision donnée et qui demeurent sensibles aux systèmes dynamiques fortement non linéaires et couplés comme le modèle du quadri-rotor.

Pour le reste des simulations, on a choisi de continuer avec une méthode numérique scripte intitulée « Runge Koutta 4 » vu la simplicité de l'implémentation des lois de commandes et également ses performances du calcul précis dans un temps très réduit par rapport aux « Simulink » dont on a vécu beaucoup de problèmes de singularités en simulations.

### A.3.2 Stabilisation des angles

Le vol du quadri-rotor doit satisfaire dans un premier lieu la stabilisation des angles par rapport aux conditions de l'environnement réel.

On s'intéresse donc à stabiliser la surface de « Slotine » vers l'origine.

On choisi les paramètres :  $\gamma_1 = \gamma_2 = \gamma_3 = 15$  et  $K1 = K2 = K3 = 150$

Les conditions initiales :  $X = \begin{pmatrix} 5 \\ 0 \\ -3 \\ 0 \\ 3 \\ 0 \end{pmatrix}$

Les commandes initiales :  $U1 = m \cdot g + 1$ ,  $U2 = U3 = U4 = 1$

Ce qui en résulte :

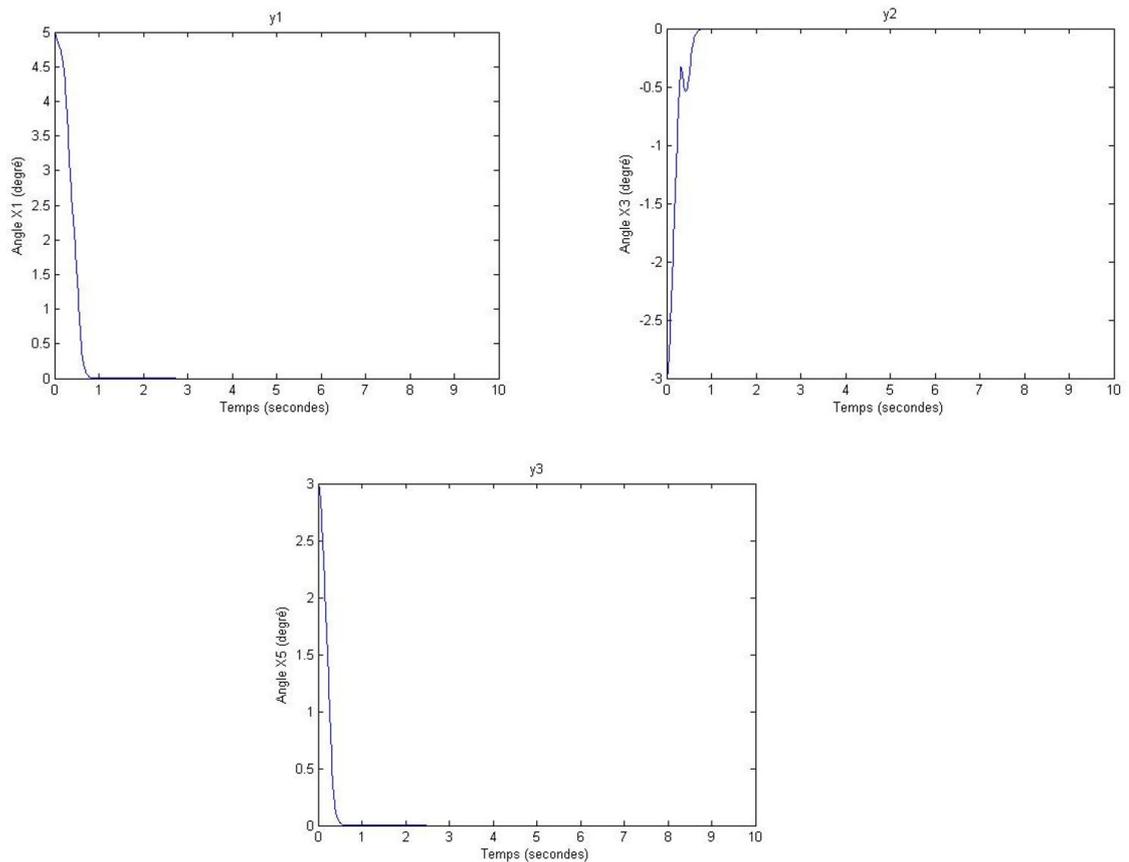


FIG. A.2 : Les sortie y1, y2 et y3

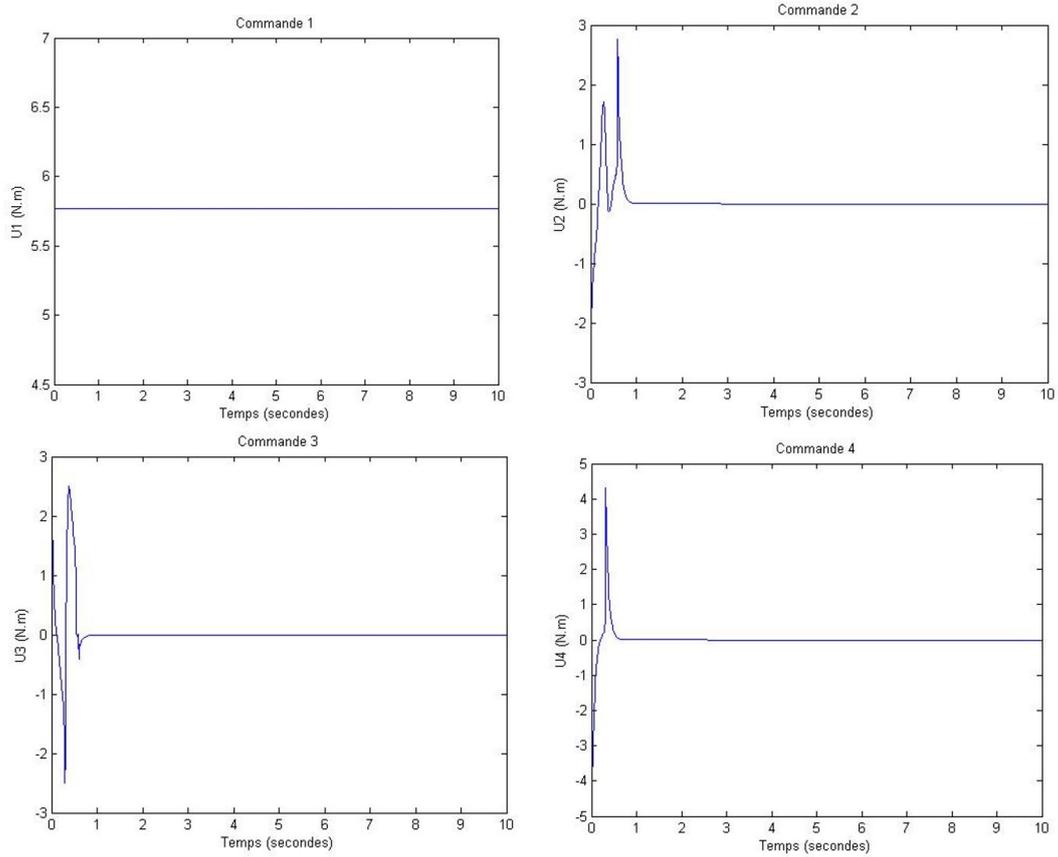


FIG. A.3 : Les signaux de commandes

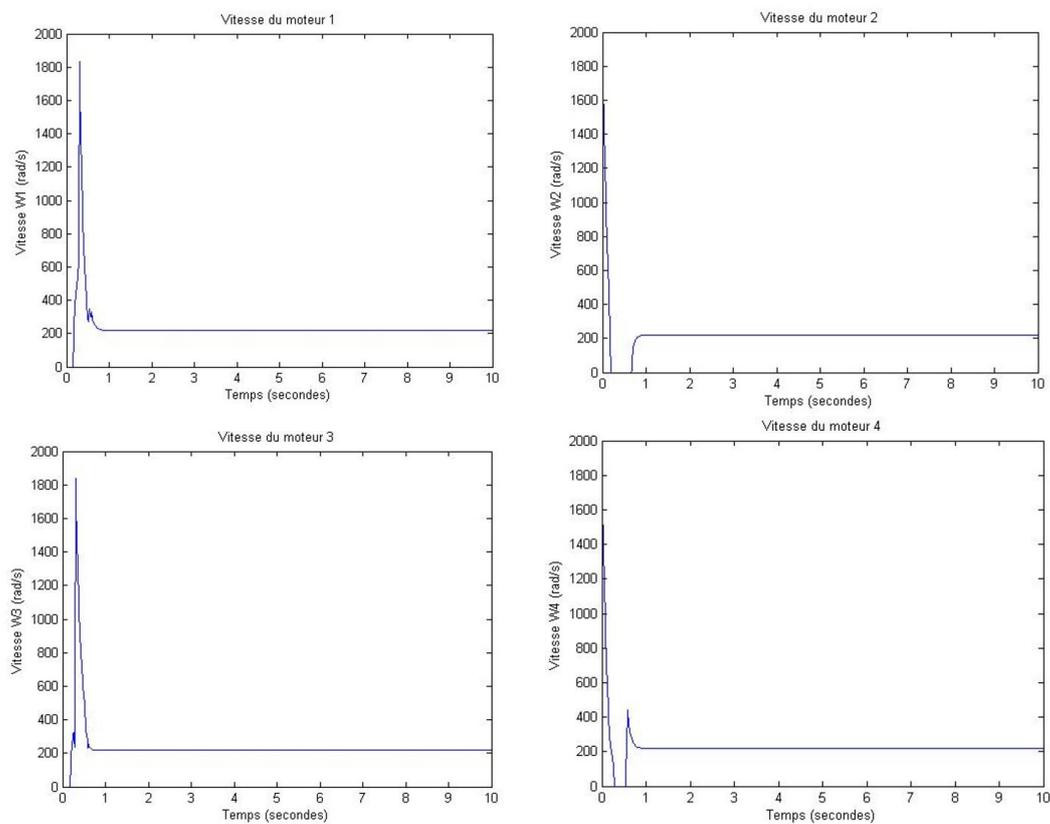


FIG. A.4 : Les vitesses des moteurs

On aperçoit que les sorties se stabilisent vers l'origine au bout de moins d'une seconde tout en gardant une évolution des signaux de commandes lisse n'ayant pas un broutement et réaliste pour les actionneurs ce qui est bien indiqué par l'évolution des vitesses des propulseurs dont on observe une variation critique du à la régulation des angles et puis il se stabilisent en 200 rad/s afin de retourner vers le régime du vol ordinaire.

### A.3.3 Poursuite des trajectoires

La poursuite des consignes variables étant une généralisation de la forme employée lors de la stabilisation, on s'intéresse maintenant à mettre le modèle dans des conditions plus réalistes auxquelles le quadri-rotor devrait maintenir une position désirée selon une performance donnée.

Pour cela, on reprend le sous-système de position afin de synthétiser selon la meme procédure un mode de glissement de poursuite.

Le modèle comme ca été mentionné précédemment repose sur la forme suivante

$$\begin{aligned}
 \dot{X}_7 &= X_8 \\
 \dot{X}_8 &= a_9 X_8 + \frac{1}{m} U_x U_1 \quad \text{(I)} \\
 \dot{X}_9 &= X_{10} \\
 \dot{X}_{10} &= a_{10} X_{10} + \frac{1}{m} U_y U_2 \quad \text{(II)} \\
 \dot{X}_{11} &= X_{12} = 0 \\
 \dot{X}_{12} &= a_{11} X_{12} + \frac{\cos(X_1) \cos(X_3)}{m} U_1 - g = 0 \quad \text{(III)}
 \end{aligned} \tag{A.30}$$

Pour chaque sous-modèle on trouve le degré relatif vaut  $r = 2$

Ce qui permet de définir les surfaces de « Slotine » ci-dessous :

$$S_x = \left( \gamma_x + \frac{d}{dt} \right) e_x = \gamma_x e_x + \dot{e}_x, \quad e_x = X_7 - Y_{r_x} \tag{A.31}$$

$$S_y = \left( \gamma_y + \frac{d}{dt} \right) e_y = \gamma_y e_y + \dot{e}_y, \quad e_y = X_9 - Y_{r_y} \tag{A.32}$$

$$S_z = \left( \gamma_z + \frac{d}{dt} \right) e_z = \gamma_z e_z + \dot{e}_z, \quad e_z = X_{11} - Y_{r_z} \tag{A.33}$$

$$\gamma_x, \gamma_y, \gamma_z > 0 \tag{A.34}$$

En se basant sur la meme condition de stabilité (I) on déduit les lois de commandes de position par mode glissant :

$$U_1 = U_z = \frac{m}{\cos(X_1) \cos(X_3)} \left( -K_z \tanh(S_z) + g - \gamma_z \dot{e}_z - a_{11} X_{12} + \ddot{Y}_{r_z} \right) \tag{A.35}$$

$$U_x = \frac{m}{U_z} \left( -K_x \tanh(S_x) - \gamma_x \dot{e}_x - a_9 X_8 + \ddot{Y}_{r_x} \right) \tag{A.36}$$

$$U_y = \frac{m}{U_z} \left( -K_y \tanh(S_y) - \gamma_y \dot{e}_y - a_{10} X_{10} + \ddot{Y}_{r_y} \right) \tag{A.37}$$

## A.4 Simulation des références variables

Comme ca été cité dans le chapitre de la modélisation, le vol en trajectoire dans l'espace 3D s'établi par le moyen de deux boucles en cascade à chaque instant d'échantillonnage la boucle maitre de position génère des lois de commandes qui se transforment selon les contraintes non-holonomes en sollicitations d'angles la boucle esclave fait donc l'objet de la régulation des angles vers les inclinaisons adéquates pour que le corps rigide puisse se déplacer vers la position voulue.

Afin de simuler précisément cet enchainement de bouclage, nous avons créé deux boucles imbriquées d'asservissement une boucle globale de position ayant une dynamique assez rapide et une autre boucle de régulation en angle ayant un horizon de control ne dépassant pas une seconde.

Dans le plan pratique le régulateur global sollicite des consignes en angles et puis une boucle qui prend une seconde s'établi entre les instants d'échantillonnage de l'asservissement en position.

On choisi les paramètres qui suivent :

$$\gamma_x = \gamma_y = 10, \quad \gamma_z = 4$$

$$K_x = K_y = K_z = 10$$

$$\gamma_1 = \gamma_2 = \gamma_3 = 15$$

$$K_1 = K_2 = K_3 = 100$$

Les commandes initiales :  $U_1 = m \cdot g + 1$ ,  $U_2 = U_3 = U_4 = 1$

Les consignes voulues :

$$Y_{rx} = \cos(0.33t), \quad Y_{ry} = \sin(0.33t), \quad Y_{rz} = 5 \tag{A.38}$$

Ca en résulte :

### A.4.1 La dynamique des angles

L'état est en vert et la consigne est en rouge :

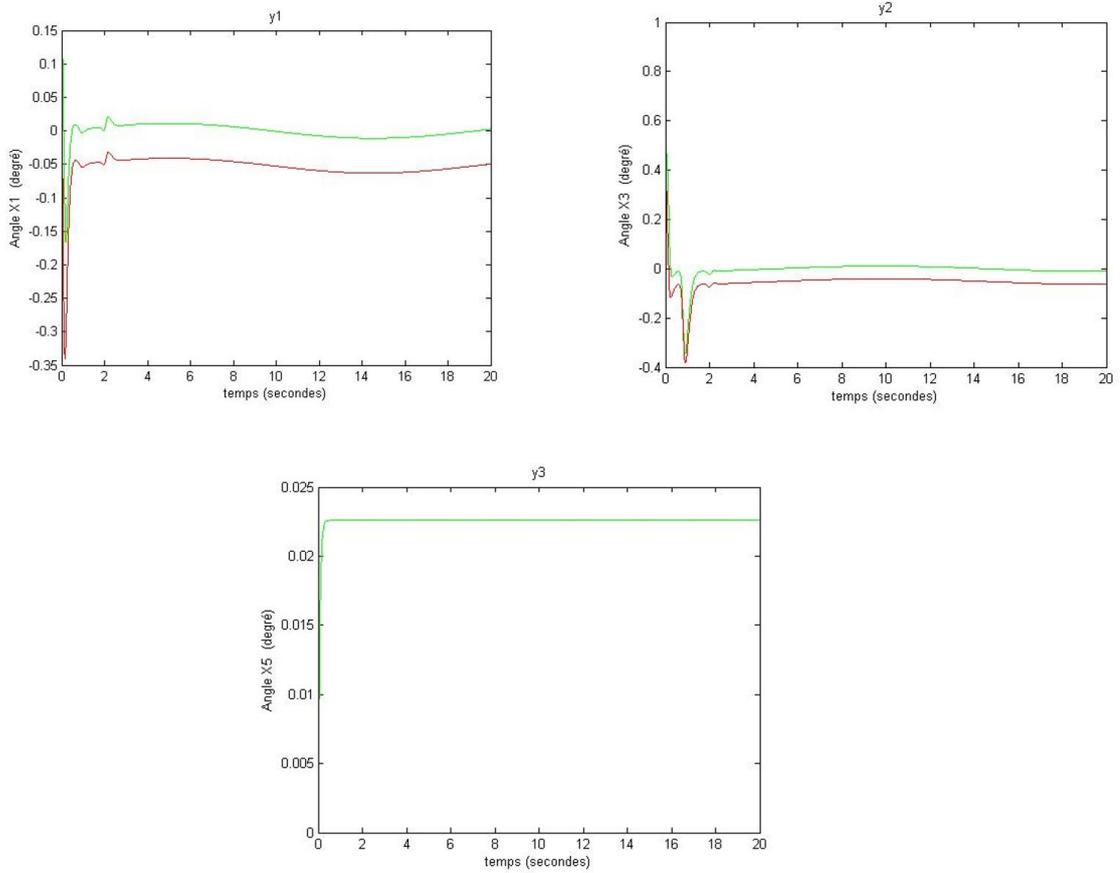


FIG. A.5 : Les sortie y1, y2 et y3

### A.4.2 La dynamique de position

L'état est en vert et la consigne est en rouge :

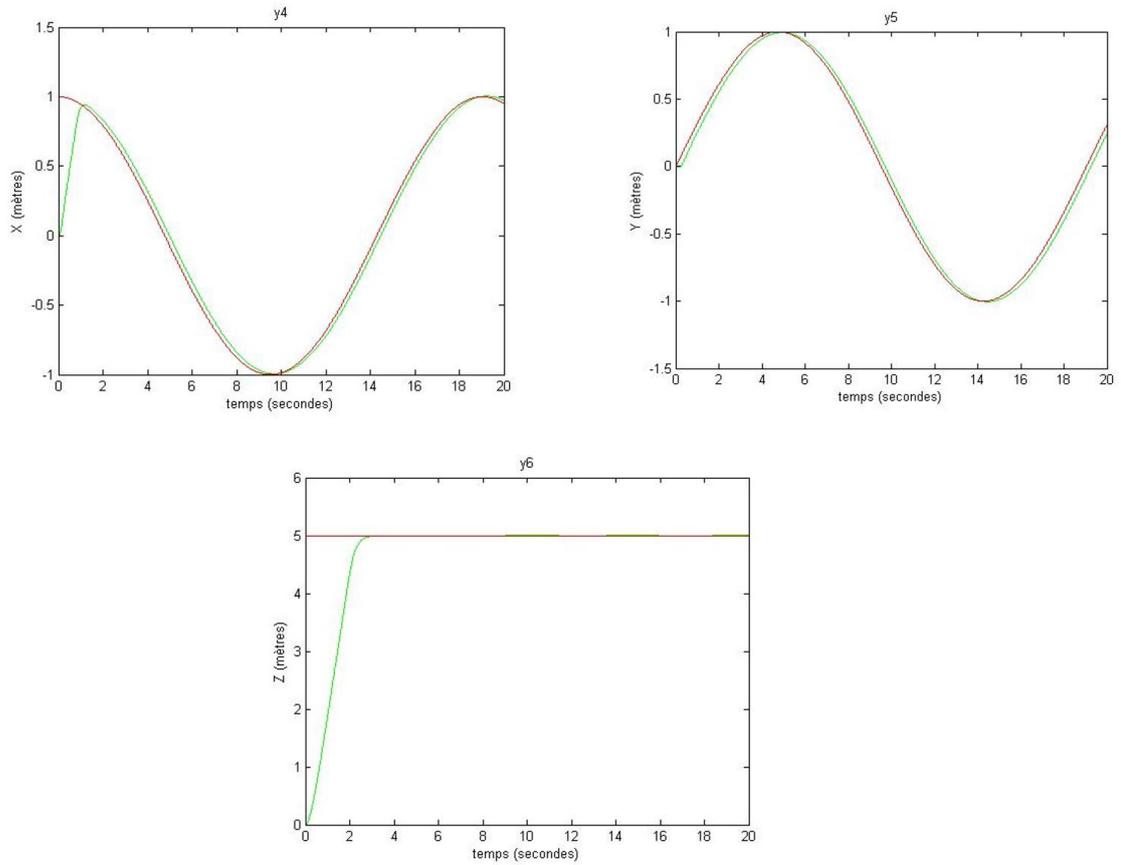


FIG. A.6 : Les sortie y1, y2 et y3

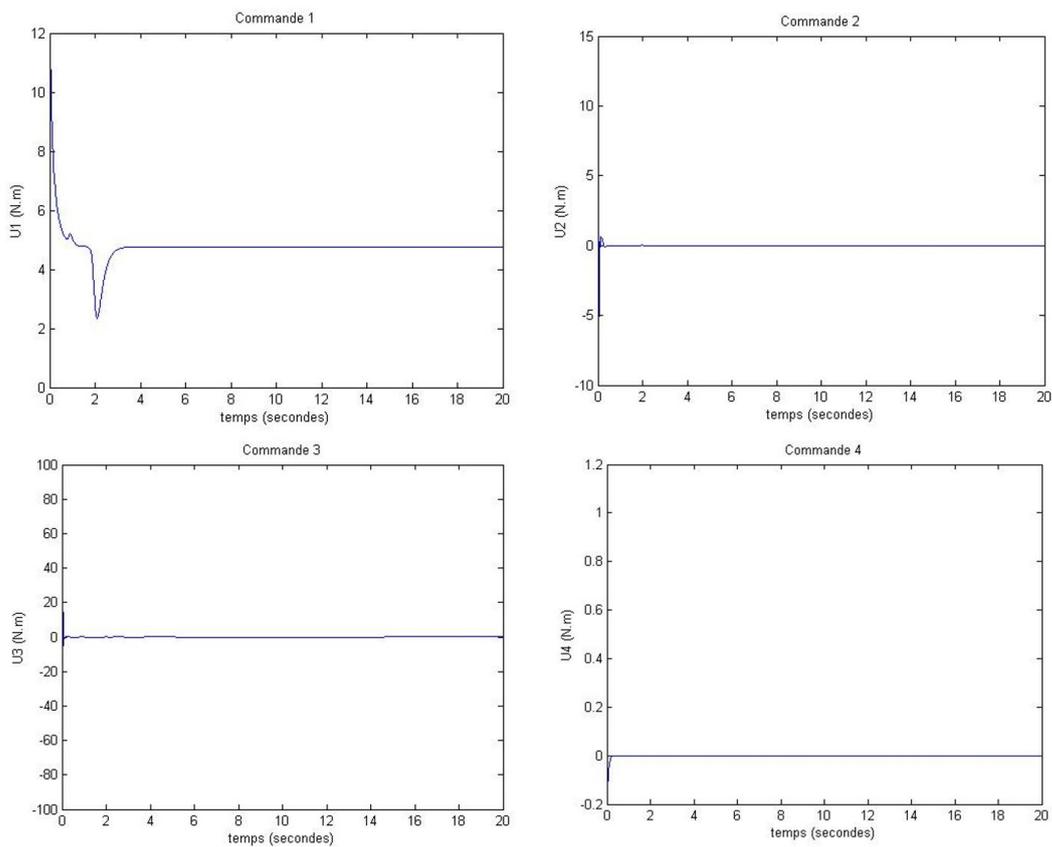


FIG. A.7 : Les signaux de commandes

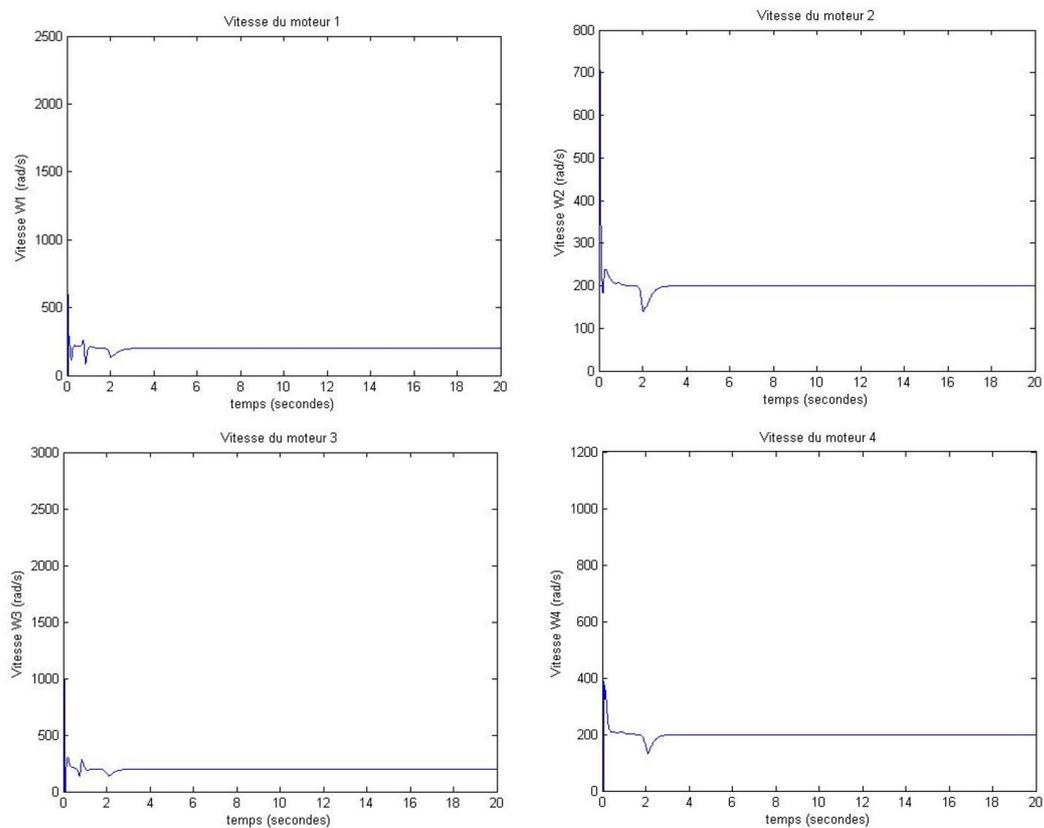


FIG. A.8 : Les vitesses des moteur

Le mode glissant arrive à rejoindre les références variables avec une dynamique hyper rapide tout en gardant une bonne précision de poursuite.

La poursuite obtenue est meilleure que celle envisagée dans le retour d'état.

Les signaux de commande présentent une évolution lisse et n'ayant pas de broutement ce qui confirme bien la validité de la fonction  $\tanh(x)$

### A.5 Conclusion

Le mode glissement rentre comme une manière pratique de synthèse des lois de commandes visant à exploiter directement les non linéarités du système, comme ca été traité précédemment l'élaboration de la fonction candidate en s'appuyant sur la surface de glissement a permis de satisfaire un objet de commande qui est la poursuite de trajectoire circulaire. L'exploitation des ces résultats rentre dans le cadre de la comparaison avec les approches PDC.