

Titre: Conception d'un quadrirotor à rotors inclinables pour le suivi de
Title: trajectoires agressives

Auteur: George Marian Popescu
Author:

Date: 2020

Type: Mémoire ou thèse / Dissertation or Thesis

Référence: Popescu, G. M. (2020). Conception d'un quadrirotor à rotors inclinables pour le
suivi de trajectoires agressives [Mémoire de maîtrise, Polytechnique Montréal].
Citation: PolyPublie. <https://publications.polymtl.ca/5476/>

Document en libre accès dans PolyPublie

Open Access document in PolyPublie

URL de PolyPublie: <https://publications.polymtl.ca/5476/>
PolyPublie URL:

Directeurs de recherche: Maxime Raison, & Sofiane Achiche
Advisors:

Programme: Génie mécanique
Program:

POLYTECHNIQUE MONTRÉAL

affiliée à l'Université de Montréal

**Conception d'un quadrirotor à rotors inclinables pour le suivi de trajectoires
agressives**

GEORGE MARIAN POPESCU

Département de génie mécanique

Mémoire présenté en vue de l'obtention du diplôme de *Maîtrise ès sciences appliquées*
Génie mécanique

Septembre 2020

POLYTECHNIQUE MONTRÉAL

affiliée à l'Université de Montréal

Ce mémoire intitulé :

**Conception d'un quadrirotor à rotors inclinables pour le suivi de trajectoires
agressives**

présenté par **George Marian POPESCU**

en vue de l'obtention du diplôme de *Maîtrise ès sciences appliquées*

a été dûment accepté par le jury d'examen constitué de :

Abolfazl MOHEBBI, président

Maxime RAISON, membre et directeur de recherche

Sofiane ACHICHE, membre et codirecteur de recherche

Ali SAFAEI, membre

REMERCIEMENTS

J'aimerais remercier chaleureusement mon directeur Maxime Raison et mon co-directeur Sofiane Achiche pour leurs soutiens tout au long du projet de recherche, leurs avis réfléchis et de m'avoir permis de faire un projet aussi audacieux. Je remercie également Ali Safaei et Abolfazl Mohebbi pour avoir accepté de participer à mon jury de maîtrise recherche.

RÉSUMÉ

Les quadrirotors sont des plateformes robotiques aériennes peu coûteuses et agiles. Plusieurs applications sont envisageables avec ces robots tels que l'exploration des mines ou les opérations de reconnaissance et sauvetage. Ces missions nécessitent de naviguer dans des environnements encombrés et imprédictibles. Le véhicule utilisé doit pouvoir éviter rapidement des obstacles tout en circulant à haute vitesse. Le quadrirotor étant sous-actionné est limité dans son agressivité puisqu'il doit s'incliner avant d'accélérer. De plus, les contrôleurs conventionnels utilisés ne prédisent pas le comportement qu'aura le véhicule durant la trajectoire en utilisant sa dynamique ce qui l'empêche de planifier assidument les manœuvres complexes.

Dans ce contexte, l'objectif principal de ce mémoire est de s'affranchir de ces deux limitations en développant un quadrirotor capable d'incliner ses moteurs pour accélérer plus rapidement et d'utiliser un contrôleur prédictif pour le suivi de trajectoire.

Plus spécifiquement, une modification au design conventionnel du quadrirotor est proposée par l'ajout d'un seul actuateur pour permettre des manœuvres agressives dans un seul axe. Puis, un ILQR qui est un contrôleur prédictif sans optimisation numérique, est développé. Celui-ci tient compte de l'état à jour du quadrirotor pour la linéarisation et la résolution du problème de contrôle optimal.

En premier lieu, le modèle dynamique du quadrirotor à moteurs inclinables est présenté. Puis, une loi de contrôle basé sur un schéma de contrôle en cascade avec une boucle régulant la dynamique en translation à l'aide d'un ILQR et une autre la dynamique en rotation avec un régulateur PD sont implémentées. Ensuite, la solution proposée est testée en simulation et comparée aux approches conventionnelles tant en termes de conception mécanique qu'en asservissement. L'erreur en suivi de trajectoire est diminuée de plus de 1483% avec un impact supérieur de l'ajout de l'inclinaison des moteurs. Enfin, un prototype expérimental est conçu avec des pièces électroniques et mécaniques standards et largement accessibles. La différence entre le design conventionnel et le quadrirotor à moteurs inclinables est étudiée sur des trajectoires agressives. L'erreur diminue de plus de 26% avec l'ajout d'un actionneur alors qu'en simulation pour la même trajectoire l'erreur diminue de 38% ce qui indique que la même tendance est conservée.

Les perspectives futures de cette recherche sont d'intégrer le problème de génération de trajectoire dans le contrôle optimal pour obtenir une trajectoire unifiée qui utiliserait pleinement les capacités dynamiques du quadrirotor avec rotors inclinables pour le suivi de trajectoires agressives.

ABSTRACT

Quadruped robots are cost-effective and agile aerial robotic platforms. Numerous applications are possible with these robots like mines exploration or search and rescue operations. Nevertheless, these missions require navigating through cluttered and unpredictable environments. The vehicle used for these operations must be able to avoid newly located obstacles fast while travelling at high speeds for time critical missions. Quadruped robots are underactuated systems and therefore limited in their overall maneuvers because they need to tilt their whole body before accelerating in a direction. Also, conventional controllers used with these systems don't predict the behavior of the vehicle during a trajectory by using the system's dynamics which prevents them from planning diligently complex maneuvers.

In this context, the main objective of this master thesis is to mitigate these two limitations by developing a quadruped robot able to tilt his motors' thrust to accelerate faster and to use a predictive controller for the trajectory tracking problem.

Specifically, a modification to the conventional quadruped mechanical system is proposed by adding a single actuator to enable aggressive motions in a single axis. Then, an ILQR, which is a predictive controller and does not require parameter optimization, is developed. The latter is a state-dependent controller who behaves as a nonlinear controller by considering the known updated state of the vehicle to solve the optimal control problem.

First, the dynamic model of the quadruped with tilting motors is found. Then, a control law based on a cascade control scheme with a loop for the translational dynamics regulated by an ILQR controller and another loop for the rotational dynamics with a PD controller is implemented. Afterwards, the proposed solution is tested in simulations and compared with conventional approaches in terms of mechanical design and control. Trajectory tracking error is reduced by more than 1483% with the tilting motors modification having a superior impact on performance. Finally, an experimental prototype is designed with standard electronic and mechanical pieces available off-the-shelf. The difference between the conventional design and the quadruped with tilting motors is studied on this custom-made quadruped on aggressive trajectories. The error has decreased by more than 26% by adding an actuator while in simulation for the same trajectory this error decrease by 38% which indicates that the same trend is maintained.

The perspectives of this research are to integrate the trajectory generation problem into the optimal control problem to obtain a unified trajectory that considers the quadrotor dynamics and more specifically its ability to tilt for aggressive trajectory tracking.

TABLE DES MATIERES

REMERCIEMENTS	III
RÉSUMÉ.....	IV
ABSTRACT	VI
TABLE DES MATIERES	VIII
LISTE DES TABLEAUX.....	XI
LISTE DES FIGURES.....	XII
LISTE DES SIGLES ET ABRÉVIATIONS	XV
CHAPITRE 1 INTRODUCTION.....	1
CHAPITRE 2 REVUE DE LITTÉRATURE	3
2.1 Concepts généraux	3
2.2 Limitations	6
2.2.1 Dynamique	6
2.2.2 Contrôle	7
2.3 Contrôleur pour suivi de trajectoires agressives	8
2.4 Multirotors à rotors inclinables	11
2.5 Conclusion.....	13
CHAPITRE 3 RATIONNELLE DU PROJET	14
3.1 Résumé de la problématique	14
3.2 Objectif général	14
3.3 Objectifs spécifiques	14
CHAPITRE 4 ARTICLE 1 : AGGRESSIVE AUTONOMOUS QUADROTOR FLIGHT IN INDOOR CLUTTERED ENVIRONMENTS USING ITERATIVE LQR AND TILTING ARMS	
16	16
4.1 Introduction	16

4.2	Methods	18
4.2.1	Modeling	19
4.2.2	Control.....	27
4.2.3	Experimental Setup	34
4.2.4	Analysis.....	37
4.3	Results	38
4.3.1	Simulations Results	38
4.3.2	Experimental Results.....	47
4.4	Discussion	55
4.5	Conclusion.....	58
CHAPITRE 5 MÉTHODES ET RÉSULTATS COMPLÉMENTAIRES		65
5.1	Architecture logicielle	65
5.1.1	Paquets	65
5.1.2	Nœuds.....	65
5.1.3	Messages	66
5.1.4	Sujets	66
5.1.5	Implémentation.....	66
5.2	Estimation de l'état du quadrirotor.....	69
5.3	Conception mécanique et limitations	71
CHAPITRE 6 DISCUSSION GÉNÉRALE		75
6.1	Synthèse de l'article	75
6.1.1	Simulation	75
6.1.2	Expérimental	76
6.2	Limites et perspectives	77

6.2.1	Design.....	77
6.2.2	Modèle dynamique	79
6.2.3	Contrôle	79
6.2.4	Planification de la trajectoire.....	81
6.2.5	Estimation d'état	82
6.2.6	Autonomie	83
CHAPITRE 7 CONCLUSION ET RECOMMANDATIONS		84
RÉFÉRENCES		86

LISTE DES TABLEAUX

Tableau 6.1. : Configurations du quadrirotor mises à l'essai en simulation	75
--	----

LISTE DES FIGURES

Fig. 2.1. : Quadrirotor AR.Drone 2.0 avec ses systèmes de coordonnées (© 2014 IEEE [7]).	4
Fig. 2.2. : Contrôle de l'orientation du quadrirotor en variant la vitesse de rotation des moteurs. ..	4
Fig. 2.3. : Différence de trainée entre un vol avec un quadrirotor sans et avec moteurs inclinables.	
.....	6
Fig. 2.4. : Quadrirotor traversant des cerceaux à haute vitesse (© 2011 IEEE [16])......	9
Fig. 2.5. : Trajectoire à travers une fenêtre étroite avec en bleu la phase de préparation et en rouge l'exécution (© 2017 IEEE [26])......	10
Fig. 2.6. : Drone omnidirectionnel sans rotors inclinables à gauche (© 2016 IEEE [31]) et quadrirotor avec rotors inclinables sur deux axes à droite [32]	12
Fig. 4.1 : (a) Inertial and quadrotor body coordinate systems. (b) Upper view of the vehicle with the lever arms L	20
Fig. 4.2 : Rotor coordinate system with the direction of a positive defined tilting angle α	21
Fig. 4.3 : Control structure overview.	27
Fig. 4.4 : Quadrotor experimental platform with arms tilted forwards and his various components	35
Fig. 4.5 : Overview of the control architecture.	37
Fig. 4.6. : Desired (dashed red), LQR conventional (solid yellow), iLQR conventional (solid magenta), LQR with tilt (solid green) and iLQR with tilt (solid blue) trajectories for slalom path.....	40
Fig. 4.7. : Desired (dashed red), LQR conventional (solid yellow), iLQR conventional (solid magenta), LQR with tilt (solid green) and iLQR with tilt (solid blue) Cartesian positions for slalom path.	41
Fig. 4.8. : Desired (dashed red), LQR conventional (solid yellow), iLQR conventional (solid magenta), LQR with tilt (solid green) and iLQR with tilt (solid blue) Cartesian velocities for slalom path.	42

Fig. 4.9. : LQR conventional (solid yellow), iLQR conventional (solid magenta), LQR with tilt (solid green) and iLQR with tilt (solid blue) angles for slalom path.	43
Fig. 4.10. : LQR conventional (solid yellow), iLQR conventional (solid magenta), LQR with tilt (solid green) and iLQR with tilt (solid blue) accelerations for slalom path.....	44
Fig. 4.13. : Desired (dashed red), LQR conventional in simulation (solid yellow), iLQR conventional in simulation (solid magenta), LQR with tilt in simulation (solid green), iLQR with tilt in simulation (solid blue), iLQR conventional in real (solid turquoise) and iLQR with tilt in real (solid blue) trajectories for narrow gap path; (b) Quadrotor prototype with tilting arms and iLQR performing the narrow gap trajectory. Note that the planned trajectory times are written.....	49
Fig. 4.14. : Desired (dashed red), LQR conventional in simulation (solid yellow), iLQR conventional in simulation (solid magenta), LQR with tilt in simulation (solid green), iLQR with tilt in simulation (solid blue), iLQR conventional in real (solid turquoise) and iLQR with tilt in real (solid blue) Cartesian positions for narrow gap path.....	50
Fig. 4.15. : Desired (dashed red), LQR conventional in simulation (solid yellow), iLQR conventional in simulation (solid magenta), LQR with tilt in simulation (solid green), iLQR with tilt in simulation (solid blue), iLQR conventional in real (solid turquoise) and iLQR with tilt in real (solid blue) Cartesian velocities for narrow gap path.	51
Fig 4.16. : LQR conventional in simulation (solid yellow), iLQR conventional in simulation (solid magenta), LQR with tilt in simulation (solid green), iLQR with tilt in simulation (solid blue), iLQR conventional in real (solid turquoise) and iLQR with tilt in real (solid blue) angles for narrow gap path.	52
Fig. 5.1. : Architecture ROS sur le quadrirotor.....	68
Fig. 5.2. : Schéma représentant la position des marqueurs rétro-réfléchissants et les vecteurs permettant de retrouver l'orientation du drone.	69
Fig. 5.3. : Liaison du bras avec l'essieu le système de distribution	71
Fig. 5.4. : Principaux composants du système de bras inclinables.....	72
Fig. 5.5. : Jeu autour d'une poulie de distribution.....	73

Fig. 5.6. : Accélérations brutes perçues par le PX4 en vol stationnaire.....74

LISTE DES SIGLES ET ABRÉVIATIONS

DARPA	Defense Advanced Research Projects Agency
DDL	Degrés De Liberté
DNN	<i>Deep Neural Network</i> (traduction : réseau de neurones profonds)
ESC	<i>Electronic Speed Controller</i> (traduction : contrôleur de vitesse électrique)
ILQR	<i>Iterative Linear Quadratic Regulator</i> (traduction : régulateur linéaire quadratique itératif)
IMU	<i>Inertial Measurement Unit</i> (traduction : centrale à inertie)
LQR	<i>Linear Quadratic Regulator</i> (traduction : régulateur linéaire quadratique)
LQT	<i>Linear Quadratic Tracking</i> (traduction : suiveur linéaire quadratique)
MPC	<i>Model Predictive Control</i> (traduction : commande prédictive)
PD	Proportionnel Dérivé
PID	Proportionnel Intégral Dérivé
PLA	<i>Polylactic Acid</i> (traduction : acide polyacitique)
RNN	<i>Recurrent Neural Network</i> (traduction : réseau de neurones récurrents)
UAV	<i>Unmanned Aerial Vehicle</i> (traduction : véhicule aérien non-habité)

CHAPITRE 1 INTRODUCTION

Le domaine des véhicules aérien non-habité (UAV) est en pleine expansion. Plus particulièrement, le quadrirotor est utilisé dans plusieurs applications comme la surveillance, l'exploration de zones sinistrées, la livraison de colis, la photographie aérienne et les opérations de recherche et sauvetage [1]. Sa structure mécanique simple, son faible coût, sa grande manœuvrabilité et sa capacité à se déplacer dans un environnement confiné [2] sont toutes de caractéristiques qui font de lui le candidat idéal pour ces applications. Pour exploiter pleinement les capacités du quadrirotor, les chercheurs s'intéressent plus particulièrement à la planification et la perception dans les environnements déstructurés et inconnues, la collaboration d'équipe multi-agents et à l'exécution de manœuvres agiles. La tendance actuelle est de développer un système robotique pleinement autonome navigant à haute vitesse et utilisant seulement des ressources disponibles à bord du véhicule. La *Defense Advanced Research Projects Agency* (DARPA) a lancé en 2015 le programme *Fast Lightweight Autonomy* qui encourage ces recherches [3].

Lors d'une catastrophe naturelle, il est essentiel de trouver rapidement les personnes piégées sous des débris ou dans des espaces confinés. Il est souvent impossible de se déplacer sur le sol à cause des éboulements et un véhicule aérien à faible coût est alors la solution idéale pour sauver des vies. Un UAV se déplaçant rapidement et précisément peut aussi être utile pour explorer des espaces inconnues et étroits comme des mines ayant des corridors aussi minces que 1.5m de diamètre [4]. Des manœuvres rapides peuvent être requises pour éviter un obstacle nouvellement localisé dans ces environnements incertains. Ces mêmes habiletés pourraient permettre d'éviter des objets mobiles lancés sur le véhicule [5] ou de naviguer dans des environnements très encombrés comme des forêts [6].

Dans ce contexte, ce mémoire adresse le problème de navigation à haute vitesse dans des espaces confinés avec un véhicule aérien. Plus spécifiquement, un quadrirotor capable de suivre avec précision des trajectoires agressives est développée.

Le chapitre 2 présente une revue de littérature en situant la problématique et les approches déjà employés pour la résoudre. Le chapitre 3 présente la rationnelle du projet de recherche. Les résultats principaux de cette recherche sont présentés au chapitre 4 dans l'article *Aggressive Autonomous Quadrotor Flight in Indoor Cluttered Environments using Iterative LQR and Tilting Arms* soumis au journal *Robotics and Autonomous Systems*. Au chapitre 5 sont expliqués des détails

d'implémentations complémentaires omis dans l'article. Au chapitre 6 une discussion générale est donnée. Le chapitre 7 présente une conclusion à ce mémoire et des recommandations sont données.

CHAPITRE 2 REVUE DE LITTÉRATURE

D'une part, cette revue de littérature vise à donner une base générale des concepts reliés aux quadrirotors pour permettre la compréhension du mémoire et des résultats obtenus. D'autre part, elle permet de positionner ce projet de recherche par rapport à l'état de l'art en matière de suivi de trajectoire agressive avec des quadrirotors et les différentes avancées dans ce domaine en pleine effervescence. Pour ce faire, les multirotors sont présentés ainsi que leur mode de fonctionnement général. Leurs limitations principales sont ensuite introduites. Puis, l'état de l'art sur le contrôle en suivi de trajectoires agressives, la conception de multirotors à rotors inclinables et le suivi de trajectoires avec ces derniers sont présentés dans les sections subséquentes de ce chapitre.

2.1 Concepts généraux

Les multirotors sont des plateformes aériennes populaires depuis que le coût et les dimensions des composantes électroniques nécessaires à leur construction ont diminués. Typiquement, ils ont une conception mécanique simple requérant seulement l'assemblage de composants peu dispendieux. Pour pouvoir se déplacer, ce robot a besoin de moteurs munis dans la majorité des cas d'hélices à angle fixe. Les moteurs avec leurs hélices sont les seules parties mobiles du robot. Ainsi, la conception mécanique est simplifiée comparativement à un hélicoptère qui nécessite un plateau cyclique pour fonctionner.

Dépendamment du nombre de moteurs fixé sur la plateforme robotique, le multirotor peut être un octorotor (8), un hexarotor (6) ou un quadrotor (4) la plus populaire des configurations. Celui-ci possède le nombre minimum d'actuateurs nécessaires pour contrôler les 4 degrés de liberté (DDL) indépendants du robot, (la translation en x , y et z et l'angle de lacet). La Fig. 2.1 illustre un quadrirotor populaire parmi les chercheurs, le AR.Drone 2.0 avec son système d'axe. Les angles de roulis φ , tangage θ et lacet ψ sont définis comme les rotations autour des axes inertielles x_W , y_W et z_W respectivement. Les axes x_W , y_W et z_W sont fixés sur le centre de masse du robot et se déplacent avec lui.



Fig. 2.1. : Quadrirotor AR.Drone 2.0 avec ses systèmes de coordonnées (© 2014 IEEE [7])

La poussée générée par les moteurs est contrôlable en variant la vitesse de rotation de ceux-ci avec des contrôleurs de vitesses électriques (ESC). L’orientation du véhicule est quant à elle contrôlable en variant les forces générées par chacun des moteurs par rapport aux autres. Puisqu’une hélice qui tourne ne génère pas seulement une force de portance vers le haut, mais également un moment dû à la traînée [8] l’angle de lacet peut également être contrôlé. La Fig. 2.2 illustre comment l’alternance de l’amplitude des forces générées par les moteurs peut changer l’orientation du quadrirotor.

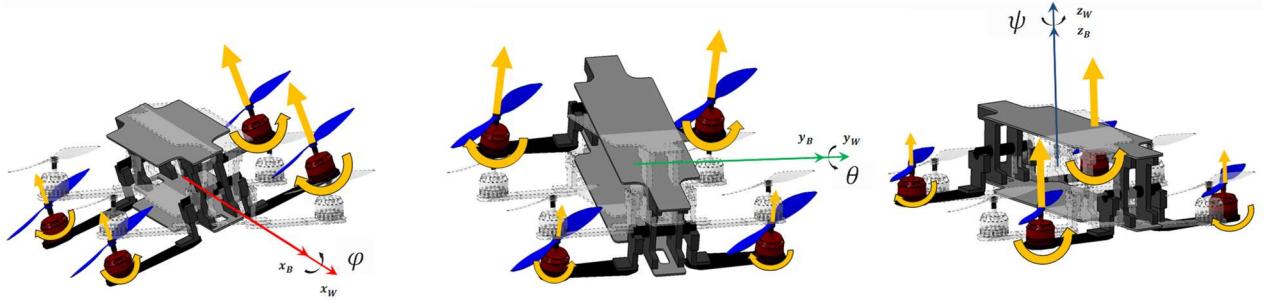


Fig. 2.2. : Contrôle de l’orientation du quadrirotor en variant la vitesse de rotation des moteurs

Mise à part leur conception mécanique simple, les quadrirotors ont l’avantage de décoller et atterrir verticalement, ce qui leurs permet d’atteindre des zones étroites. De plus, ils peuvent rester dans les airs sans avoir besoin de se déplacer ce qui peut s’avérer utile pour l’inspection et la surveillance. Ils sont également une plateforme de choix pour des manœuvres agiles puisque

dépendamment du choix des moteurs, il est relativement facile d'atteindre des ratios de portance versus poids assez élevés.

Les quadrirotors sont des systèmes mécaniques instables, c'est-à-dire qu'une force constante appliquée aux actionneurs n'est pas suffisante pour stabiliser le système en tout temps. De par sa nature, une minime déviation de sa position de vol stationnaire déstabilise le véhicule qui ne reviendra pas à sa position d'équilibre sans une stratégie de contrôle adéquate. La stabilisation est assurée avec une haute fréquence de contrôle pour constamment faire revenir le quadrirotor vers son unique point d'équilibre ponctuel, le vol stationnaire et plus spécifiquement lorsque les moteurs sont orientés vers le haut.

Les quadrirotors sont également des systèmes mécaniques sous-actionnées possédant seulement 4 actionneurs pour 6 DDLs. Tous les degrés de liberté du quadrirotor ne peuvent pas être contrôlés indépendamment. En effet, la position cartésienne et l'angle de lacet peuvent être commandés directement, mais lorsqu'on change l'un des deux degrés de liberté restant, soit l'angle de roulis et de tangage, la position du quadrirotor est affectée. N'ayant pas assez d'entrées de commande dans sa configuration originale, le quadrirotor ne peut pas suivre une trajectoire arbitraire dans son espace vectoriel. De plus, la sous-actuation empêche la linéarisation complète du modèle dynamique du quadrirotor [1]. La sous-actuation n'est pas un problème en vol libre, puisque seulement les mouvements de translation doivent être contrôlés pour éviter de rentrer en collision avec un obstacle sans se soucier des angles de tangages et de roulis du véhicule. Par contre, puisque le quadrirotor ne peut exercer des forces arbitrairement orienté indépendamment de son orientation, il est désavantage pour des tâches d'interaction avec l'environnement comme la préhension d'objets [9] ou de contact [10]. Pour ce type de tâche, des hélicoptères [11] ou des multirotors avec des moteurs inclinables [12] vont être un choix plus adapté.

La stratégie de contrôle la plus répandu sur le quadrirotor est basée une boucle de rétroaction en cascade. Le système est divisé en 2 sous-systèmes : une boucle interne qui gère la dynamique en orientation et une boucle externe qui gère la dynamique en translation du véhicule avec chacun un contrôleur de rétroaction indépendant. En pratique, la boucle interne roule à plus haute fréquence, environ 5 à 10 fois plus vite, que la boucle externe [1]. Finalement, les deux boucles sont connectées en utilisant la boucle externe pour envoyer une référence à suivre à la boucle interne.

Cette séparation est possible puisque l'échelle de temps de la dynamique rotationnelle est plus petite que celle en translation [13]. Néanmoins, le contrôle direct sur l'attitude de l'appareil en tant que tel est perdu et l'accélération voulue est limité par le taux de roulis ou de tangage.

2.2 Limitations

La sous-actuation est la principale limitation du quadrirotor pour accomplir des manœuvres plus agiles. Intimement liée à cette particularité, le type de contrôle employé peut également être un facteur limitant. Ces deux aspects sont discutés dans les deux sections suivantes.

2.2.1 Dynamique

Pour changer sa position, le quadrirotor doit préalablement s'incliner pour orienter son vecteur de force dans la direction voulue. Ce couplage entre le mouvement de rotation et le mouvement de translation dégrade sa capacité à réagir et à se déplacer rapidement en raison du temps de réponse de la boucle d'orientation. Des UAVs lourds comme l'hexacoptère AscTec NEO avec une masse de 3.42 kg a une constante de temps pour atteindre l'orientation voulue de 0.1901 sec [14]. Un quadrirotor plus léger comme le AscTec hummingbird a quant à lui une constante de temps de 0.281 sec [15]. Pour circuler à des vitesses aussi élevées que 2.6 m/s, des angles aussi élevés que 40° sont nécessaires [16]. D'une part, en s'inclinant le quadrirotor occupe plus d'espace que nécessaire ce qui pourrait le rapprocher d'éventuels obstacles lorsqu'il circule dans des environnements encombrés. D'autre part, à haute vitesse, la traînée est plus élevée sur le véhicule lorsqu'il s'incline ce qui peut le ralentir [17], voir Fig. 2.3.

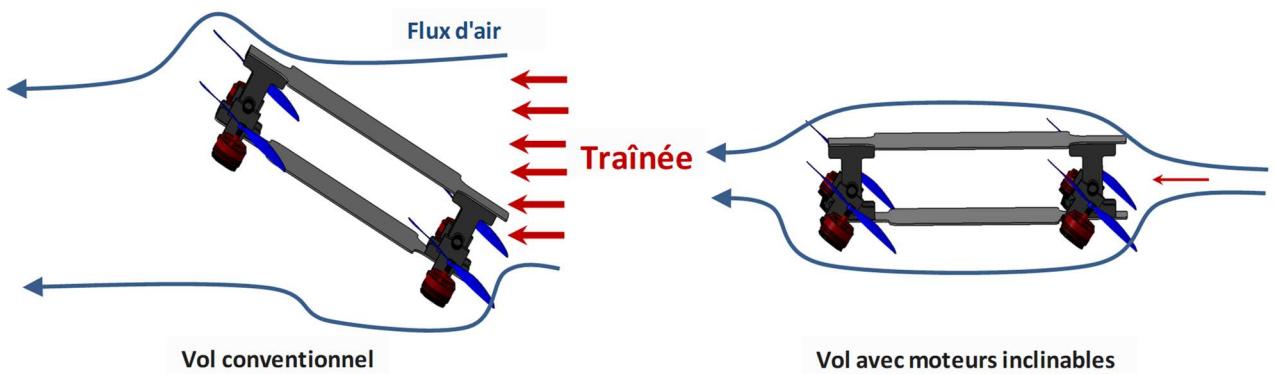


Fig. 2.3. : Différence de trainée entre un vol avec un quadrirotor sans et avec moteurs inclinables

Plus le véhicule s'incline pour effectuer des manœuvres agressives, plus il sort de son enveloppe linéaire ce qui nécessite de tenir compte des non-linéarités inhérentes à ces orientations agressives dans la conception de la loi de commande. Par exemple, suite à une manœuvre où le véhicule est sorti de ses limites linéaires, il est nécessaire d'adoucir les gains du contrôleur pour récupérer de cette orientation [18]. Les manœuvres agressives sont courtes et elles ne peuvent être rapidement relancées ce qui limite l'agressivité globale du véhicule sur une plage de temps plus grande.

2.2.2 Contrôle

Hormis les schémas de contrôle les plus communs composés de deux boucles de contrôle en cascade, plusieurs autres approches ont été étudiées dans la littérature. Pour suivre des trajectoires agressives, d'une part, la dynamique non-linéaire du quadrirotor doit être prise en compte. Une approche est d'utiliser un contrôleur LQR (Régulateur Linéaire Quadratique), mais de linéariser le modèle dynamique à un pas de temps donné pour tenir compte du changement d'orientation du véhicule [13]. Le contrôleur respecte alors toujours son enveloppe linéaire puisqu'elle est mise à jour durant la trajectoire.

D'autre part, les commandes aux actuateurs doivent être anticipées tout au long de la trajectoire sans se baser uniquement sur l'erreur, l'une des limitations du contrôleur le plus commun, le contrôleur PID (Proportionnel Dérivé Intégral). Dans [16, 19], des termes de contrôle en boucle ouverte sont trouvés à partir de la trajectoire à suivre et utilisés avec une boucle fermée pour avoir une longueur d'avance sur la commande à effectuer.

Les méthodes prédictives ont également la capacité de tenir compte des futures positions désirées et des positions prédites du robot. Par exemple, un MPC (Modèle de Contrôle Prédictif) calcule explicitement le comportement du robot sur un horizon de temps. Le choix de la trajectoire à suivre et des contrôles à appliquer au temps présent tient compte de l'impact de ce choix sur les temps futures et si celui-ci va engendrer des difficultés [20]. Dans un scénario de manœuvre agressive, cela implique de ne pas atteindre des orientations au-dessus de l'enveloppe linéaire du quadrirotor pour suivre à tout prix la trajectoire sans pouvoir stabiliser le quadrirotor à la fin. Un MPC a une meilleure robustesse, suivi de trajecaStoire et précision qu'un contrôleur LQR [21]. Une étude comparative en simulation a montré qu'un MPC surpassé un LQR avec moins d'erreur en suivi de

trajectoire, moins d'effort de contrôle et moins de fluctuations sur le contrôle ce qui le rend plus sécuritaire pour le système commandé [22]. De plus, en utilisant des gains variant dans le temps pour un contrôleur LQT (Linéaire Quadratique Suiveur) plutôt qu'un LQR, des résultats comparables à un MPC et sa capacité anticipative sont obtenus, mais avec moins de puissance de calcul requise [23].

2.3 Contrôleur pour suivi de trajectoires agressives

La littérature sur les trajectoires agressives ne présente pas une définition officielle pour qualifier un quadrirotor d'agressif. Dans [24], un quadrirotor de 1 kg suit une trajectoire agressive avec des vitesses de 8.2 m/s et des accélérations de 19.62 m/s². Dans [25], un quadrirotor de 250 g atteint des vitesses de 4.5 m/s et des accélérations de 3.68 m/s². Dans [16], un quadrirotor de 500 g atteint des vitesses de 3 m/s et des accélérations de 5 m/s² en traversant un cerceau lancé dans les airs. Dans [26], un quadrirotor de 830 g atteint des vitesses de 3 m/s et des accélérations de 5 m/s² en voltant à travers une fenêtre étroite. Dans [27], un quadrirotor de 1.65 kg atteint des vitesses de 8 m/s en navigant dans un environnement intérieur encombré. Ces études utilisent toutes des quadrirotors pour le suivi de trajectoires agressives puisqu'ils ont moins de rotors et sont plus légers ce qui leurs permettent d'atteindre des vitesses et accélérations plus élevées que d'autres types de multirotors [8]. Dans le cadre de ce travail, une trajectoire est considérée agressive si elle possède des accélérations supérieures à 5 m/s² ou des vitesses dépassant 3 m/s ou si elle requiert que le quadrirotor sorte de son enveloppe de vol linéaire. Ceci se produit lorsque l'angle de tangage ou l'angle de roulis dépasse 30° typiquement. Plusieurs caractéristiques d'une trajectoire agressive peuvent se cumuler pour un effet encore plus agressif exploitant pleinement les capacités de vol du quadrirotor.

La possibilité d'effectuer des trajectoires agressives avec des quadrirotors a déjà été étudiée dans la littérature scientifique. En générant une trajectoire complexe par décomposition séquentielle, il est possible de traverser des fenêtres verticales à plus de 45° et même de se percher sur un mur. Cette approche décompose les différentes phases de la trajectoire en donnant un objectif distinct pour chacune d'elle comme suivre une position, suivre un angle puis planer pour un retour au calme. Certains phases sont plus ou moins agressives ce qui permet d'avoir des gains adaptés pour

chacune d'elle [18]. Néanmoins, les trajectoires sont difficilement généralisables puisqu'il faut préparer un élan par exemple pour passer à travers les fenêtres verticales.

D'autres approches utilisent le fait que le système dynamique du quadrirotor est différentiellement plat, c'est-à-dire que les états et les entrées de commandes peuvent être écrits comme des fonctions algébriques de 4 sorties plates et un nombre finies de leurs dérivées [28]. Ainsi, une trajectoire agressive polynomiale en position et en lacet peut permettre de trouver les termes anticipatifs à fournir à un contrôleur de rétroaction [16, 19]. Cette technique requiert donc de générer une trajectoire différentiable qui sera ensuite suivi agressivement. L'algorithme de contrôle a une structure en cascade puisque les angles à suivre sont trouvés et suivis, puis les positions. Les accélérations possibles dépendent des taux de rotation et seulement les trajectoires polynomiales dérivable 4 fois peuvent être suivies ce qui limite la performance du quadrirotor. Néanmoins, des vitesses de 3m/s ont pu être atteintes en moins de 0.6 sec tout en passant à travers un cerceau lancé dans les airs d'un diamètre de 0.4m [16]. Des trajectoires périodiques sont également présentées où le quadrirotor traverse trois cerceaux du même diamètre atteignant une vitesse maximale de 2.6 m/s et des angles de tangages et de roulis de 40°, voir Fig. 2.4.

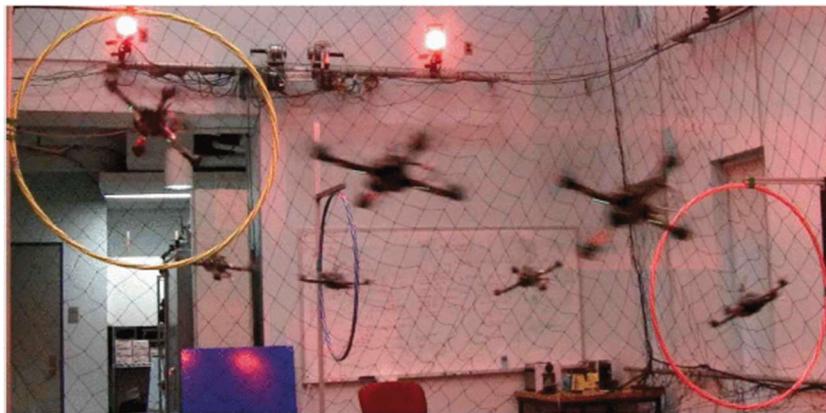


Fig. 2.4. : Quadrirotor traversant des cerceaux à haute vitesse (© 2011 IEEE [16])

La même loi de contrôle est employée dans [27], mais avec une stratégie différente pour générer des trajectoires polynomiales à fournir au contrôleur en boucle ouverte. Des vitesses allant jusqu'à 8 m/s sont atteintes dans des environnements intérieurs encombrés.

Ce travail est davantage étendu avec tous les calculs effectués à bord incluant l'estimation d'état. Des accélérations de 5m/s^2 sur des trajectoires de type slalom sont atteintes avec un contrôleur

géométrique non-linéaire utilisant la même boucle de contrôle en cascade [25]. D'autres chercheurs, quoique focalisés sur le problème d'estimation d'état à bord du véhicule, arrivent à traverser une fenêtre étroite avec un jeu de seulement 12.5cm et 8cm de chaque côté du quadrirotor [26]. Des accélérations de 5m/s^2 et des vitesses de 3m/s sont atteintes lors des traversés les plus agressives. Or, l'approche présentée comporte deux inconvénients. D'une part, pour préparer la trajectoire et le passage de la fenêtre, le quadrirotor doit reculer pour se donner un élan puisqu'il ne peut produire des accélérations assez grandes en si peu de temps. La Fig. 2.5 illustre en bleu la phase de préparation et en rouge la phase d'exécution. La première phase augmente donc le temps total de la trajectoire inutilement en plus de nécessiter plus d'espace pour la manœuvre ce qui n'est pas toujours possible dans des environnements encombrés. D'autre part, un système de récupération automatique [29] est nécessaire pour stabiliser le quadrirotor après cette manœuvre agressive.

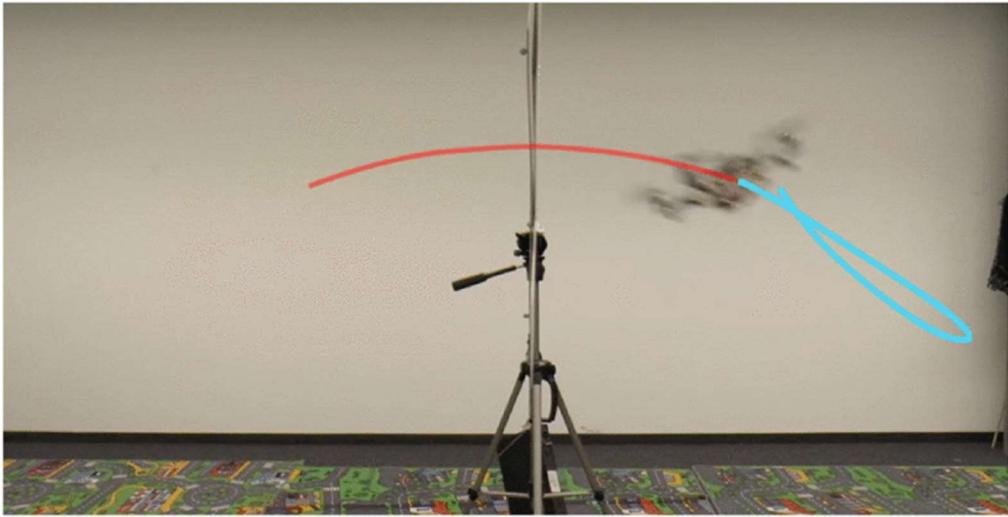


Fig. 2.5. : Trajectoire à travers une fenêtre étroite avec en bleu la phase de préparation et en rouge l'exécution (© 2017 IEEE [26])

Certains vont plus loin en rajoutant une rétroaction sur la vitesse des moteurs et les vitesses et accélérations angulaires du quadrirotor. Ainsi, une accélération maximale de 19 m/s^2 avec une vitesse maximale de 8.2 m/s et une erreur moyenne en suivi de trajectoire de 4 cm sont obtenus [24]. Néanmoins, le volume de vol est de 6.5 m de long, 6.5 m de large et 1.5 m de haut ce qui laisse plus d'espace pour accélérer et décélérer.

Une autre approche pour le suivi de trajectoire agressive est d'utiliser de la commande prédictive, comme un MPC. Celui-ci peut être linéaire ou non-linéaire. L'avantage de cette approche est l'anticipation de la position future du drone ce qui permet de se préparer à faire une trajectoire agressive. De plus, en gardant le modèle non-linéaire du quadrirotor, les trajectoires avec un grand angle de roulis et/ou de tangage peuvent être mieux suivies. L'erreur moyenne en suivi de trajectoire sur la position est de 10.8 cm pour un MPC linéaire et 7.1 cm sur un MPC non-linéaire sur des trajectoires ne dépassant pas 2 m/s [14]. D'autres contrôleurs linéarisent itérativement la dynamique du système autour de la trajectoire de référence. Ainsi, les points de la trajectoire agressive à suivre sont des contraintes molles dans la formulation du contrôleur et celui-ci peut les violer légèrement si la trajectoire n'est pas dynamiquement faisable. Cela permet de continuellement s'adapter en suivant de manière optimale la trajectoire voulue malgré un décalage vis-à-vis de la trajectoire en cours. Les expériences démontrent que des tâches qui tirent parti de toute la dynamique du système peuvent être accomplis. Des points à suivre incluant tous les degrés de libertés du quadrirotor peuvent être donnés et suivis avec des erreurs de moins de 6 cm [30].

2.4 Multirotors à rotors inclinables

Une autre piste pour suivre des trajectoires agressives en dehors de l'enveloppe linéaire du quadrirotor est de résoudre le problème de sous-actuation de ce système dynamique. Générer des forces dans des directions arbitraires indépendamment de l'assiette du drone le rend plus agile puisqu'il peut accélérer immédiatement dans une direction voulue sans s'incliner.

Pour pouvoir générer des forces dans n'importe quelle direction, des actionneurs doivent être rajoutés au système mécanique. Deux options sont envisageables : rajouter des rotors générant une poussée dans des directions différentes aux autres rotors ou incliner les rotors déjà présents, voir la Fig. 2.6. La première option permet de résoudre le problème de sous-actuation sans ajouter d'autres parties mobiles excepté les hélices, mais la conception mécanique est plus complexe puisqu'il y a beaucoup de rotors à installer. De plus, en utilisant un contrôleur non-linéaire et en conservant la boucle de contrôle en cascade, les résultats montrent qu'il existe un couplage entre la dynamique en translation et en rotation [31]. La deuxième option implique de rajouter des servomoteurs pour rendre les rotors inclinables. Cette option sera étudiée plus en détail puisqu'elle nécessite d'acheter moins de moteurs et de garder en majorité le design mécanique simple du quadrirotor.

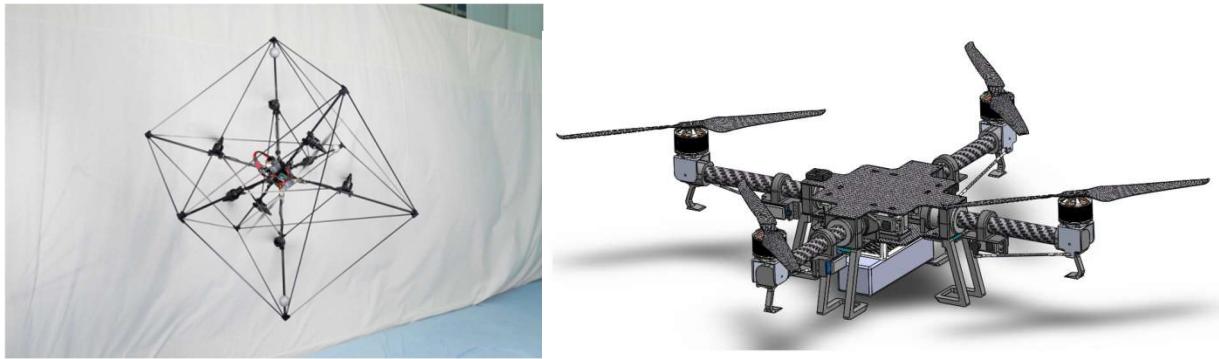


Fig. 2.6. : Drone omnidirectionnel sans rotors inclinables à gauche (© 2016 IEEE [31]) et quadrirotor avec rotors inclinables sur deux axes à droite [32]

L’ajout d’uniquement un nouvel axe de rotation permet de gagner en agilité tout en minimisant la complexité du nouveau système mécanique et du poids. Par exemple, l’angle de tangage est découpée par rapport au mouvement vers l’avant dans un étude ce qui permet d’avancer sans s’incliner [33]. L’étude montre aussi que le quadrirotor a la capacité de freiner dans l’air, plus spécifiquement de passer d’une vitesse de croisière de 10 m/s à zéro en moins d’une seconde. Par contre, seulement un contrôle régulant la vitesse en translation est présenté ce qui ne permet pas de contrôler l’appareil en position et aucun essai n’est effectué avec le système réel. D’autres recherches présentent des quadrirotors ou hexarotors avec des servomoteurs permettent de contrôler un [34, 35] ou deux angles [32, 36] sur le même rotor. Pour les contrôler, certains gardent l’architecture de contrôle en cascade en utilisant la propriété du système d’être différentiablement plat pour trouver l’assiette du quadrirotor désirée en tout temps [37]. Ainsi, le quadrirotor peut encore s’inclinée pour avancer tout en utilisant ces rotors inclinables pour un effet combiné ce qui a le potentiel d’améliorer la rapidité du mouvement en translation. Dans [38], un quadrirotor avec un seul axe de rotors inclinables a pu atteindre des positions désirés plus rapidement qu’un design conventionnel. D’autres conçoivent un drone complètement omnidirectionnel avec des forces et des moments de rotations actionnées indépendamment [34] qui parvient à se maintenir contre un mur.

Des essais expérimentaux ont montré que des rotors inclinables améliorent l’agilité du quadrirotor lorsqu’il navigue dans des espaces étroits. La distance de dégagement nécessaire pour un quadrirotor sous-actionné est de 2.65m alors qu’elle est de 1.25m quand des rotors inclinables sont

utilisés [32]. Un autre étude a montré que permet aux bras du quadrirotor de s'incliner permet d'atteindre des accélérations suffisantes pour jouer au tennis de table [39].

Si plusieurs méthodes de contrôles communes pour des quadrirotors sous-actionnées comme la linéarisation de la commande de retour avec découplage de la dynamique [40] ou une régulation PID sont appliquées [37], elles ne sont pas mises à l'épreuve sur des trajectoires agressives. En termes de performances en suivi de trajectoire agressives, les études menées jusqu'à présent n'ont pas étudié ce dont un drone avec rotors inclinables peut accomplir. Des trajectoires en forme de huit ont été suivies avec une erreur de 4 cm en position, mais avec des vitesses et accélérations ne dépassant pas 0.2 m/s et 0.05 m/s² respectivement [40].

2.5 Conclusion

En somme, des trajectoires agressives impliquant de grandes accélérations et vitesses sur un court laps de temps et dans des environnements encombrés n'ont pas encore été accomplis avec un quadrirotor. Deux pistes sont envisageables pour exploiter pleinement l'agilité qu'un quadrirotor peut avoir :

- rajouter des actionneurs pour avoir des rotors inclinables qui puissent orienter le vecteur de force appliqué au véhicule dans une direction donnée rapidement et,
- utiliser une méthode de contrôle prédictive pour tenir compte des difficultés à venir durant une manœuvre agressive et prendre des décisions en conséquence sur les contrôles à appliquer à l'instant présent.

CHAPITRE 3 RATIONNELLE DU PROJET

3.1 Résumé de la problématique

Le plein potentiel des quadrirotors en suivi de trajectoires agressives reste encore inexploité. Or, ils seraient les plateformes idéales pour des opérations de sauvetage dans des environnements incertains et encombrés. Effectuer ces opérations rapidement est crucial pour la survie des personnes en danger. Un quadrirotor agile pourrait également explorer des zones difficiles d'accès comme des mines sans risquer de vie humaine.

Plusieurs limitations empêchent le quadrirotor de faire des manœuvres agressives dans des espaces confinés. D'une part, il est sous-actionné, ce qui l'empêche d'accélérer dans n'importe quelle direction instantanément. D'autre part, les lois de contrôle employées dans la littérature sont majoritairement non prédictives. Les commandes envoyées aux actionneurs à chaque instant ne tiennent alors pas compte de l'état futur du quadrirotor pour négocier au mieux une trajectoire difficile.

3.2 Objectif général

L'objectif général du travail présenté dans ce mémoire est de concevoir un nouveau type de quadrirotor avec un contrôleur qui puisse suivre des trajectoires agressives dans des environnements encombrés. L'idée est d'innover sur deux fronts : changer la conception mécanique originale du quadrirotor et choisir une loi de contrôle adaptée pour surpasser ce qui est atteignable en termes d'agressivité avec un quadrirotor jusqu'à présent dans la littérature.

3.3 Objectifs spécifiques

Pour accomplir l'objectif général de ce projet, plusieurs sous-objectifs doivent être accomplis.

1. Développer une plateforme robotique adaptée.
 - 1.1. Conception du circuit électrique.
 - 1.2. Concevoir les pièces mécaniques personnalisées.
2. Développer une loi de contrôle pour le quadrirotor.

- 2.1. Développer une loi de contrôle en temps réel pour le suivi de trajectoires agressives.
- 2.2. Conception d'un algorithme calculant des trajectoires de références à suivre en temps réel.
3. Tester les performances du quadrirotor en situation réel.
 - 3.1. Concevoir un système expérimental adapté.
 - 3.2. Conception d'une stratégie pour obtenir l'estimation d'état du quadrirotor en temps réel.
 - 3.3. Assembler la plateforme robotique.
 - 3.4. Mettre à l'essai le quadrirotor développé sur des trajectoires agressives et évaluer sa performance.

CHAPITRE 4 ARTICLE 1 : AGGRESSIVE AUTONOMOUS QUADROTOR FLIGHT IN INDOOR CLUTTERED ENVIRONMENTS USING ITERATIVE LQR AND TILTING ARMS

The following article was submitted to Robotics and Autonomous Systems on August 8th, 2020.

George Popescu, Maxime Raison, Radu Popescu, David Saussié, Sofiane Achiche

Abstract

Today, the aggressiveness of multirotors in confined environments, such as indoor and cluttered spaces with obstacle avoidance, is limited. This is mainly due to two reasons: 1. the underactuation of the system, and 2. they use non-predictive control schemes, which prevents them from anticipating the next move in order to track a challenging trajectory. The objective of this paper is to develop an autonomous aggressive quadrotor for confined environments. The main originality of the proposed method is the combination of 1. a novel quadrotor design with tilting arms, and 2. a state-dependent iterative Linear Quadratic Control (iLQR). This method is compared to a non-predictive control approach such as state-dependent Linear Quadratic Control (LQR), and/or without the use of tilting arms on aggressive trajectories in simulation and on an experimental prototype. The combination of iLQR and tilting arms improves the position root mean square error (RMSE), by 1483%, compared with the LQR and without the use of tilting arms when the quadrotor is following aggressive trajectories with accelerations and velocities up to 12 m/s^2 and 3 m/s respectively. The future prospects are to further exploit the aggressive trajectory following potential of quadrotors.

Keywords

Quadrotors, control, aggressive trajectories

4.1 Introduction

Unmanned Aerial Vehicles (UAVs) are known for their simple mechanical design, low cost, high maneuverability and ability to travel in small confined spaces [1]. Therefore, the autonomous flight control of such robots has been an active and challenging research topic with diverse applications in surveillance, exploration of disaster areas, parcel delivery, aerial

photography and search and rescue operations [2]. Exploring indoor and confined environments autonomously such as mines requires precise tracking in corridors as small as 1.5 m in diameter [3]. Thus, fast motions with obstacle avoidance are required in such unpredictable environments. Other applications include avoid moving objects thrown at the vehicle [4] or navigating through a highly cluttered spaces such as a forest [5].

Today, multirotors' aggressiveness in these confined environments is limited. This limitation is mainly due to two reasons: 1. the underactuation of the system [2] and 2. they use non-predictive control schemes, which prevents them from anticipating the next move in order to track a challenging trajectory [6]. Indeed, both state-of-the-art surveys, including advancements and limitations, will be developed in the next paragraphs.

The focus of this paper is on quadrotors more specifically because they have less rotors and are lighter which enables them to reach higher speeds and accelerations [7]. Studies about aggressive motions don't provide a definition for aggressivity. In [8], a 1 kg quadrotor reaches speeds of 8.2 m/s and accelerations of 19.62 m/s², in [9] a 250 g quadrotor reaches speeds of 4.5 m/s and accelerations of 3.68 m/s², in [10] a 500 g quadrotor reaches speeds of 3 m/s and accelerations of 5 m/s² while flying through a thrown hoop, in [11] a 830 g quadrotor reaches speeds of 3 m/s and accelerations of 5 m/s² while flying through a narrow gap and in [12] a 1.65 kg quadrotor reaches speeds of 8 m/s while navigating in a dense indoor environment. Therefore, in this paper, a trajectory is considered as aggressive if the speeds and accelerations along it are above 3 m/s and 5 m/s² respectively. In order to follow aggressive trajectories, the quadrotor has to generate adequate forces and thus accelerations. It is considered aggressive if it can follow the above defined trajectories.

The underactuation of the quadrotor system limits its aggressiveness in confined environment due to the large attitude angles needed to generate high translational accelerations. Indeed, the quadrotor, having less control inputs than degrees of freedom, cannot direct its thrust vector instantaneously in a desired direction. The coupling between the rotational and the translational motion degrades its capacity to react and move fast due to the response time of the attitude loop. Heavy UAVs such as the AscTec NEO hexacopter with 3.42 kg of mass have a time constant up to 0.1901 sec [13] and 0.281 sec [14] for lighter ones. In [15], a quadrotor with

one axis tilting rotors was shown to reach waypoints faster than the conventional design. Air breaking capabilities have also been shown by decelerating the quadrotor from a cruise speed of 10 m/s to zero in less than a second [16]. Furthermore, angles as high as 40° are required to achieve speeds up to 2.6 m/s [10] which can bring the quadrotor closer to potential obstacle when circulating in narrow environments. Experimental tests show that tilting rotors provide increased agility when flying through narrow gaps. The clearing distance required for conventional underactuated quadrotor is 2.65 m while it's only 1.25 m when using tilting [17].

Quadrotors usually use non-predictive control schemes [18], which prevents them from anticipating the next move in order to track a challenging trajectory [6]. Predictive methods have the capacity to take into account the future desired and predicted positions of the robot. Thereby, aggressive maneuvers could be anticipated and better handled with less tracking error compared to non-predictive control. A predictive control method such as Model Predictive Control (MPC) has better tracking ability, robustness and precision than LQR (Linear Quadratic Regulator) control [19]. The work in [13] shows that MPC can follow trajectories of up to 2 m/s with an average error in position of 7.1 cm. Other approaches use the flat outputs of the dynamic model of the quadrotor to compute polynomial trajectories and find feedforward terms which enable the quadrotor to predict the challenging portions along the trajectories. Velocities of up to 3 m/s can be reached in less than 0.6 sec, while flying through a thrown hoop of 0.4 m in diameter [10]. This work is further extended with all computation made onboard and trajectories such as a slalom are followed at accelerations up to 5m/s^2 [9].

The objective of this paper is to develop an autonomous aggressive quadrotor for confined environments. The quadrotor aggressiveness is improved by solving the problems of underactuation inherent to the conventional design and non-predictive capabilities of common control laws.

4.2 Methods

The main originality of the proposed method is the combination of 1. a novel quadrotor design with tilting arms, and 2. a state-dependent iLQR. Compared to previous approaches [8,9,20], our solution makes use of predictive control to follow aggressive trajectories and a single tilt angle

for both quadrotor arms to reduce the weight of the mechanical system while avoiding the burden of the response time of the orientation loop to generate translational forces.

The method to track aggressive trajectories with a quadrotor is as follows. Section 4.2.1 shows the dynamic model of the tilting arms quadrotor. Section 4.2.2 presents the control laws applied to the mechanical system for autonomous trajectory tracking. Section 4.2.3 presents the prototype, the system architecture and the experimental setup.

4.2.1 Modeling

The goal of this section is to present the dynamical model of the quadrotor with tilting arms developed in this work. We build on the work in [16] and extend the dynamical model of the single axis tilting arms quadrotor to also present the state model with an inner loop for easier integration with a flight controller. This step is necessary to carry out model-based control and future testing. A methodology combining rigid-body dynamics and aerodynamic modeling will be used. The following hypothesis regarding the modeling will be considered for the purpose of simplifying the modeling:

- The structure of the drone is considered rigid and symmetric.
- There is no blade flapping and the rotors are considered rigid.
- The rotational speeds of the rotors are not affecting the dynamics of the tilting motors.
- The center of gravity coincides with the quadrotor body frame origin.
- The dynamics of all the actuators (rotational motors and tilting motors) are assumed to be faster than the dynamics of the vehicle and can therefore be neglected [2].
- The inertia of the propellers is considered negligible compared to the quadrotor and thus gyroscopic effects are considered as disturbances to be rejected by the controller.

The proposed model is composed of 4 rotors mounted on 4 arms that can rotate about their x_{P_i} axis with a tilting angle that is the same for all arms because they are actuated by the same servo-motor.

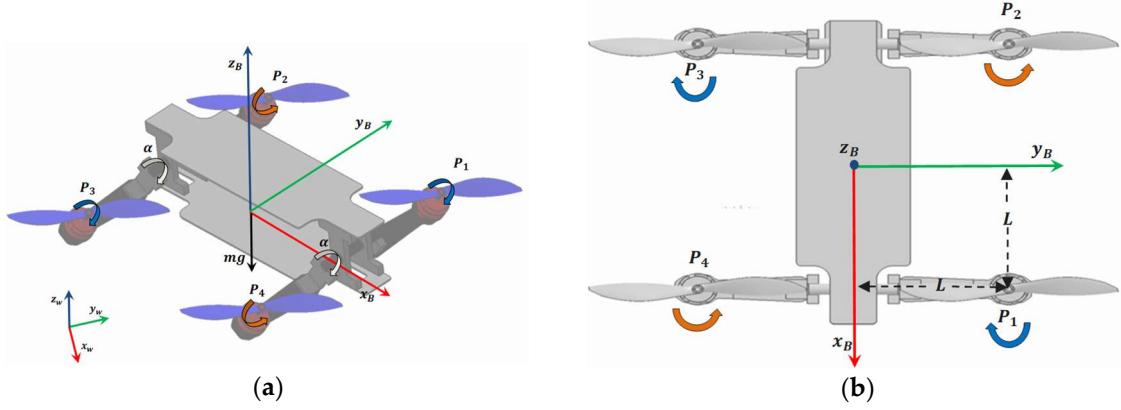


Fig. 4.1 : (a) Inertial and quadrotor body coordinate systems. (b) Upper view of the vehicle with the lever arms L

4.2.1.1 Coordinates and Conventions

The free body diagram and the coordinate systems are represented in **Error! Reference source not found..** The world frame W is defined by axis x_W , y_W and z_W with the latter pointing upward. The body frame B attached to the center of mass of the vehicle and moving with it is defined by the axis x_B pointing forward, y_B pointing to the left side of the vehicle and z_B pointing upward and perpendicular to the plane of the vehicle with the rotors directed vertically up when the arms are not tilted. Similarly, we also define the propellers coordinate system in Fig. 4.2. The propeller frame P_i is attached to each propeller at O_{P_i} and tilts around each arm axis. The axis defining this frame are x_{P_i} pointing forward, y_{P_i} pointing to the left side of the vehicle and corresponding to the tilting actuation axis and z_{P_i} perpendicular to the plane on which rests each rotor. The effective axis on which the thrust of the spinning propellers is applied is z_{P_i} . As the convention requires, we let ${}^A\mathbf{R}_B \in SO(3)$ be a rotation matrix that represents the orientation to get from coordinate system \mathcal{F}_B to coordinate system \mathcal{F}_A . Using X-Y-Z Euler angles, we can get from \mathcal{F}_W to \mathcal{F}_B using the following rotation matrix

$${}^W\mathbf{R}_B = \begin{bmatrix} c(\theta)c(\psi) & s(\varphi)s(\theta)c(\psi) - c(\varphi)s(\psi) & s(\psi)s(\varphi) + c(\varphi)s(\theta)c(\psi) \\ c(\theta)s(\psi) & s(\varphi)s(\theta)s + c(\varphi)c(\psi) & c(\varphi)s(\theta)s(\psi) - c(\psi)s(\varphi) \\ -s(\theta) & s(\varphi)c(\theta) & c(\theta)c(\varphi) \end{bmatrix}, \quad (4.1)$$

with $c(\cdot)$ and $s(\cdot)$ representing the cosine and sine functions respectively. The angles φ , θ and ψ are the roll, pitch and yaw Euler angles respectively in \mathcal{F}_W .

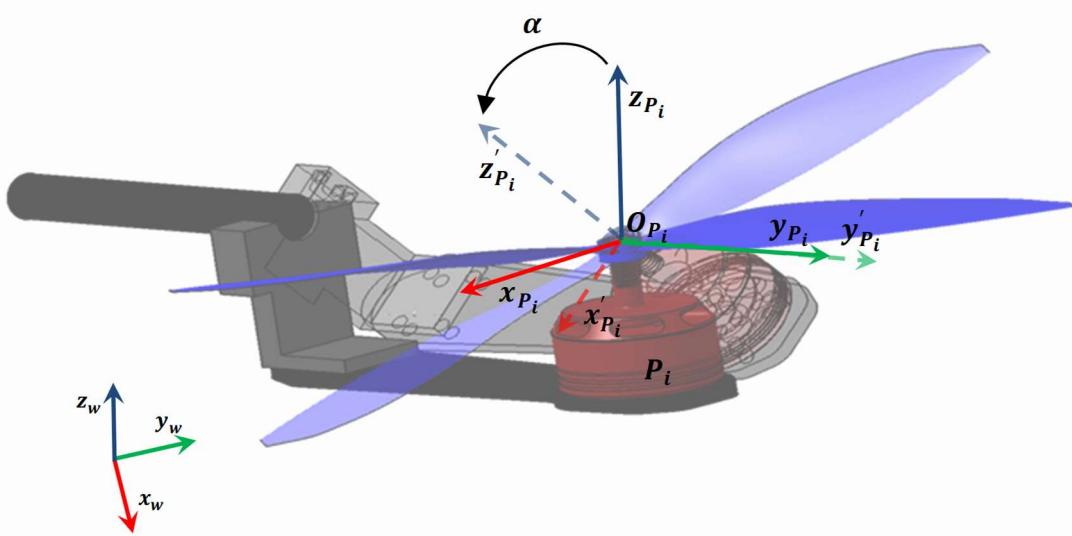


Fig. 4.2 : Rotor coordinate system with the direction of a positive defined tilting angle α

The orientation matrix to rotate the i^{th} rotor frame P_i to the vehicle body frame is defined by ${}^B\mathbf{R}_{P_i}$. By denoting $\alpha \in \mathbb{R}, \left[-\frac{\pi}{4}, \frac{\pi}{4}\right]$ the propeller tilt angle of all the rotors by the \mathbf{y}_{P_i} axis parallel to the \mathbf{y}_B axis it follows that

$${}^B\mathbf{R}_{P_i} = \begin{bmatrix} c(\alpha) & 0 & s(\alpha) \\ 0 & 1 & 0 \\ -s(\alpha) & 0 & c(\alpha) \end{bmatrix}, \quad (4.2)$$

which is an $\mathbf{R}_y(\theta)$ rotation matrix¹. For further use, we also define a vector that denotes the distance from the center of mass of the vehicle to each propeller's center

¹ All through the following, $\mathbf{R}_x(\theta)$, $\mathbf{R}_y(\theta)$ and $\mathbf{R}_z(\theta)$ will stand for the orientation matrixes of angle θ around the x , y , and z axes respectively.

$${}^B\mathbf{r}_{P_i} = \mathbf{R}_z \left((i-1) \frac{\pi}{2} \right) \begin{bmatrix} L \\ L \\ 0 \end{bmatrix}. \quad (4.3)$$

4.2.1.2 Dynamic Model

The system dynamics of the tilting arms quadrotor are obtained by exploiting standard Newton-Euler formalization. The vehicle is treated as a rigid body subject to forces generated from the rotating propellers and the gravity. As usual in quadrotor research, the propeller is simplified as a punctual non-rotational mass aggregated with the vehicle total mass that generates thrust proportional to the square of the propeller rotation speed and a counter torque due to air drag [8]. Other aerodynamic effects can be neglected. Therefore, the forces produced from the i^{th} spinning rotor are given by:

$$F_{T,i} = k_f \omega_i^2, \quad (4.4)$$

$$M_{D,i} = (-1)^i k_m F_{T,i}, \quad (4.5)$$

where k_f and k_m are positive defined constants that could be determined experimentally for each rotor and ω_i is the spinning velocity of the i^{th} rotor. Due to the tilting capability of the arms, these forces are produced in \mathbf{z}_{P_i} . Thus,

$$\mathbf{F}_{T_{P,i}} = [0 \ 0 \ k_f \omega_i^2]^T \quad (4.6)$$

$$\mathbf{M}_{D_{P,i}} = [0 \ 0 \ (-1)^i k_m k_f \omega_i^2]^T \quad (4.7)$$

represent the thrust force and moment of the i^{th} rotor in $\mathcal{F}_{P,i}$, respectively.

By applying the Euler equations of motion, one can find that the rotational dynamics of the vehicle are

$$\mathbf{J}_B \dot{\boldsymbol{\omega}}_B + \boldsymbol{\omega}_B \times \mathbf{J}_B \boldsymbol{\omega}_B = \sum_{i=1}^4 \left({}^B \mathbf{r}_{P_i} \times {}^B \mathbf{R}_{P_i} \mathbf{F}_{T_{P,i}} - {}^B \mathbf{R}_{P_i} \mathbf{M}_{D_{P,i}} \right). \quad (4.8)$$

where $\mathbf{J}_B \in \mathbb{R}^{3 \times 3}$ is the inertia matrix (constant, symmetric and positive defined), $\boldsymbol{\omega}_B \in \mathbb{R}^3$ is the angular velocity and $\dot{\boldsymbol{\omega}}_B \in \mathbb{R}^3$ is the angular acceleration of the vehicle in the body frame. The right hand-side of equation (4.8) gives the torques applied on the vehicle through the spinning velocities of the propellers as control inputs. Thus, we get

$$\mathbf{M}_B = \mathbf{J}_B \dot{\boldsymbol{\omega}}_B + \boldsymbol{\omega}_B \times \mathbf{J}_B \boldsymbol{\omega}_B = \mathcal{A}_M(\alpha) \begin{bmatrix} \omega_1^2 \\ \omega_2^2 \\ \omega_3^2 \\ \omega_4^2 \end{bmatrix}, \quad (4.9)$$

where $\mathcal{A}_M(\alpha) \in \mathbb{R}^{3 \times 4}$ is the allocation matrix for the torque applied on the quadrotor and defined as

$$\mathcal{A}_M(\alpha) = \begin{bmatrix} Lk_f c(\alpha) - k_m k_f s(\alpha) & Lk_f c(\alpha) + k_m k_f s(\alpha) & -Lk_f c(\alpha) - k_m k_f s(\alpha) & -Lk_f c(\alpha) + k_m k_f s(\alpha) \\ -Lk_f c(\alpha) & Lk_f c(\alpha) & Lk_f c(\alpha) & -Lk_f c(\alpha) \\ -Lk_f s(\alpha) - k_m k_f c(\alpha) & -Lk_f s(\alpha) + k_m k_f c(\alpha) & Lk_f s(\alpha) - k_m k_f c(\alpha) & Lk_f s(\alpha) + k_m k_f c(\alpha) \end{bmatrix}. \quad (4.10)$$

The translational dynamics can be expressed as

$$\mathbf{F}_B = m \dot{\mathbf{v}}_B + m {}^B \mathbf{R}_W \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} = \sum_{i=1}^4 {}^B \mathbf{R}_{P_i} \mathbf{F}_{T_{P,i}}, \quad (4.11)$$

with m as the total mass of the vehicle including the propellers, $\mathbf{v}_B \in \mathbb{R}^3$ the speed of the vehicle in the body frame and g the standard earth's gravitational acceleration. As for the rotational dynamics, equation (4.11) can be rewritten with an input coupling matrix

$$m \dot{\mathbf{v}}_B + m {}^B \mathbf{R}_W \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} = \mathcal{A}_F(\alpha) \begin{bmatrix} \omega_1^2 \\ \omega_2^2 \\ \omega_3^2 \\ \omega_4^2 \end{bmatrix}, \quad (4.12)$$

where $\mathcal{A}_F(\alpha) \in \mathbb{R}^{3 \times 4}$ is the allocation matrix for the translational forces applied on the quadrotor and defined as

$$\mathcal{A}_F(\alpha) = \begin{bmatrix} k_f s(\alpha) & k_f s(\alpha) & k_f s(\alpha) & k_f s(\alpha) \\ 0 & 0 & 0 & 0 \\ k_f c(\alpha) & k_f c(\alpha) & k_f c(\alpha) & k_f c(\alpha) \end{bmatrix}. \quad (4.13)$$

Combining the rotational and translational dynamics, one can find that the full system can be described as

$$\begin{pmatrix} \mathbf{M}_B \\ \mathbf{F}_B \end{pmatrix} = \mathcal{A}(\alpha) \begin{bmatrix} \omega_1^2 \\ \omega_2^2 \\ \omega_3^2 \\ \omega_4^2 \end{bmatrix}. \quad (4.14)$$

with $\mathcal{A}(\alpha) = [\mathcal{A}_M(\alpha) \quad \mathcal{A}_F(\alpha)]^T \in \mathbb{R}^{6 \times 4}$ the full system allocation matrix.

4.2.1.3 Simplified Model with Inner Loop

The focus of this section is about finding a dynamic system model more suitable for control. Solving equation (4.14) in order to control the moments and forces applied to the vehicle is a non-trivial task since $\mathcal{A}(\alpha)$ is not full rank and a nonlinear function of the tilting angles α . The allocation matrix is also non-invertible with a full row of zeros because the force in the \mathbf{y}_B direction can't be imposed directly. Therefore, in this section we introduce an inner attitude loop as usually done with quadrotors to obtain control authority over $F_{\mathbf{y}_B}$. Despite the fact that this inner loop is not required for the \mathbf{x}_B direction control, it's convenient to use it also to control not only φ but θ to facilitate integration with a commercial flight controller.

First, we need to transform the nonlinear allocation problem into a linear problem using variable transformation. Therefore, we define the following virtual control inputs

$$U_1 = Lk_f(\omega_1^2 + \omega_2^2 - \omega_3^2 - \omega_4^2) \cos \alpha + k_m k_f \sin \alpha \sum_{i=1}^4 (-1)^i \omega_i^2, \quad (4.15)$$

$$U_2 = Lk_f(-\omega_1^2 + \omega_2^2 + \omega_3^2 - \omega_4^2) \cos \alpha,$$

$$U_3 = Lk_f(-\omega_1^2 - \omega_2^2 + \omega_3^2 + \omega_4^2) \sin \alpha + k_m k_f \cos \alpha \sum_{i=1}^4 (-1)^i \omega_i^2,$$

$$U_4 = k_f \cos \alpha \sum_{i=1}^4 \omega_i^2,$$

with U_1 , U_2 and U_3 representing the total moments around \mathbf{x}_B , \mathbf{y}_B and \mathbf{z}_B respectively and U_4 the upward force in \mathbf{z}_B direction. The impact of these control inputs on the vehicle dynamics are similar as the one chosen for a quadrotor with no tilting arms with the difference that they are a function of not only ω_i but also α , see [8]. Thus, (4.14) becomes

$$\begin{pmatrix} \mathbf{M}_B \\ \mathbf{F}_B \end{pmatrix} = \begin{bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \tan \alpha \\ 0 \\ U_4 \end{bmatrix} \quad (4.16)$$

This formulation can be used for control assuming the control inputs have fast transients, which is the case because they depend only on ω_i and α which are faster than the quadrotor dynamics.

Next, we simplify the model with an inner loop losing direct control over attitude but gaining control over the vehicle positions. The same cascaded approach as in [13,14] is used by assuming that a high-rate commercial controller tracks the attitude of the vehicle. Therefore, desired angles can be given as control inputs in our model. The inner loop attitude dynamics have to be considered in the high level position commands for accurate tracking because in contrary to the torque control input, attitude control inputs are not tracked instantaneously. Thus, the closed-loop response controller dynamic equations can either be found and integrated in the quadrotor model or identified by a system identification procedure that is more convenient for an unknown commercial flight controller attitude loop. We chose to consider the inner loop as a first order closed-loop as suggested in [13]

$$\dot{\varphi} = \frac{1}{\tau_\varphi} (k_\varphi \varphi_{cmd} - \varphi), \quad (4.17)$$

$$\dot{\theta} = \frac{1}{\tau_\theta} (k_\theta \theta_{cmd} - \theta),$$

$$\dot{\psi} = \dot{\psi}_{cmd},$$

where τ_φ , τ_θ and k_φ , k_θ are the time constants and gains of the roll and pitch respectively to be determined experimentally for a tuned commercial flight controller. φ_{cmd} , θ_{cmd} and $\dot{\psi}_{cmd}$ are the new control inputs, the commanded roll and pitch angles and the vehicle heading angular velocity respectively. The heading rate is assumed to be tracked instantaneously which is a reasonable assumption if we are only interested in position control and not in attitude control. Indeed, the heading has no effect on the vehicle's position, i.e. the vehicle can follow a trajectory with any heading so keeping the same heading is not a requirement.

The translational and rotational dynamics of the quadrotor with tilting arms can then be expressed in a state-space model as

$$\left. \begin{array}{l} \dot{x}_1 = \frac{1}{\tau_\varphi} (k_\varphi \varphi_{cmd} - x_1) \\ \dot{x}_2 = \frac{1}{\tau_\theta} (k_\theta \theta_{cmd} - x_2) \\ \dot{x}_3 = \dot{\psi}_{cmd} \\ \dot{x}_4 = x_5 \\ \dot{x}_5 = \frac{T_{cmd}}{m} (s(x_3)s(x_1) + c(x_1)s(x_2)c(x_3) + c(x_2)c(x_3)t(\alpha_{cmd})) \\ \dot{x}_6 = x_7 \\ \dot{x}_7 = \frac{T_{cmd}}{m} (c(x_1)s(x_2)s(x_3) - s(x_1)c(x_3) + c(x_2)s(x_3)t(\alpha_{cmd})) \\ \dot{x}_8 = x_9 \\ \dot{x}_9 = -g + \frac{T_{cmd}}{m} (c(x_3)c(x_1) - s(x_3)t(\alpha_{cmd})) \end{array} \right\}, \quad (4.18)$$

with the following defined state vector

$$x = (\varphi \ \theta \ \psi \ \mathbf{p}^T \ \mathbf{v}^T)^T, \quad (4.19)$$

and control input vector

$$\mathbf{u} = (T_{cmd} \varphi_{cmd} \theta_{cmd} \dot{\psi}_{cmd} \alpha_{cmd})^T, \quad (4.20)$$

where $\mathbf{p} \in \mathbb{R}^3$ and $\mathbf{v} \in \mathbb{R}^3$ are the vehicle Cartesian positions and velocities respectively in \mathcal{F}_W . T_{cmd} is the total thrust applied by the spinning rotors in \mathbf{z}_P and α_{cmd} the commanded tilting angle. Note that $t(\cdot)$ is the tangent function.

4.2.2 Control

The control strategies adopted in this paper to track aggressive trajectories in position with the tilting arms quadrotor are presented in the following section. We start with the general control structure applied. Since a significant contribution of the paper is the comparison between LQR and iLQR for aggressive trajectory tracking, we present both controllers. Then, the trajectory planning is explained.

4.2.2.1 General Control Structure

A cascaded approach is used for the controllers presented in this paper. An inner loop is responsible of tracking the desired orientation of the vehicle while an outer loop tracks the Cartesian positions and velocities of the vehicle, see Fig. 4.3.

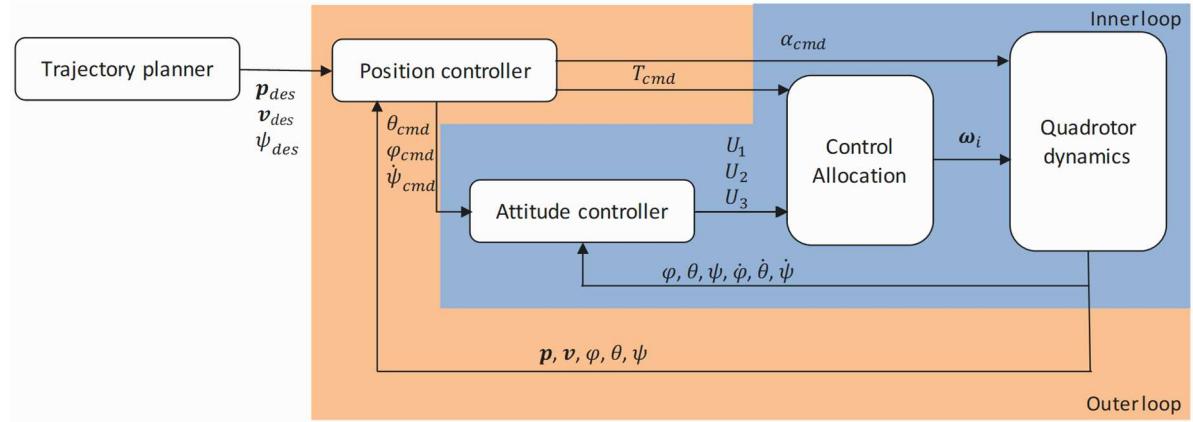


Fig. 4.3 : Control structure overview

Essentially, the LQR and iLQR controllers are responsible for the high-level control while a proportional-derivative (PD) controller is used for the inner loop as usually done [18] or this task

is left to the attitude control loop of the onboard flight controller. Therefore, the inner loop control used in simulation is

$$\begin{aligned} U_1 &= k_{p,\varphi}(\varphi_{cmd} - \varphi) + k_{d,\varphi}(\dot{\varphi}_{des} - \dot{\varphi}), \\ U_2 &= k_{p,\theta}(\theta_{cmd} - \theta) + k_{d,\theta}(\dot{\theta}_{des} - \dot{\theta}), \\ U_3 &= k_{d,\psi}(\dot{\psi}_{cmd} - \dot{\psi}), \end{aligned} \quad (4.21)$$

with $k_{p,\varphi}, k_{d,\varphi}, k_{p,\theta}, k_{d,\theta}, k_{d,\psi}$ gains to be tuned. Note that $\dot{\varphi}_{des}$ and $\dot{\theta}_{des}$ are desired angular rates that can be set to zero or computed from the reference trajectory as explained in [16, 19]. We chose to set them to zero for simplicity.

4.2.2.2 Linear Quadratic Regulator

A Linear Quadratic Regulator (LQR) returns the optimal solution for a linear time-invariant system of the form

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}, \quad \mathbf{y} = \mathbf{C}\mathbf{x}, \quad (4.22)$$

at the reference state $\mathbf{x}_{ref} = [0 \ 0 \ \psi_{des} \ \mathbf{p}_{des} \ \mathbf{v}_{des}]$, which is the desired state given by the trajectory planner at each time and the reference input $\mathbf{u}_{ref} = [mg \ 0 \ 0 \ 0 \ 0]$ equivalent to the control inputs at hover. The performance index for the infinite-time solution to be minimised is

$$J(\mathbf{x}, \mathbf{u}) = \int_0^{\infty} (\Delta\mathbf{x}^T \mathbf{Q} \Delta\mathbf{x} + \Delta\mathbf{u}^T \mathbf{R} \Delta\mathbf{u}) dt, \quad (4.23)$$

with the errors $\Delta\mathbf{x} = \mathbf{x} - \mathbf{x}_{ref}$ and $\Delta\mathbf{u} = \mathbf{u} - \mathbf{u}_{ref}$ and $\mathbf{Q} \succcurlyeq \mathbf{0}$ and $\mathbf{R} \succ \mathbf{0}$ cost matrices associated with state errors and magnitude of the control effort. One approach to find the optimal solution is to find $\mathbf{P}(t)$ of the continuous algebraic Riccati equation

$$\dot{\mathbf{P}} = -\mathbf{P}\mathbf{A} - \mathbf{A}^T\mathbf{P} - \mathbf{Q} + \mathbf{P}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}, \quad (4.24)$$

Then the optimal control input to apply is given by

$$\mathbf{u} = \mathbf{u}_{ref} + K\Delta\mathbf{x} \quad \text{with } \mathbf{K} = \mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}. \quad (4.25)$$

To adjust the LQR gains to the state of the vehicle, we introduce a state-dependent linearization that solves the optimal control problem with an updated model state space. Therefore, the state space jacobians

$$\mathbf{A} = \frac{\partial \mathbf{f}}{\partial \mathbf{x}}_{\mathbf{x}_{ref}, \mathbf{u}_{ref}}, \mathbf{B} = \frac{\partial \mathbf{f}}{\partial \mathbf{u}}_{\mathbf{x}_{ref}, \mathbf{u}_{ref}}, \quad (4.26)$$

where $\mathbf{f}(\mathbf{x}, \mathbf{u})$ is the state space model, are continuously updated during the trajectory by the high-level controller (see Appendix A). The Schur method is used to solve the Riccati equation [41].

4.2.2.3 Iterative Linear Quadratic Regulator

Since an LQR optimal controller disregards state changes in the future, it is unsuitable for trajectory tracking and in particular aggressive trajectory tracking where anticipation is non-negotiable. Therefore, we apply an iterative LQR optimal controller to the quadrotor with tilting arms to optimize the whole control sequence on a trajectory while considering the changing dynamics of the vehicle concurrently. This algorithm uses iterative linearization of the non-linear dynamics around a nominal trajectory at each step of the iteration, and minimizes a performance index to find an optimal control input. In short, it consists in a Gauss-Newton approximation to Differential Dynamic Programming (DDP) as explained in [23, 24] where the hessians of \mathbf{f} are neglected. Because of this approximation, DDP should give more optimal results with a higher computational cost than iLQR.

In the laboratory, the iLQR was already applied to the whole-body grasping movement on a 25-DOF humanoid robot [25]. Especially in this paper, we described a square-root formulation of the DDP algorithm, by taking advantage of the square shape of the cost and derivatives to propose a more efficient formulation of this algorithm. This shape was also used to make some correlations with the classical inverse kinematics.

This method was also compared to the DDP and LQR methods with the same robot [26]. Especially in this paper, the method with DDP can be seen as a generalization to a large class of actuators and to non-linear situations of the LQR.

In [27] for the torque control of an arm, the muscle control of an arm and the inverted pendulum problems, iLQR converged in less iterations and with lower costs to the optimal solutions than DDP. Therefore, neglecting the hessians in iLQR does not reduce the optimality of the solution. Since iLQR is a Gauss-Newton approximation it converges to at least a local minimum [27].

We start with a system with discrete-time dynamics modeled by the function \mathbf{f} that describes the evaluation of the system from time t to $t + 1$

$$\mathbf{x}_{t+1} = \mathbf{f}(\mathbf{x}_t, \mathbf{u}_t), \quad (4.27)$$

corresponding to (4.18) for the system studied in this paper. The total cost which is a sum of running costs l and final cost l_f over the horizon N is

$$J(\mathbf{x}_0, \mathbf{U}) = \sum_{t=0}^{N-1} l(\mathbf{x}_t, \mathbf{u}_t) + l_f(\mathbf{x}_N), \quad (4.28)$$

with the trajectory $\{\mathbf{X}, \mathbf{U}\}$ as a sequence of states $\mathbf{X} = \{\mathbf{x}_0, \mathbf{x}_1, \dots, \mathbf{x}_N\}$ that results from applying the control sequence $\mathbf{U} = \{\mathbf{u}_0, \mathbf{u}_1, \dots, \mathbf{u}_{N-1}\}$ satisfying (4.27). The states \mathbf{X} are derived by integrating the systems dynamics from (4.27) from the initial state and an initial control sequence set to

$$\mathbf{U}_0 = \left\{ \begin{bmatrix} mg \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}_1, \begin{bmatrix} mg \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}_2, \dots, \begin{bmatrix} mg \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}_{N-1} \right\}, \quad (4.29)$$

who is basically a stationary hover of the vehicle during all the trajectory. Next, we define a *cost-to-go* as a sum of costs from time t to N

$$J_t(\mathbf{x}_0, \mathbf{U}_t) = \sum_{k=t}^{N-1} l(\mathbf{x}_k, \mathbf{u}_k) + l_f(\mathbf{x}_N), \quad (4.30)$$

and a value function as the optimal *cost-to-go* from a given state at time t

$$V_t(\mathbf{x}) = \min_{\mathbf{U}_t} J_t(\mathbf{x}, \mathbf{U}_t). \quad (4.31)$$

At final time step N , this value is $V_N(\mathbf{x}_N) = l_f(\mathbf{x}_N)$. Then the minimization over a sequence of controls \mathbf{U}_t reduces to a sequence of minimizations over a single control \mathbf{u} by finding the value function at time t from the value function at time $t + 1$ and the immediate cost

$$V(\mathbf{x}) = \min_u [l(\mathbf{x}, \mathbf{u}) + V'], \quad (4.32)$$

where $V' = V(\mathbf{f}(\mathbf{x}, \mathbf{u}))$ is the value function at $t + 1$.

After the initial guess \mathbf{U}_0 , we want to improve our control sequence and converge to a local minimum. To do so, we add small perturbation values $(\delta\mathbf{x}, \delta\mathbf{u})$ to $(\mathbf{x}_t, \mathbf{u}_t)$ and use them to minimize the value function. Let

$$Q(\delta\mathbf{x}, \delta\mathbf{u}) = l(\mathbf{x} + \delta\mathbf{x}, \mathbf{u} + \delta\mathbf{u}) + V(\mathbf{f}(\mathbf{x} + \delta\mathbf{x}, \mathbf{u} + \delta\mathbf{u})), \quad (4.33)$$

be the change in the value function, which is also known as the *pseudo-Hamiltonian*. Using a quadratic Taylor expansion of Q

$$\begin{aligned} Q_{\mathbf{x}} &= l_{\mathbf{x}} + \mathbf{f}_{\mathbf{x}}^T V'_{\mathbf{x}}, \\ Q_{\mathbf{u}} &= l_{\mathbf{u}} + \mathbf{f}_{\mathbf{u}}^T V'_{\mathbf{x}}, \\ Q_{\mathbf{xx}} &= l_{\mathbf{xx}} + \mathbf{f}_{\mathbf{x}}^T V'_{\mathbf{xx}} \mathbf{f}_{\mathbf{x}} + V'_{\mathbf{x}} \cdot \mathbf{f}_{\mathbf{xx}}, \\ Q_{\mathbf{ux}} &= l_{\mathbf{ux}} + \mathbf{f}_{\mathbf{u}}^T V'_{\mathbf{xx}} \mathbf{f}_{\mathbf{x}} + V'_{\mathbf{x}} \cdot \mathbf{f}_{\mathbf{ux}}, \\ Q_{\mathbf{uu}} &= l_{\mathbf{uu}} + \mathbf{f}_{\mathbf{u}}^T V'_{\mathbf{xx}} \mathbf{f}_{\mathbf{u}} + V'_{\mathbf{x}} \cdot \mathbf{f}_{\mathbf{uu}}, \end{aligned} \quad (4.34)$$

we can compute a local solution to (4.32). Note that $V'_j \cdot \mathbf{f}_{jj}$ is the product of a vector but because we neglect the Hessians of \mathbf{f} in contrary to DDP algorithm, this product is omitted. The optimal control modification $\delta\mathbf{u}^*$ can then be computed

$$\delta\mathbf{u}^*(\delta\mathbf{x}) = \underset{\delta\mathbf{u}}{\operatorname{argmin}} Q(\delta\mathbf{x}, \delta\mathbf{u}) = \mathbf{k} + \mathbf{K}\delta\mathbf{x}, \quad (4.35)$$

with locally-linear feedback policy $\mathbf{k} = -Q_{uu}^{-1}Q_u$ and $\mathbf{K} = -Q_{uu}^{-1}Q_{ux}$. With this policy and the quadratic expansion of Q , a quadratic model of V is found

$$\begin{aligned} V_x &= Q_x - \mathbf{K}^T Q_{uu} \mathbf{k}, \\ V_{xx} &= Q_{xx} - \mathbf{K}^T Q_{uu} \mathbf{K}, \end{aligned} \tag{4.36}$$

Starting from V_N , one can now recursively compute the value function and evaluate the new trajectory with the optimal control sequence as

$$\begin{aligned} \hat{\mathbf{x}}_0 &= \mathbf{x}_0, \\ \hat{\mathbf{u}}_t &= \mathbf{u}_t + \mathbf{k}_t + \mathbf{K}_t(\hat{\mathbf{x}}_t - \mathbf{x}_t), \\ \hat{\mathbf{x}}_{t+1} &= \mathbf{f}(\hat{\mathbf{x}}_t, \hat{\mathbf{u}}_t), \end{aligned} \tag{4.37}$$

where $\hat{\cdot}$ denotes newly found control and state sequences. This process is then repeated until convergence.

The iLQR algorithm optimizes the control output over the horizon N while LQR optimizes it at t . Therefore, the optimal solution between the two algorithms is comparable only on the time horizon with the iLQR having the advantage of taking into consideration N successive controls and states.

This control algorithm could be implemented to control the quadrotor in real-time because of a limited horizon of 50 steps and the simple model dynamics of the robot (18).

4.2.2.4 Trajectory Generation

The same approach for trajectory planning as in [16] is used to generate a reference trajectory for the controller. We decompose a trajectory into segments and find an optimal path that transitions smoothly through keyframes at specific transition times. Specifically, we consider trajectories in the form of piecewise polynomial functions of order n over m time intervals

$$\sigma_T(t) = \begin{cases} \sum_{i=0}^n \sigma_{Ti1} t^i & t_0 \leq t < t_1 \\ \sum_{i=0}^n \sigma_{Ti2} t^i & t_1 \leq t < t_2 \\ \vdots \\ \sum_{i=0}^n \sigma_{Tim} t^i & t_{m-1} \leq t < t_m \end{cases}, \quad (4.38)$$

where σ_T is the trajectory of a flat output. That is one of the states that enables to write the quadrotor dynamics as algebraic functions of the selected flat outputs and their derivatives, which are x, y, z and ψ for the quadrotor [16]. We can then minimize the integral of the k th derivative of the norm of the flat output squared to find an optimal trajectory that passes through each keyframe at time t_i

$$\begin{aligned} & \min \int_{t_0}^{t_m} \mu \left\| \frac{d^k \sigma_T}{dt^k} \right\|^2 dt, \\ & \text{s.t.} \quad \sigma_T(t_i) = \sigma_i, \quad i = 0, \dots, m \\ & \quad \left. \frac{d^p \sigma_T}{dt^p} \right|_{t=t_j} = 0 \text{ or free}, \quad j = 0, m; p = 1, \dots, k \end{aligned} \quad (4.39)$$

where μ is a constant that makes the integrand dimensionless. We choose to optimize over the snap (4-th derivative) as usually done to obtain smooth motion for quadrotors.

Therefore, solving (4.29) can provide reference states up to the 4-th derivative for the flat outputs of the quadrotor. Overall, the inputs of the algorithm are the keyframe times and constraints on the flat outputs up to the 4-th derivative and the outputs are the polynomials of each flat output that follow the desired trajectory defined by keyframes and their constraints. From polynomials, the required flat outputs and derivatives are deduced on a discretization time of choice.

4.2.3 Experimental Setup

4.2.3.1 Prototype

A custom-made quadrotor was developed as a prototype since it is a novel titling arms design. The electronics composing the prototype are easily available and massively used in the world of UAVs. The three main frames on which rest the electronic parts and the arms are made of carbon fiber. It is lightweight, resistant and a common adopted material for racing quadrotors. As actuators, we choose 2000 kV brushless motors (Cobra CM-2208/20) with Gemfan 6045 carbon nylon propellers which provide 28N of thrust at full speed with at lithium 3 cells (11.1V, 1300mAh, 75C) battery. This propulsion system is ideal for aggressive motion of an average weight (1kg) quadrotor which is the weight targeted. Each arm is linked to a timing pulley driven by a timing belt. The timing belt is set in motion by a servo motor (Corally CS-5016) that is responsible of the tilting capability of both arms. Using a single, powerful servo-motor saves weight and assures that both arms are tilted to the same angle at the same time. It has a maximum torque of 1.62 Nm and a maximum rotational speed of 20.9 rad/s. The maximum range is 90 deg, therefore the tilting arms travel range is ± 45 deg. As it can be observed in Fig. 4.4, the barycenter of the propeller is aligned with the center of mass of the vehicle as it was modeled in Sect. 4.2.1. The motors have to be set lower than the axles of rotation so the propellers keep the same position as the arms tilt about the axles. The rationale behind this is to ensure that the position of the center of mass of the vehicle is the same as the arms tilts to avoid parametric uncertainties.

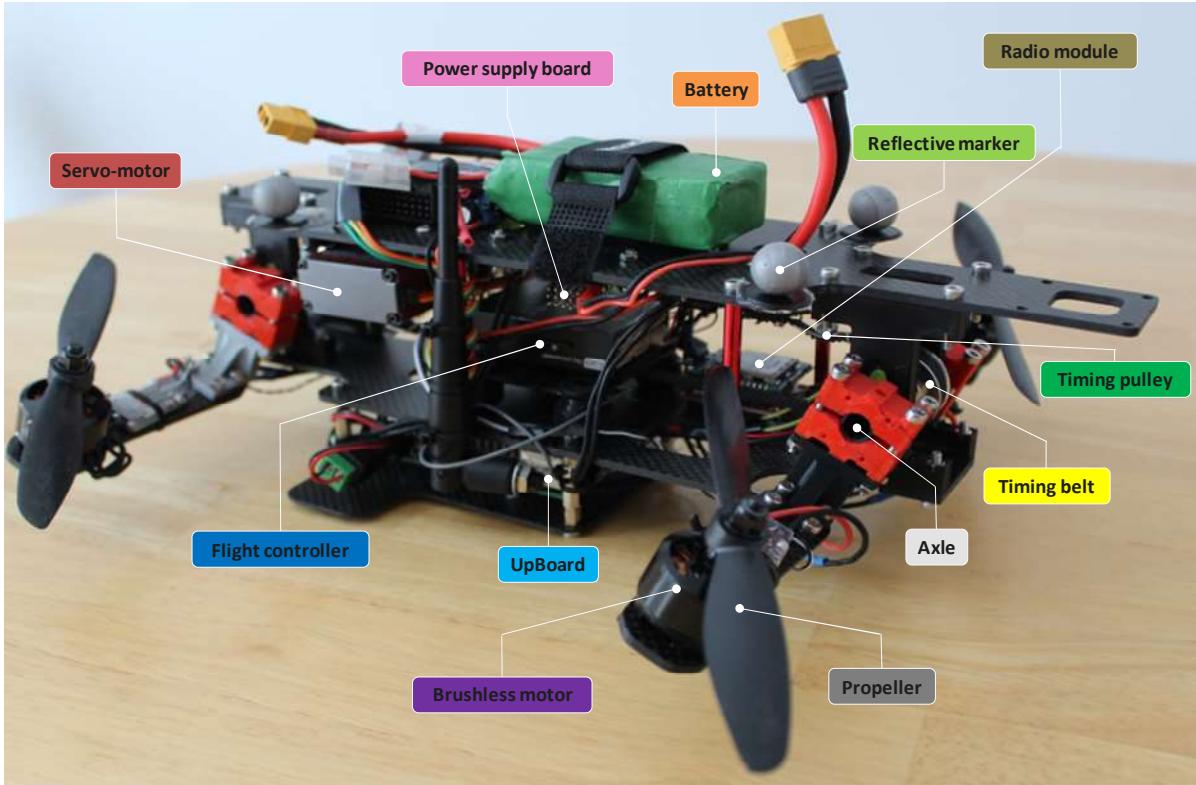


Fig. 4.4 : Quadrotor experimental platform with arms tilted forwards and his various components

The high-level controller runs on an UpBoard [24] computer which is an Intel Atom board who has 1.92GHz processors with 4Gb of RAM. The onboard computer is equipped with Wi-Fi for low frequency communications such as trajectory generation commands. For higher communication rates, a Laird RM024 radio module is used which is connected via UART to the UpBoard. The computer is powered by the same battery as the motors, but since 5V is required a power-supply board containing a 12V DC/DC power converter for the necessary voltage conversion and stabilization is also mounted on the quadrotor.

Furthermore, for the low-level controller, we choose a PX4 mini equipped with a STM32F765 processor and two inertial measurement units (ICM-20689 and BMI055). The firmware is documented and it is widely used for custom-made UAVs. To take into account the dynamics of the low-level controller, system identification is performed to identify the required constants needed in the high-level control. After tuning the flight controller gains for aggressive motion, roll and pitch commands were send manually and the systems response was recorded to fit it to a first-

order closed-loop response. The prototype parameters are shown in Table 4.1, with the nominal mass weighted and the inertias obtained from a high detail CAD model.

Table 4.1 : Parameters of the quadrotor experimental platform

Parameter	Value
mass	1.14 kg
L	0.11 m
J_{xx}	0.004 kgm ²
J_{yy}	0.008 kgm ²
J_{zz}	0.01 kgm ²
τ_ϕ	0.22 sec.
τ_θ	0.17 sec.
k_ϕ	1.15
k_θ	1.10
$\alpha_{cmd,max}$	45 deg
$\alpha_{cmd,min}$	-45 deg
$T_{cmd,max}$	28 N

4.2.3.2 System Architecture

The main software is split amongst a fast low-level attitude controller and our superimposed high-level position controller as explained previously (see Sec. 4.2.2.1). The system architecture is illustrated in Fig. 4.5. On the PX4 flight controller, the attitude controller runs at 400 Hz, reading his own IMU to adjust the vehicles attitude in response to the high-level controller commands. He achieves this by computing PWM signals which can spin the propellers to required velocities ω . IMU readings from the PX4 at 100Hz are then used into a multi-sensor fusion framework (MSF) [25] with vehicle attitude angles provided by a lower rate sensor, a motion capture system [26], received via the radio module connected to the onboard computer. Knowing p , the linear velocity v is obtained by numerical differentiation and filtered. Due to the limited frequency of the motion

capture system (45 Hz) and the radio module (60 Hz), p and v are updated at 45 Hz but the high-level controller still runs at 100 Hz given updates of the orientation of the vehicle from the IMU. Tilting angle commands calculated by the high-level controller are sent to the PX4 that adjusts the PWM to control the servo-motor at 50 Hz. The position controller and the MSF run on the UpBoard using Ubuntu Linux [27] with ROS [28]. Trajectories to follow are sent to the onboard computer through Wi-Fi by specifying the high-level constraints. Then, full trajectories references are computed by the trajectory generation algorithm running on the UpBoard.

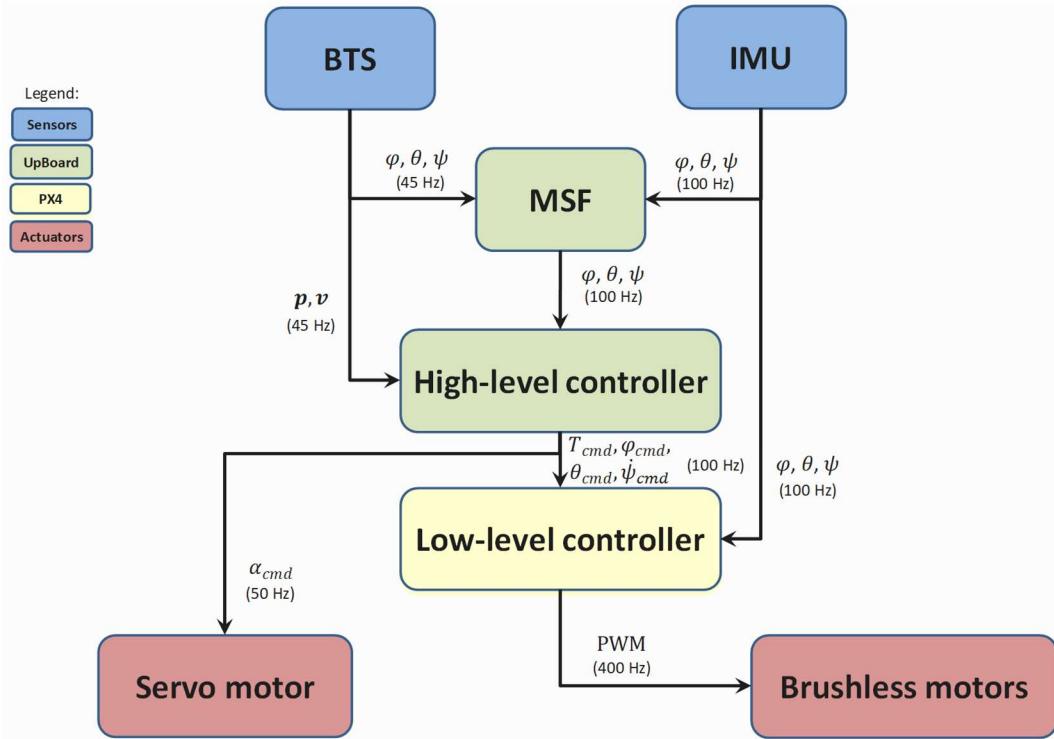


Fig. 4.5 : Overview of the control architecture

4.2.4 Analysis

On the one hand, the proposed iLQR controller (see Section 4.2.2.3) is compared to a non-predictive control approach such as a state-dependent LQR see Section 4.2.2.2). On the other hand, the proposed tilting arms quadrotor design is compared to a conventional quadrotor. The combination of tilting arms and a state-dependant iLQR is also compared to a state-dependant LQR controller without tilting arms.

These experimentations are conducted in simulations except for the comparison between adding tilting arms or not that is also made with the prototype presented in Section 4.2.3. The quadrotor is tested on two aggressive trajectories with one of them also reproduced on the real prototype in order to compare experimental to simulation performances.

4.3 Results

The aim of this section is to present the aggressive trajectory tracking capabilities of the tilting arms quadrotor and his controllers both in simulation and on the real prototype.

4.3.1 Simulations Results

We start by reporting on some idealistic simulations where the assumptions made in Sec. 4.2.1 are maintained. These results are meant to illustrate the trajectory tracking capability of the vehicle and his controller in ideal conditions so we are not considering control frequencies, actuation delays, mechanical vibrations and sensors noises. However, we want to demonstrate the robustness of the controller against non-linearities in the rotational dynamics of the vehicle who are neglected at the control design stage since an inner loop controller is used (see (4.16)) but included in the quadrotor dynamics (see (4.8)). For the simulations, the high-level and low-level controllers both run at 100 Hz with instantaneous actuation capabilities for α_{cmd} and ω . Therefore, the desired trajectory is sampled and provided at 100 Hz and the prediction horizon used for the iLQR is $N = 50$ corresponding to 0.5s at this control rate.

We consider an aggressive slalom path where the quadrotor has to cross two poles situated at 1.2 m of distance. This trajectory is challenging because the quadrotor has to travel about 5 m with two stops in less than 3 seconds to successfully track the desired path with velocities and accelerations above 3m/s and 10m/s² respectively. The shape of the trajectory can be seen in Fig. 4.6 and the tracking performance in subsequent Fig. 4.7, Fig. 4.8, Fig. 4.9. We present the performance of the LQR and iterative LQR controllers both considering a quadrotor without and with tilting arms. The results are reported in

Table 4.2.

The gains were manually tuned to:

$$\mathbf{Q} = \text{diag}([100, 100, 100, 1, 1, 1, 1, 10, 1]) \text{ and } \mathbf{R} = \text{diag}([0.1, 5, 10, 10]),$$

$$\mathbf{Q} = \text{diag}([1, 1, 1, 100, 10, 100000, 1, 100000, 10]) \text{ and } \mathbf{R} = \text{diag}([0.1, 5, 10, 10]),$$

$$\mathbf{Q} = \text{diag}([1, 1, 1, 100, 1, 1, 1, 100, 10]) \text{ and } \mathbf{R} = \text{diag}([0.1, 5, 10, 10, 1]),$$

$$\mathbf{Q} = \text{diag}([1, 1, 1, 100, 10, 100000, 1, 100000, 10]) \text{ and } \mathbf{R} = \text{diag}([0.1, 5, 10, 10, 1]),$$

in simulation, for the LQR controller with the conventional design, iLQR controller with conventional design, LQR controller with tilting arms and iLQR controller with tilting arms experiments respectively. Note that \mathbf{R} is of length 4 in the conventional design experiments since there is no control input α_{cmd} . The inner loop controller gains from (4.21) are $k_{p,\varphi} = 4.5$, $k_{d,\varphi} = 0.78$, $k_{p,\theta} = 2.5$, $k_{d,\theta} = 0.78$, $k_{d,\psi} = 0.25$. These gains are also tuned manually and the same are used in Section 4.2.3. To tune the inner loop controller, angular setpoints were used to find gains that avoid overshoots and with a response time of at least 0.3 second. The position controller gains are tuned by allowing more or less importance to some states and controls. For example, to use more T_{cmd} than α_{cmd} the gain for T_{cmd} has to be lower to put a lower penalty on a high T_{cmd} . Likewise, the states are tuned by according more or less importance to some specific states and asserting the RMSEs for each state. To avoid losing control of the quadrotor with high roll and pitch setpoints, the angular states gains should be set higher so deviation from the desired orientation of zero are more penalizing in the computation of the optimal controls.

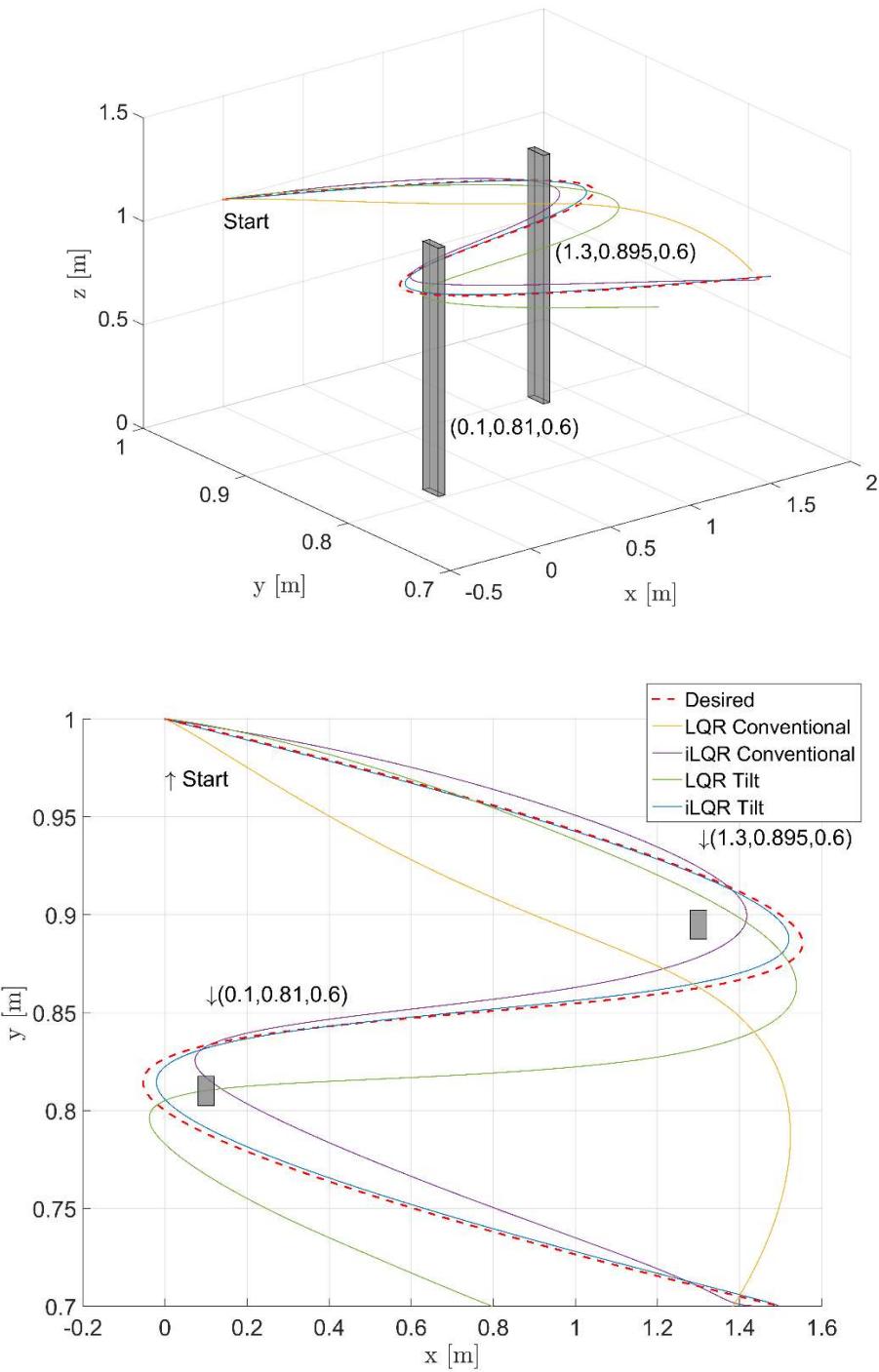


Fig. 4.6. : Desired (dashed red), LQR conventional (solid yellow), iLQR conventional (solid magenta), LQR with tilt (solid green) and iLQR with tilt (solid blue) trajectories for slalom path

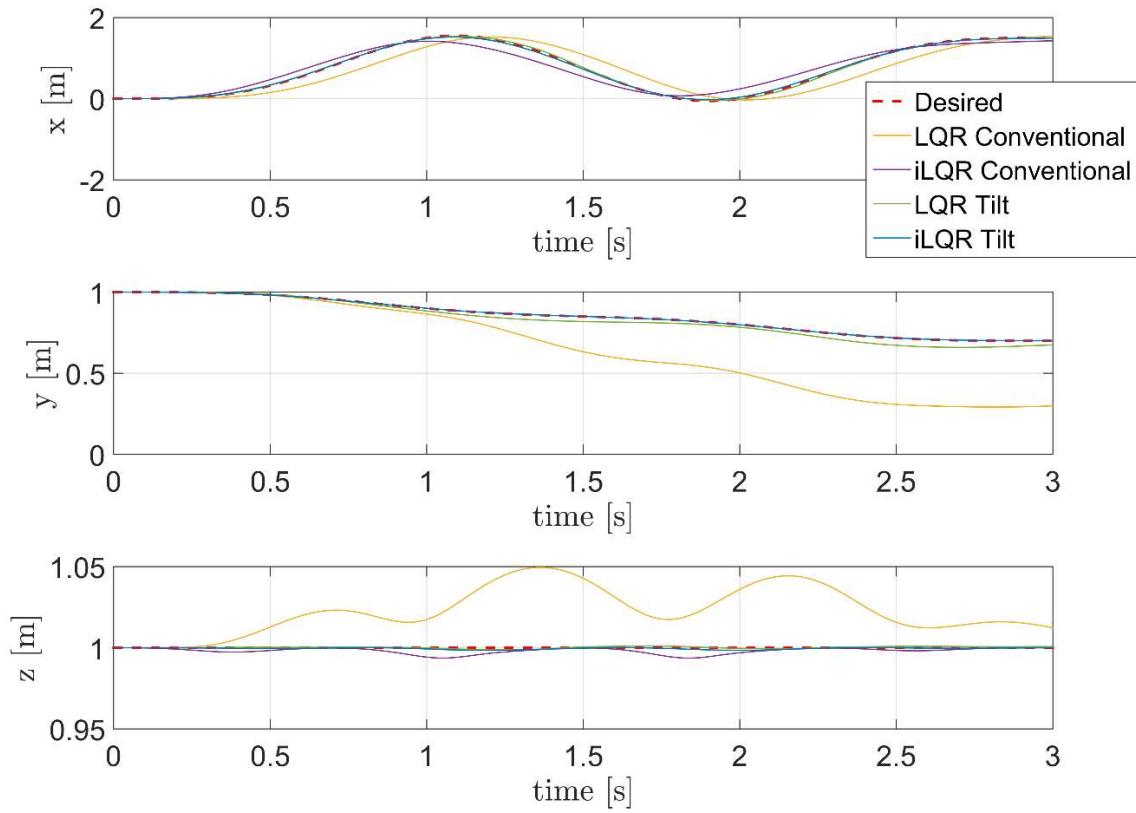


Fig. 4.7. : Desired (dashed red), LQR conventional (solid yellow), iLQR conventional (solid magenta), LQR with tilt (solid green) and iLQR with tilt (solid blue) Cartesian positions for slalom path

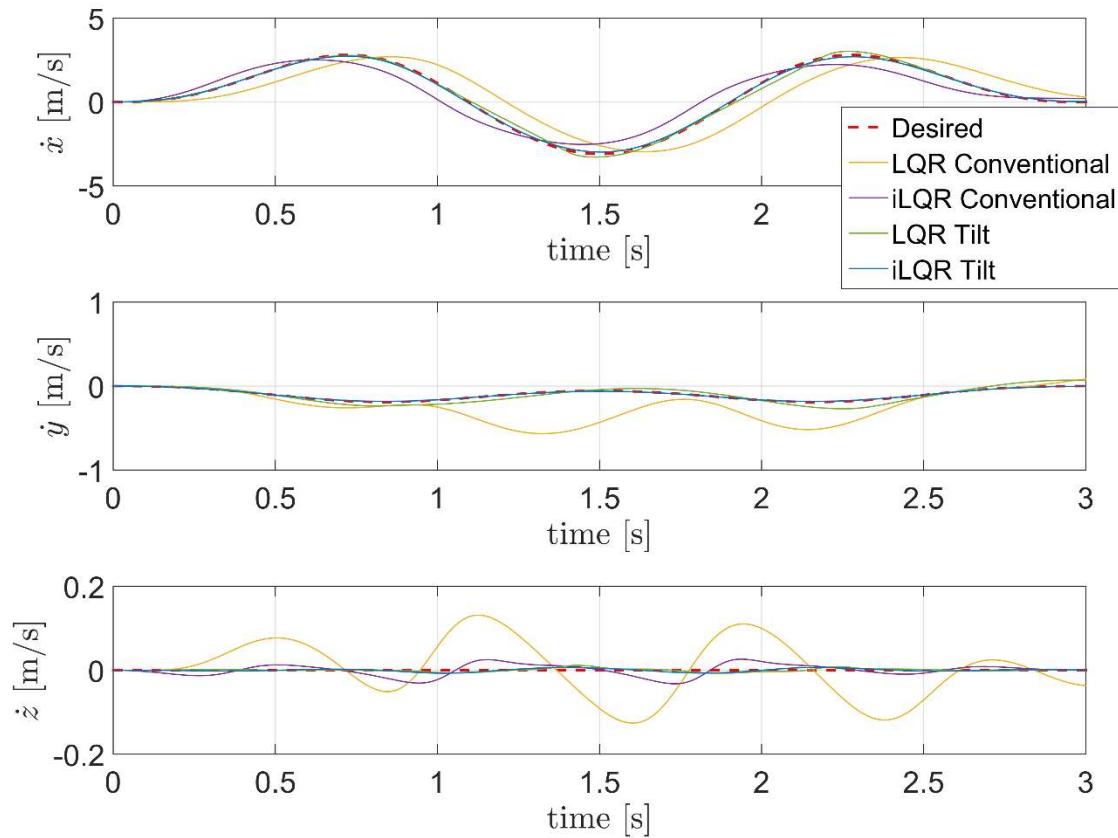


Fig. 4.8. : Desired (dashed red), LQR conventional (solid yellow), iLQR conventional (solid magenta), LQR with tilt (solid green) and iLQR with tilt (solid blue) Cartesian velocities for slalom path

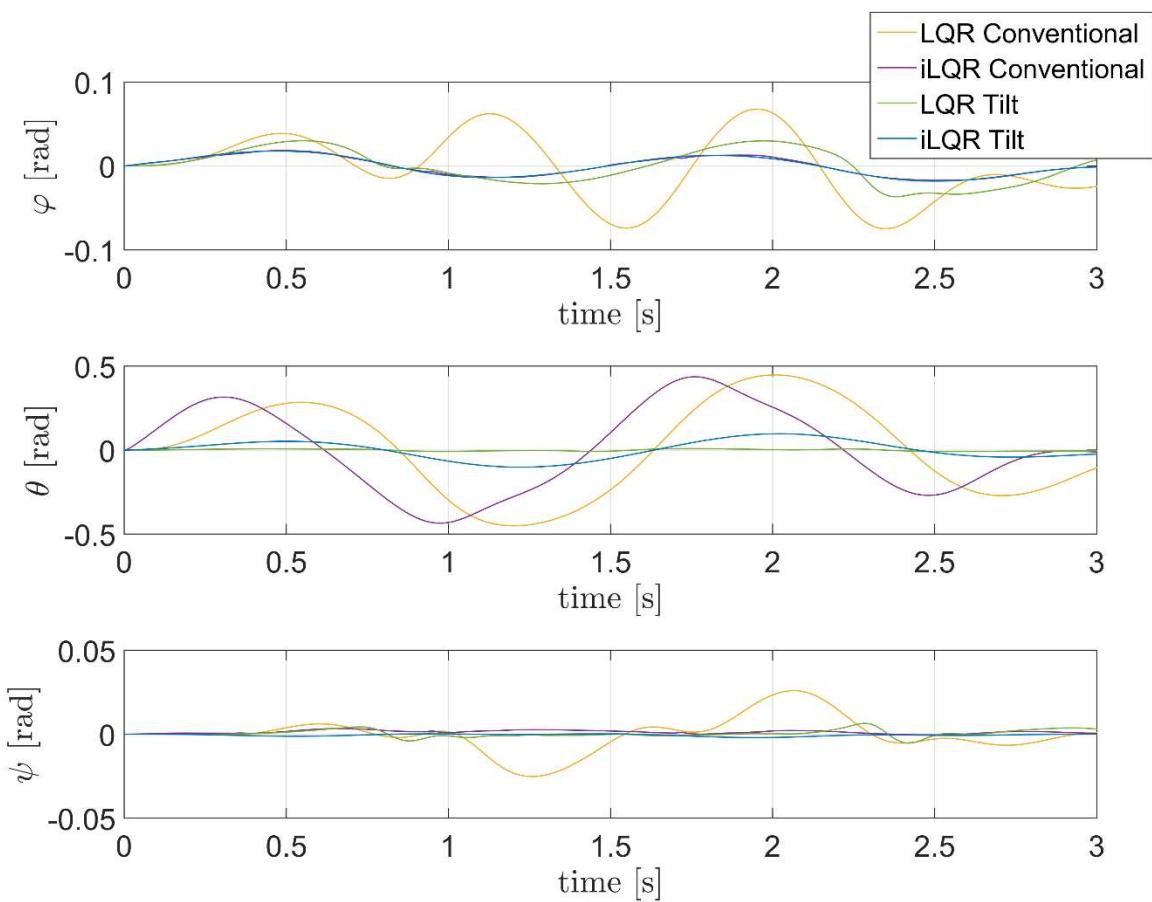


Fig. 4.9. : LQR conventional (solid yellow), iLQR conventional (solid magenta), LQR with tilt (solid green) and iLQR with tilt (solid blue) angles for slalom path

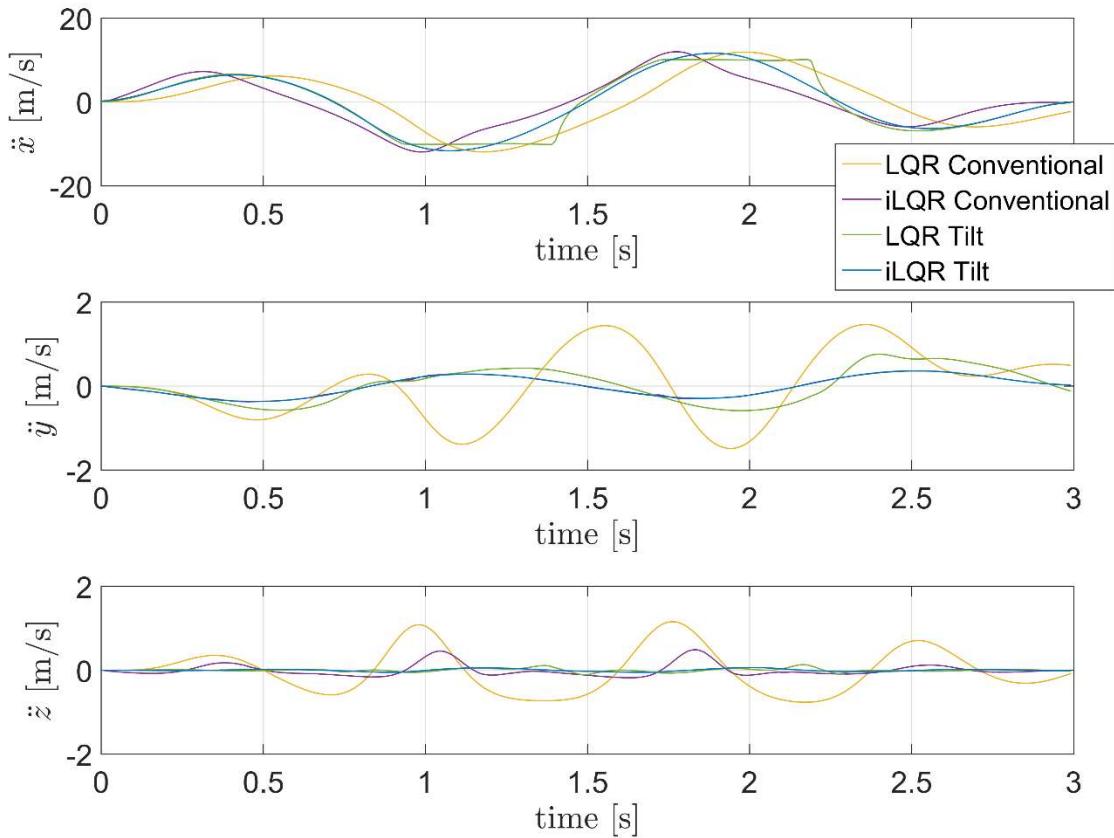


Fig. 4.10. : LQR conventional (solid yellow), iLQR conventional (solid magenta), LQR with tilt (solid green) and iLQR with tilt (solid blue) accelerations for slalom path

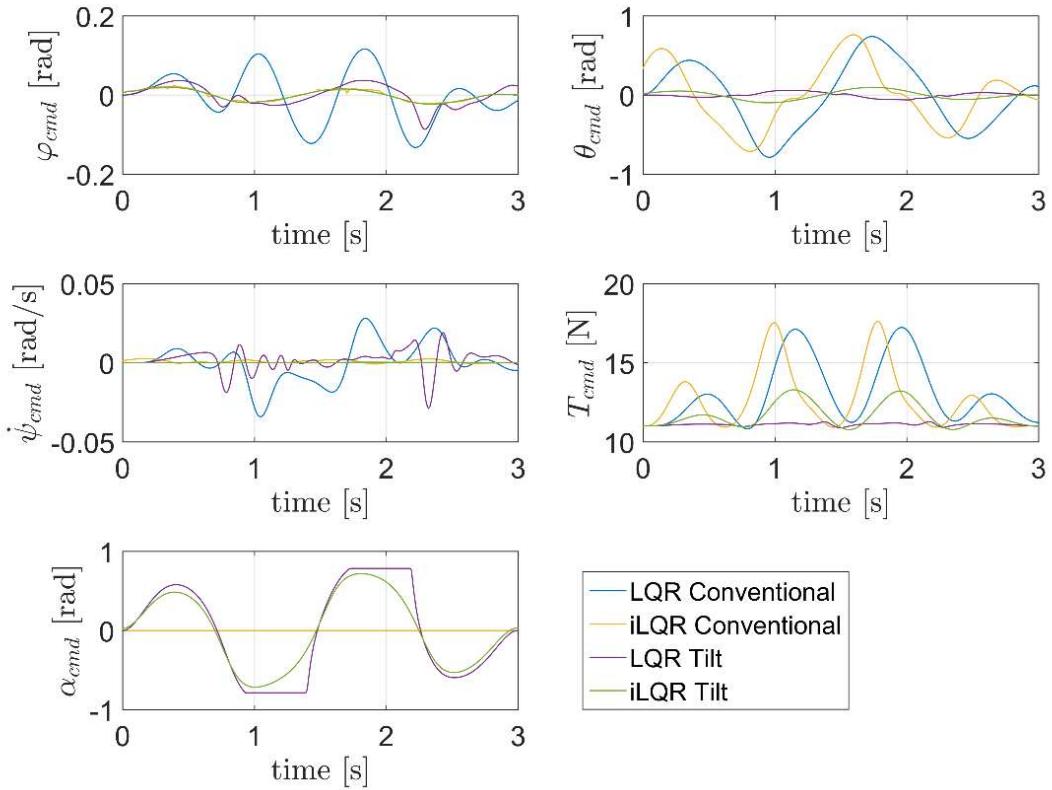


Fig. 4.11: LQR conventional (solid yellow), iLQR conventional (solid magenta), LQR with tilt (solid green) and iLQR with tilt (solid blue) control outputs for slalom path

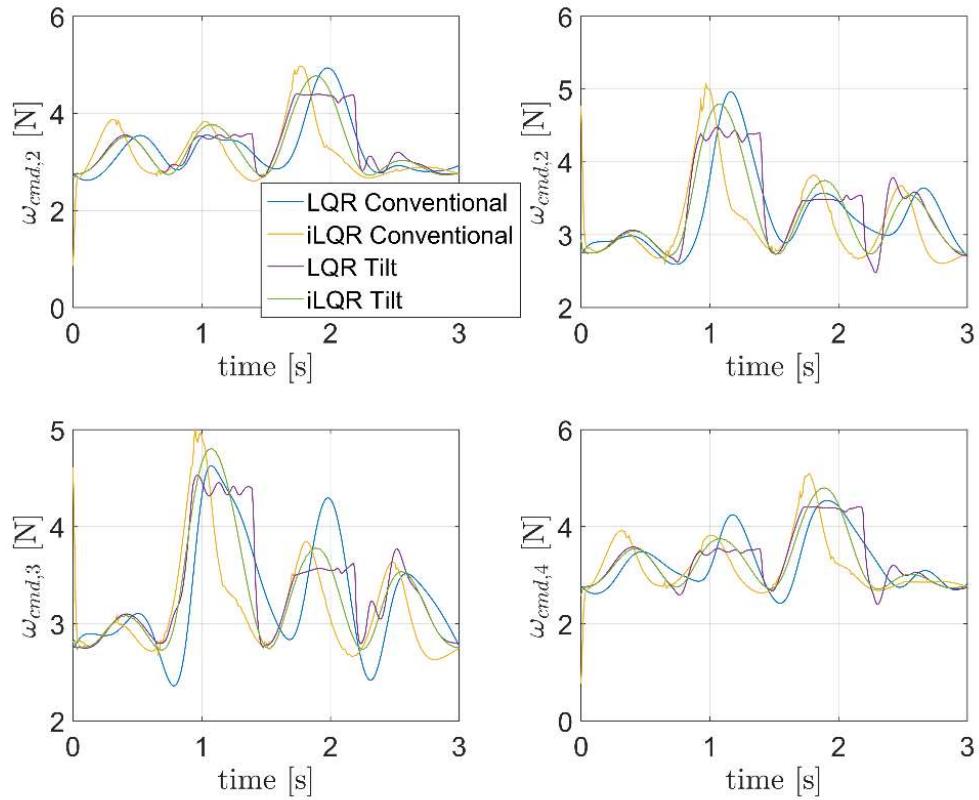


Fig. 4.12: LQR conventional (solid yellow), iLQR conventional (solid magenta), LQR with tilt (solid green) and iLQR with tilt (solid blue) actuator outputs for slalom path

Table 4.2 : Root-mean-squared errors of each controller and design for the slalom case in simulation

Controller and design	Sum Actuators Effort (N)	Max Acc. (m/s ²)	Max Vel. (m/s)	Max Pitch (deg)	Max Roll (deg)	Cartesian Component	RMSE position (m)	RMSE velocity (m)
LQR	3942	12.05	3.02	25.81	4.28	x	0.220	0.841
						y	0.254	0.203
						z	0.027	0.070
iLQR	3822	12.00	2.54	25.02	1.05	x	0.152	0.578
						y	0.001	0.004
						z	0.003	0.010
LQR tilting prop.	4023	10.16	3.29	0.52	2.10	x	0.017	0.136
						y	0.025	0.041
						z	0.001	0.004
iLQR tilting prop.	3950	11.62	2.98	5.82	1.03	x	0.021	0.071
						y	0.001	0.004
						z	0.001	0.004

All configurations (combinations of controllers and mechanics) reach a maximum acceleration above 10 m/s² and a maximum velocity above 2.5 m/s on the slalom path.

4.3.2 Experimental Results

In this section, we report on the experiments performed using the quadrotor platform presented in Section 4.2.3.1. The total flying area is a volume of 4 x 3 x 2 m³. The vehicle has to pass through a narrow gap by getting around two obstacles. The width of the gap is 0.4 m while the vehicle width is 0.3 m. This second trajectory is also challenging because the quadrotor has to travel 2.2 m in 2.2 s. with two full stops while avoiding collisions with the nearby obstacles. In Fig. 4.11, Fig. 4.12, Fig. 4.13 and Fig 4.14, we present both simulation and experimental results for the same aggressive trajectory for comparison purposes and to assess the impact of the neglected effects in simulation on the performance of the real quadrotor. The same prototype with tilting arms

was used for the experiment without tilting arms but $\alpha_{cmd} = 0$ was send to the servo-motor throughout the maneuver. Results are reported in Table 4.3. The gains were manually tuned to:

$$\mathbf{Q} = \text{diag}([1, 20, 1, 2.7, 0.5, 2.2, 0.5, 3.5, 0.5]) \text{ and } \mathbf{R} = \text{diag}([0.1, 5, 10, 10, 5])$$

on the physical quadrotor and

$$\mathbf{Q} = \text{diag}([100, 100, 100, 1, 1, 1, 1, 10, 1]) \text{ and } \mathbf{R} = \text{diag}([0.1, 5, 10, 10]),$$

$$\mathbf{Q} = \text{diag}([1, 1, 1, 100, 10, 100, 1, 100, 10]) \text{ and } \mathbf{R} = \text{diag}([0.1, 5, 10, 10]),$$

$$\mathbf{Q} = \text{diag}([1, 1, 1, 100, 1, 1, 1, 100, 10]) \text{ and } \mathbf{R} = \text{diag}([0.1, 5, 10, 10, 1]),$$

$$\mathbf{Q} = \text{diag}([1, 1, 1, 100, 10, 100000, 1, 100000, 10]) \text{ and } \mathbf{R} = \text{diag}([0.1, 5, 10, 10, 1]),$$

in simulation, for the LQR controller with the conventional design, iLQR controller with conventional design, LQR controller with tilting arms and iLQR controller with tilting arms experiments respectively. To avoid losing control of the prototype due to overshoots, the gains are lower than in simulations.

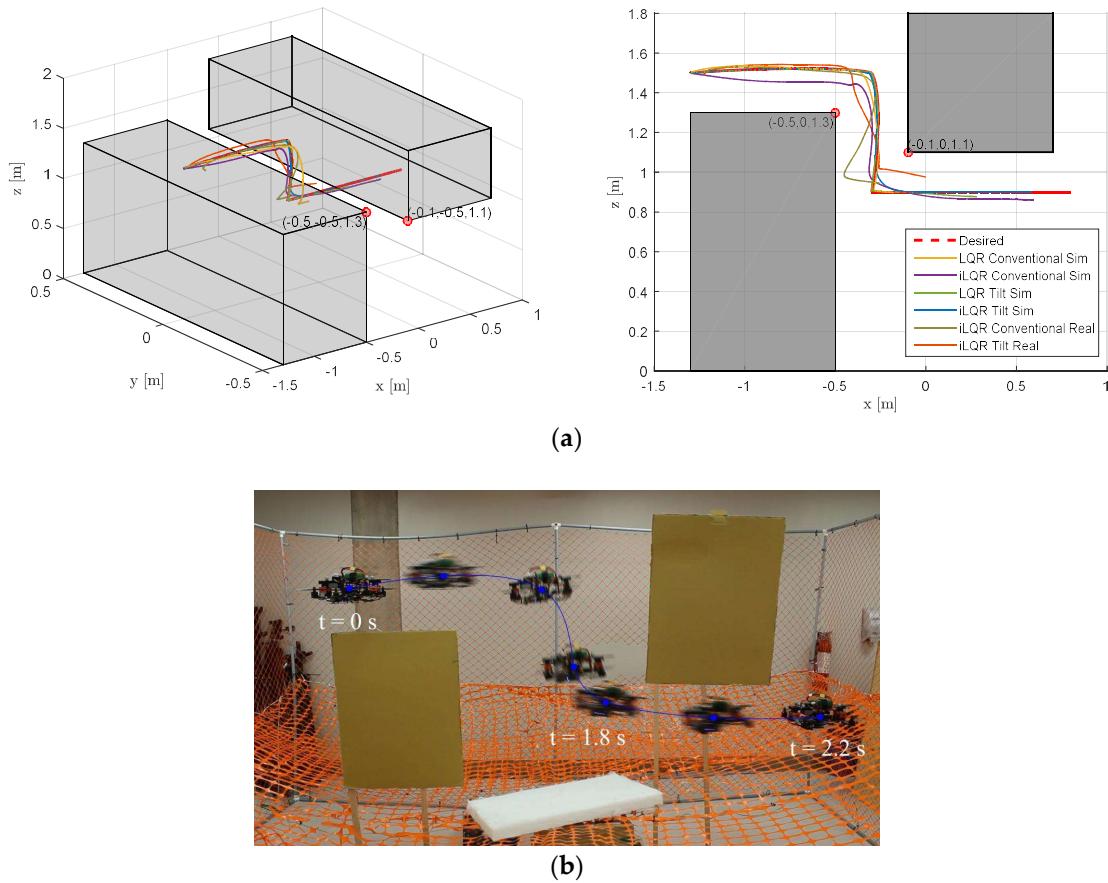


Fig. 4.11. : Desired (dashed red), LQR conventional in simulation (solid yellow), iLQR conventional in simulation (solid magenta), LQR with tilt in simulation (solid green), iLQR with tilt in simulation (solid blue), iLQR conventional in real (solid turquoise) and iLQR with tilt in real (solid blue) trajectories for narrow gap path; (b) Quadrotor prototype with tilting arms and iLQR performing the narrow gap trajectory. Note that the planned trajectory times are written.

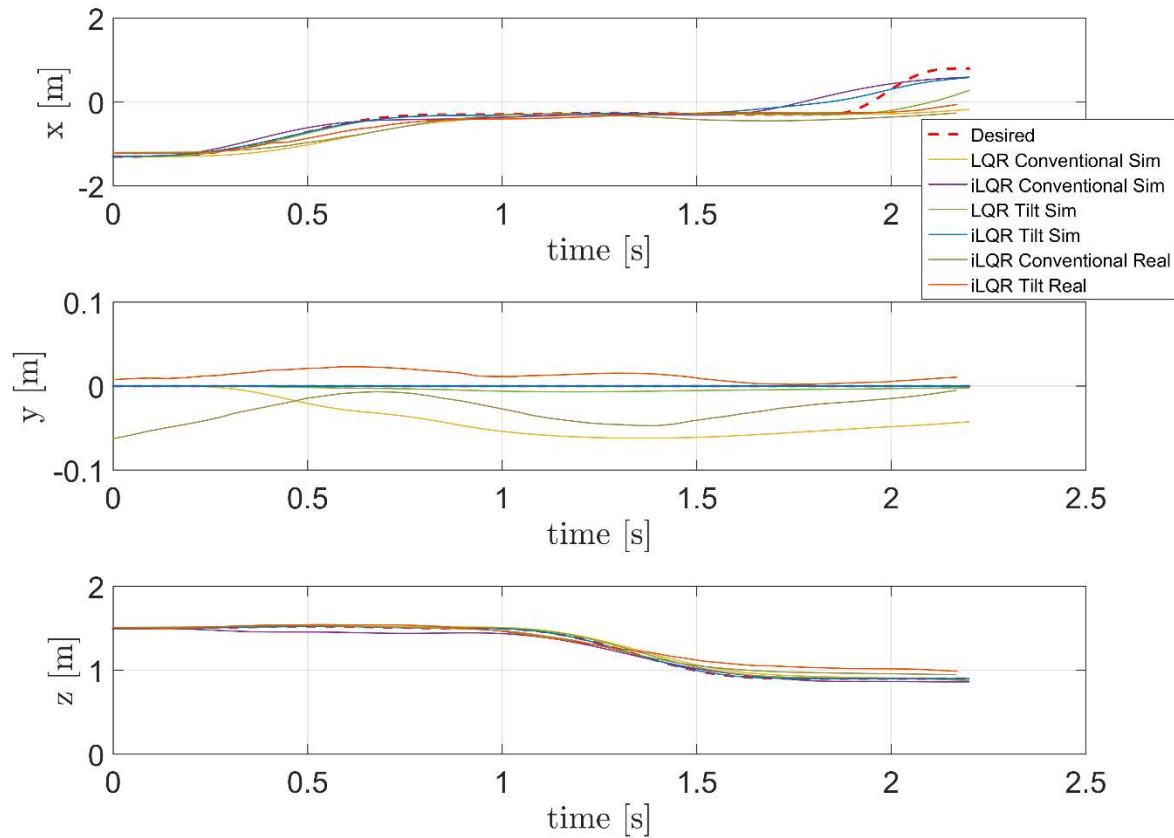


Fig. 4.12. : Desired (dashed red), LQR conventional in simulation (solid yellow), iLQR conventional in simulation (solid magenta), LQR with tilt in simulation (solid green), iLQR with tilt in simulation (solid blue), iLQR conventional in real (solid turquoise) and iLQR with tilt in real (solid blue) Cartesian positions for narrow gap path

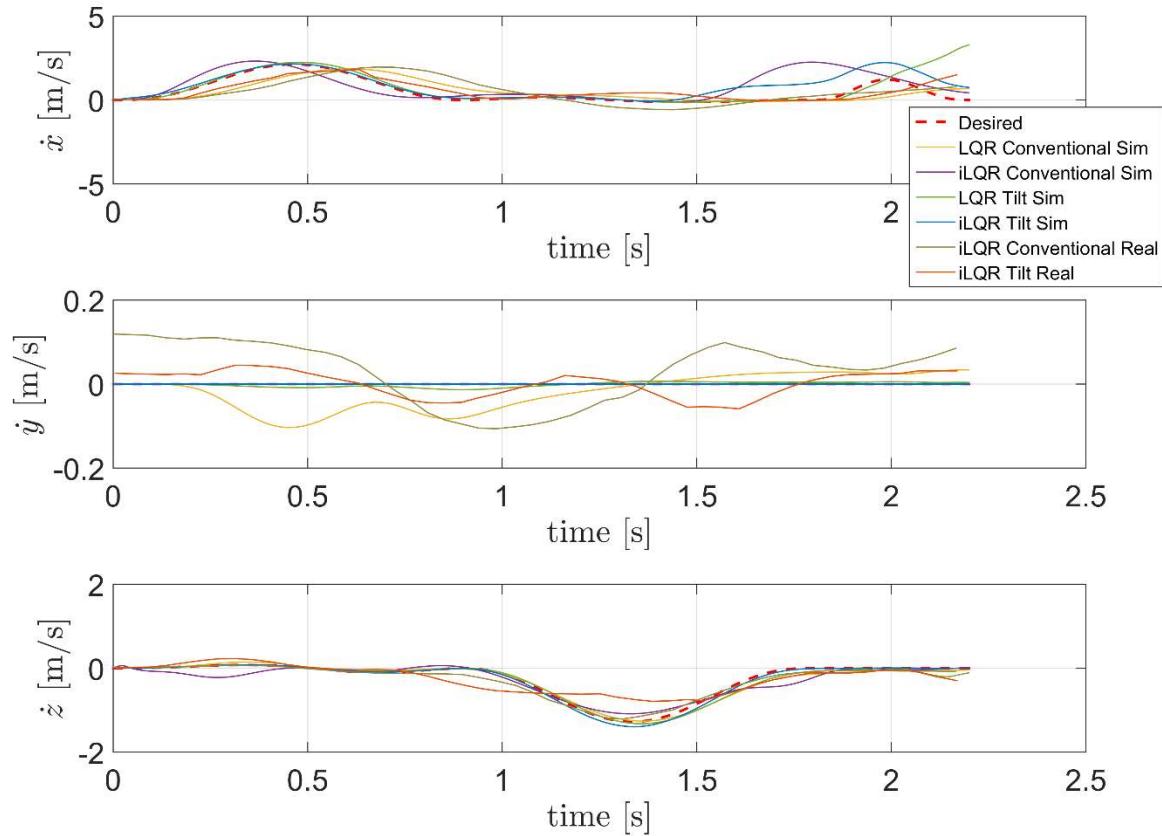


Fig. 4.13. : Desired (dashed red), LQR conventional in simulation (solid yellow), iLQR conventional in simulation (solid magenta), LQR with tilt in simulation (solid green), iLQR with tilt in simulation (solid blue), iLQR conventional in real (solid turquoise) and iLQR with tilt in real (solid blue) Cartesian velocities for narrow gap path

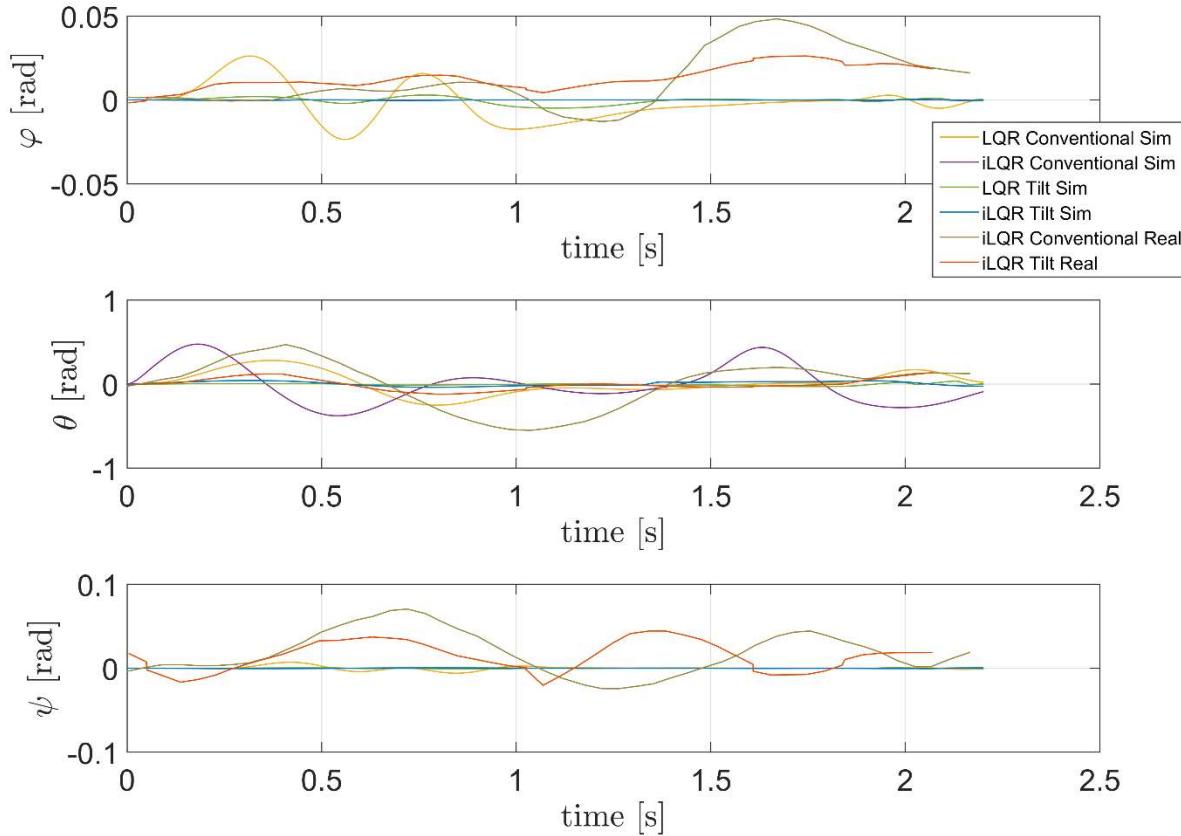


Fig 4.14. : LQR conventional in simulation (solid yellow), iLQR conventional in simulation (solid magenta), LQR with tilt in simulation (solid green), iLQR with tilt in simulation (solid blue), iLQR conventional in real (solid turquoise) and iLQR with tilt in real (solid blue) angles for narrow gap path

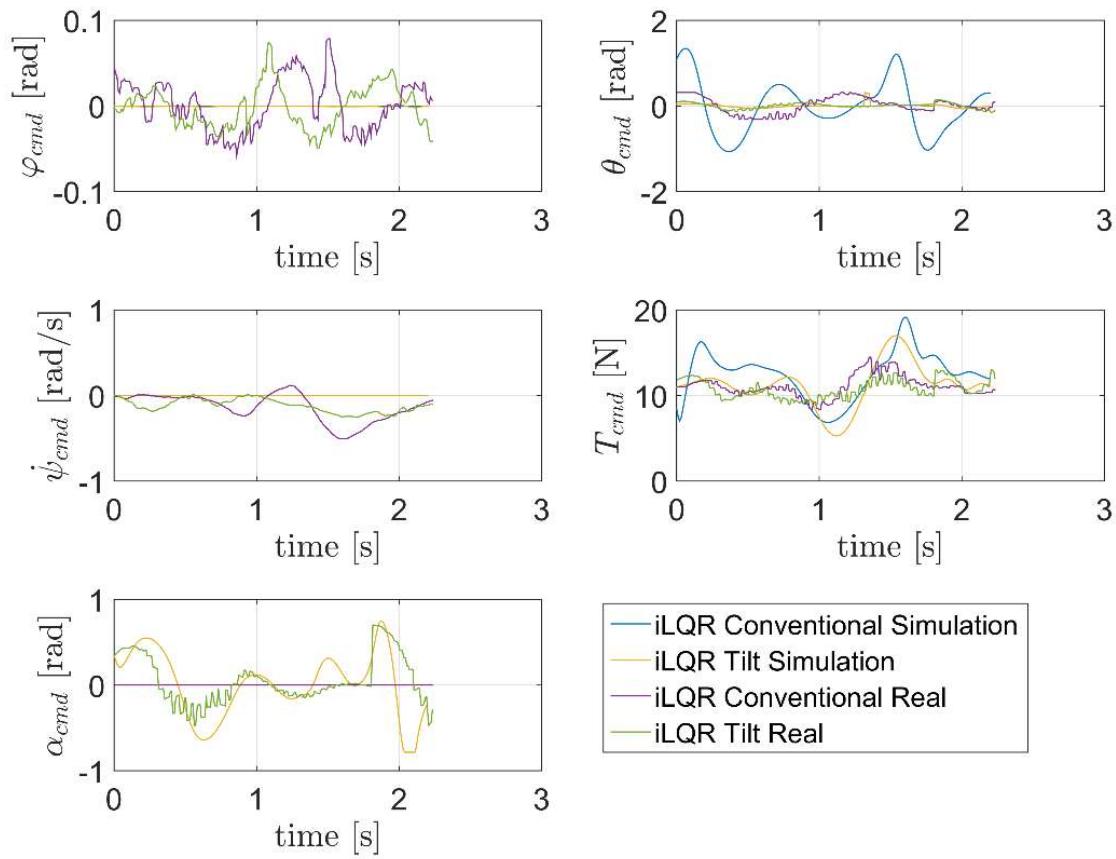


Fig. 4.17: iLQR conventional in simulation (solid blue), iLQR with tilt in simulation (solid yellow), iLQR conventional in real (solid magenta) and iLQR with tilt in real (solid green) control outputs for a narrow gap path

Table 4.3 : Root-mean-squared errors of each controller and design for the narrow gap case experimentally and in simulation

Controller and design	Cost	Max Acc. (m/s²)	Max Vel. (m/s)	Max Pitch (deg)	Max Roll (deg)	Cartesian Component	RMSE position (m)	RMSE velocity (m)
LQR simulation	-	6.07	1.86	16.16	1.50	x	0.320	0.516
						y	0.045	0.047
						z	0.030	0.086
iLQR simulation	7343	12.83	2.31	27.26	0.01	x	0.157	0.775
						y	0.000	0.000
						z	0.046	0.140
LQR tilting prop. simulation	-	12.27	3.28	2.02	0.28	x	0.212	0.646
						y	0.004	0.006
						z	0.018	0.080
iLQR tilting prop. simulation	5167	10.56	2.23	2.46	0.02	x	0.114	0.533
						y	0.000	0.000
						z	0.001	0.049
iLQR real	1446	5.89	2.20	32.34	3.51	x	0.354	0.694
						y	0.032	0.082
						z	0.046	0.112
iLQR tilting prop. real	1045	8.24	2.36	8.30	2.15	x	0.281	0.505
						y	0.014	0.036
						z	0.082	0.227

The simulated quadrotor reaches a maximal acceleration of about 12 m/s² and a maximum velocity of 3 m/s through the narrow gap. The experimental prototype reaches a maximum acceleration of 8.24 m/s² and a maximum velocity of 2.36 m/s following a trajectory of 2 m in length and requiring a break in the middle to avoid an obstacle.

The attached multimedia material shows the experiments performed on the real quadrotor. The aggressiveness of the tilting arm configuration can be observed just as its air breaking capability to avoid obstacles.

4.4 Discussion

The results showed that both using a predictive controller and tilting arms improve the position RMSE of the quadrotor on aggressive trajectories. Indeed, for the quadrotor without tilting arms, the position RMSE on a trajectory with accelerations and velocities up to 12 m/s^2 and 3 m/s with iLQR is improved by 122% compared with the LQR (see Table 4.2). For the quadrotor with LQR, the position RMSE on a trajectory with accelerations and velocities up to 12 m/s^2 and 3 m/s with tilting arms is improved by 1023% compared without the use of tilting arms (see Table 4.3). This design comparison is also made while keeping an iLQR on a real prototype which improves the position RMSE by 22% while following a trajectory with accelerations up to 8.24 m/s^2 in distances as small as 1 m (see Table 4.3). The combination of iLQR and tilting arms provides the best improvements, by 1483% (see Table 4.3), compared with the LQR and without the use of tilting arms. This highlights that it is the combination of iLQR and tilting arms that better improves the aggressiveness of the quadrotor. This can be explained by the capabilities to 1. anticipate changes in accelerations along the trajectories and 2. produce the required accelerations faster. To the best of our knowledge, this paper is the first to compare quadrotors accelerations and tracking performances along aggressive trajectories between a tilting arms design and a conventional one. In [17], the clearing distance between obstacles is improved by 112% for a tilting arms quadrotor while for the narrow gap experiment the position RMSE is improved by 22% (see Table 4.3). While in both cases there is an improvement in position tracking with tilting arms, the results are not comparable because the trajectories are very different.

Without tilting arms, the iLQR outperforms the LQR with a lower RMSE both in positions and velocities while following the slalom path (see Table 4.2). The inability to anticipate the aggressive motion causes the LQR controller to impose high control inputs just as the tracking error builds up and thereafter lose control and deviate widely from the desired path. The iLQR controller has a higher θ than the LQR to begin in order to follow x_{des} whereas the LQR falls behind on the path to follow (see Fig. 4.9). We achieve better results than in [6] where a LQT

improves the position tracking by 75% as compared to an LQR. Indeed, the iLQR controller in this paper outperforms the LQT in [6] by improving the position RMSE by 122% (see Table 4.2) on the slalom path and 98% (see Table 4.3) on the narrow gap trajectory.

However, on the slalom path, adding tilting arms greatly improves both LQR and iLQR RMSEs because the quadrotor now has the ability to accelerate and brake instantaneously which relieves it of its inability to predict the sudden changes of direction. The difference from no tilting is less important with the iLQR controller (see Table 4.2) as it gets rid of the handicap of having to pitch before accelerating. The major difference is the capability to track the desired velocities with an RMSE-x of only 0.07 m (see Table 4.2). The difference between using tilting arms with an iLQR instead of a LQR controller is mostly an improvement in RMSE-y and velocities RMSEs (see Table 4.2). Fig. 4.6 illustrates that only the iLQR controller with tilting arms successfully crosses the two poles.

The virtual control outputs in (20) are illustrated in Fig. 4.11 and Fig. 4.17. The iLQR controller with tilting arms has lower control efforts. Also, the iLQR avoids the saturation of α_{cmd} (see Fig. 4.11). The individual actuators thrusts of the 4 spinning motors are however similar (see Fig. 4.12) with the LQR with tilting arms having the lowest actuators summation over the slalom trajectory (see Table 4.2). Furthermore, adding tilting arms and overcoming the underactuated limitation of the quadrotor enables him to follow more optimal trajectories because the cost (28) on the whole trajectory is lower with tilting arms (see Table 4.3).

On the one hand, these results confirm the robustness of the controllers to non-linearities of the rotational dynamics that were not considered to design the inner attitude loop. On the other hand, they demonstrate the high accuracy tracking ability of the iLQR controller that outperforms the LQR for aggressive trajectories. For trajectories involving sharp turns, adding tilting arms improves the tracking ability of each controller by removing the coupling between rotational and translational motion in the direction in which the arms can orientate the acceleration of the vehicle, which is x in our case (see Fig. 4.6). More specifically, the response time of the orientation loop no longer slows down the time required for the vehicle to acquire a desired translational acceleration since the conventional design with LQR reaches an acceleration of 6.11 m/s^2 in 0.5s as compared to 0.35s for the quadrotor with tilting arms on the slalom path. However, the iLQR

controller with the conventional design can reach 7.2 m/s^2 in only 0.3s but needs to brake faster to negotiate the first turn of the slalom and follow the trajectory as closely as possible. It reaches 0 m/s^2 at 0.62 s as compared to 0.72 s for the tilting arms quadrotor (see Fig. 4.10). Therefore, with tilting arms high accelerations can be commanded without having to worry about breaking too much in advance to avoid overshoot. For comparison, the quadrotor breaks from 2.7 m/s in 0.25 s (see Fig. 4.8) achieving the same deceleration of 10 m/s^2 as in [16] who breaks from 10 m/s to zero in 1s. However, we go a step further by demonstrating comparable accelerations on a real prototype (see Table 4.3).

The configuration with an iLQR controller and tilting arms still outperforms the one with no tilting arms on the real prototype. Specifically, it achieves higher speeds and accelerations enabling better tracking and is closer to the final x_{des} despite that neither configuration reaches it in 2.2 s as desired (see Fig. 4.11). Adding tilting arms improves RMSE-x of 37% in position and 45% in velocity tracking in simulation while on the real prototype these improvements are of 26% and 37%, respectively (see Table 4.3). This indicates that the general trend is the same but the RMSEs absolute values are higher on the real prototype due to control frequencies, delays, vibrations and other non-linearities affecting the overall performance of the quadrotor.

The tilting arms quadrotor with iLQR control achieves higher accelerations compared to some of the quadrotors presented in the literature on aggressive trajectories. Lighter quadrotors reached accelerations of 5 m/s^2 [10,11] and 3.68 m/s^2 [9] while the quadrotor with iLQR and tilting arms in this paper reached 8.24 m/s^2 (see Table 4.3). However, in [8], a 1 kg quadrotor reached speeds of 8.2 m/s and accelerations of 19.62 m/s^2 which is better but it was on trajectories of about 7 m in length while the slalom path presented in this paper is 1.5 m in length (see Fig. 4.6) and the narrow gap is 1 m in length before requiring a full stop for obstacle avoidance (see Fig. 4.11).

Finally, to give a practical perspective, to explore mines the quadrotor needs to travel in corridors as small as 1.5 m in diameter [3]. The quadrotor with an iLQR controller and tilting arms has traveled to speeds of 2 m/s into a corridor of 0.4 m (see Fig. 4.11) making it a good candidate to explore mines. Also, a pitching angle of only 8.3 deg for the tilting arms quadrotor was necessary while the non-tilting arms quadrotor had to pitch to more than 32.34 deg to reach a speed of 2.2 m/s (see Table 4.3) similarly to [10] where angles as high as 40 deg were necessary to reach

speeds of 2.6 m/s. Therefore, the tilting arms design is less close to the edge of the corridor when traveling at high speeds making it safer (see Narrow Gap video in Supplementary Materials). However, in real world applications, the quadrotor presented in this work has to use VIO (Visual Inertial Odometry) for state estimation which has been demonstrated to have errors less than 10cm along aggressive trajectories while a motion capture system is more precise [9]. The impact of such an error on the control algorithms presented in this article could be studied.

4.5 Conclusion

The objective of this paper was to develop an aggressive quadrotor for confined environments. We presented a novel quadrotor design with tilting arms in a single axis with an iLQR controller capable of aggressive trajectory tracking. The mechanical design relieved the quadrotor from the burden of having to rotate before accelerating while the predictive controller anticipated aggressive maneuvers to negotiate sharp turns and stops.

We presented the proposed approach both in simulations and in experimental tests and compared the performance of the iLQR controller with a LQR controller and our quadrotor design with a non-tilting arms quadrotor. The main results were that it is the combination of iLQR and tilting arms that improved the most the position RMSE along aggressive trajectories, by 1483%, compared with the LQR and without the use of tilting arms.

These results were that the addition of tilting capability and the use of predictive control for aggressive trajectory tracking in indoor cluttered environments enhance the tracking performance and allow close obstacle avoidance.

To extend the study, the trajectory generation problem could be integrated into an optimal control problem enabling unified trajectory optimization and tracking. We believe that taking into

account the quadrotor dynamics, and more specifically its ability to tilt into the trajectory generation problem, could allow it to fully exploit its potential to follow aggressive trajectories.

Supplementary Materials: The following is attached to this document, Video: Narrow Gap.

Author Contributions: Conceptualization, G.P. and R.P.; methodology, G.P. and R.P.; software G.P. and R.P.; resources, M.R.; writing—original draft preparation, G.P.; writing—review and editing, M.R., S.A. and D.S.; supervision, M.R. and S.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Polytechnic School of Montreal and by the Natural Sciences and Engineering Research Council of Canada (NSERC).

Acknowledgments: The authors would like to thank all the colleagues who contributed to this research work.

Appendix A

The state space jacobians from (4.26) are defined as:

$$\mathbf{A} = \begin{bmatrix} -\frac{1}{\tau_\phi} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -\frac{1}{\tau_\phi} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ A_{51} & A_{52} & A_{53} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ A_{71} & A_{72} & A_{73} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ A_{91} & 0 & A_{93} & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \quad (4.40)$$

with

$$A_{51} = \frac{T_{cmd}}{m} (s(x_3)c(x_1) + s(x_1)s(x_2)c(x_3)), \quad (4.41)$$

$$A_{52} = \frac{T_{cmd}}{m} (c(x_1)c(x_2)c(x_3) - s(x_2)c(x_3)t(\alpha_{cmd})), \quad (4.42)$$

$$A_{53} = \frac{T_{cmd}}{m} (c(x_3)s(x_1) - c(x_1)s(x_2)s(x_3) - c(x_2)s(x_3)t(\alpha_{cmd})), \quad (4.43)$$

$$A_{71} = \frac{T_{cmd}}{m} (-s(x_1)s(x_2)s(x_3) - c(x_1)c(x_3)), \quad (4.44)$$

$$A_{72} = \frac{T_{cmd}}{m} (c(x_1)c(x_2)s(x_3) - s(x_2)s(x_3)t(\alpha_{cmd})), \quad (4.45)$$

$$A_{73} = \frac{T_{cmd}}{m} (c(x_1)s(x_2)c(x_3) + s(x_1)s(x_3) + c(x_2)c(x_3)t(\alpha_{cmd})), \quad (4.46)$$

$$A_{91} = \frac{T_{cmd}}{m} (-c(x_3)s(x_1)), \quad (4.47)$$

$$A_{93} = \frac{T_{cmd}}{m} (-s(x_3)c(x_1) - c(x_3)t(\alpha_{cmd})), \quad (4.48)$$

and

$$\mathbf{B} = \begin{bmatrix} 0 & \frac{k_\varphi}{\tau_\varphi} & 0 & 0 & 0 \\ 0 & 0 & \frac{k_\theta}{\tau_\theta} & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ B_{51} & 0 & 0 & 0 & B_{55} \\ 0 & 0 & 0 & 0 & 0 \\ B_{71} & 0 & 0 & 0 & B_{75} \\ 0 & 0 & 0 & 0 & 0 \\ B_{91} & 0 & 0 & 0 & B_{95} \end{bmatrix}, \quad (4.49)$$

with

$$B_{51} = \frac{1}{m} (s(x_3)s(x_1) + c(x_1)s(x_2)c(x_3) + c(x_2)c(x_3)t(\alpha_{cmd})), \quad (4.50)$$

$$B_{55} = \frac{T_{cmd}}{m} (c(x_2)c(x_3)(1 + t(\alpha_{cmd})^2)), \quad (4.51)$$

$$B_{71} = \frac{1}{m} (c(x_1)s(x_2)s(x_3) - s(x_1)c(x_3) + c(x_2)s(x_3)t(\alpha_{cmd})), \quad (4.52)$$

$$B_{75} = \frac{T_{cmd}}{m} (c(x_2)s(x_3)(1 + t(\alpha_{cmd})^2)), \quad (4.53)$$

$$B_{91} = \frac{1}{m} (c(x_3)c(x_1) - s(x_3)t(\alpha_{cmd})), \quad (4.54)$$

$$B_{95} = \frac{T_{cmd}}{m} (-s(x_3)(1 + t(\alpha_{cmd})^2)). \quad (4.55)$$

References

- [1] R. Clarke, "Understanding the drone epidemic," *Computer Law & Security Review*, vol. 30, no. 3, pp. 230-246, 2014.
- [2] B. J. Emran and H. Najjaran, "A review of quadrotor: An underactuated mechanical system," *Annual Reviews in Control*, vol. 46, pp. 165-180, 2018.
- [3] P. Gohl et al., "Towards autonomous mine inspection," in *Proceedings of the 2014 3rd International Conference on Applied Robotics for the Power Industry*, 2014, pp. 1-6: IEEE.
- [4] D. Falanga, S. Kim, and D. Scaramuzza, "How fast is too fast? the role of perception latency in high-speed sense and avoid," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 1884-1891, 2019.
- [5] S. Karaman and E. Frazzoli, "High-speed flight in an ergodic forest," in *2012 IEEE International Conference on Robotics and Automation*, 2012, pp. 2899-2906: IEEE.
- [6] E. C. Suicmez and A. T. Kutay, "Optimal path tracking control of a quadrotor UAV," in *2014 International Conference on Unmanned Aircraft Systems (ICUAS)*, 2014, pp. 115-125: IEEE.

- [7] R. Mahony, V. Kumar, and P. Corke, "Multirotor aerial vehicles: Modeling, estimation, and control of quadrotor," *IEEE robotics & automation magazine*, vol. 19, no. 3, pp. 20-32, 2012.
- [8] E. Tal and S. Karaman, "Accurate tracking of aggressive quadrotor trajectories using incremental nonlinear dynamic inversion and differential flatness," in *2018 IEEE Conference on Decision and Control (CDC)*, 2018, pp. 4282-4288: IEEE.
- [9] G. Loianno, C. Brunner, G. McGrath, and V. Kumar, "Estimation, control, and planning for aggressive flight with a small quadrotor with a single camera and IMU," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 404-411, 2016.
- [10] D. Mellinger and V. Kumar, "Minimum snap trajectory generation and control for quadrotors," in *2011 IEEE International Conference on Robotics and Automation*, 2011, pp. 2520-2525: IEEE.
- [11] D. Falanga, E. Mueggler, M. Faessler, and D. Scaramuzza, "Aggressive quadrotor flight through narrow gaps with onboard sensing and computing using active vision," in *2017 IEEE international conference on robotics and automation (ICRA)*, 2017, pp. 5774-5781: IEEE.
- [12] C. Richter, A. Bry, and N. Roy, "Polynomial trajectory planning for aggressive quadrotor flight in dense indoor environments," in *Robotics Research*: Springer, 2016, pp. 649-666.
- [13] M. Kamel, M. Burri, and R. Siegwart, "Linear vs nonlinear mpc for trajectory tracking applied to rotary wing micro aerial vehicles," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 3463-3469, 2017.
- [14] M. Blösch, S. Weiss, D. Scaramuzza, and R. Siegwart, "Vision based MAV navigation in unknown and unstructured environments," in *2010 IEEE International Conference on Robotics and Automation*, 2010, pp. 21-28: IEEE.
- [15] R. Kumar, A. Nemati, M. Kumar, K. Cohen, and F. Cazaurang, "Position and attitude control by rotor tilt and rotor speed synchronization for single axis tilting-rotor quadcopter," in *ASME 2017 Dynamic Systems and Control Conference*, 2017: American Society of Mechanical Engineers Digital Collection.

- [16] A. Alkamachi and E. Ercelebi, "Modelling and control of H-shaped racing quadcopter with tilting propellers," *Facta Universitatis, Series: Mechanical Engineering*, vol. 15, no. 2, pp. 201-215, 2017.
- [17] A. Bin Junaid, A. Diaz De Cerio Sanchez, J. Betancor Bosch, N. Vitzilaios, and Y. Zweiri, "Design and implementation of a dual-axis tilting quadcopter," *Robotics*, vol. 7, no. 4, p. 65, 2018.
- [18] A. Zulu and S. J. a. p. a. John, "A review of control algorithms for autonomous quadrotors," 2016.
- [19] L. Li, L. Sun, and J. Jin, "Survey of advances in control algorithms of quadrotor unmanned aerial vehicle," in *2015 IEEE 16th International Conference on Communication Technology (ICCT)*, 2015, pp. 107-111: IEEE.
- [20] D. Mellinger, N. Michael, and V. Kumar, "Trajectory generation and control for precise aggressive maneuvers with quadrotors," *The International Journal of Robotics Research*, vol. 31, no. 5, pp. 664-674, 2012.
- [21] M. Faessler, A. Franchi, and D. Scaramuzza, "Differential flatness of quadrotor dynamics subject to rotor drag for accurate tracking of high-speed trajectories," *IEEE Robotics and Automation Letters*, vol. 3, no. 2, pp. 620-626, 2017.
- [22] A. Laub, "A Schur method for solving algebraic Riccati equations," *IEEE Transactions on automatic control*, vol. 24, no. 6, pp. 913-921, 1979.
- [23] Y. Tassa, N. Mansard, and E. Todorov, "Control-limited differential dynamic programming," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, 2014, pp. 1168-1175: IEEE.
- [24] E. Todorov and W. Li, "A generalized iterative LQG method for locally-optimal feedback control of constrained nonlinear stochastic systems," in *Proceedings of the 2005, American Control Conference*, 2005., 2005, pp. 300-306: IEEE.
- [25] P. Geoffroy, N. Mansard, M. Raison, S. Achiche, and E. Todorov, "From inverse kinematics to optimal control," in *Advances in Robot Kinematics*: Springer, 2014, pp. 409-418.

- [26] P. Geoffroy, O. Bordron, N. Mansard, M. Raison, O. Stasse, and T. Bretl, "A two-stage suboptimal approximation for variable compliance and torque control," in 2014 European Control Conference (ECC), 2014, pp. 1151-1157: IEEE.
- [27] W. Li and E. Todorov, "Iterative linear quadratic regulator design for nonlinear biological movement systems," in ICINCO (1), 2004, pp. 222-229.
- [28] (26/07/2020). UP board specifications [Web Page]. Available: <https://up-board.org/up/specifications/>
- [29] S. Lynen, M. W. Achtelik, S. Weiss, M. Chli, and R. Siegwart, "A robust and modular multi-sensor fusion approach applied to mav navigation," in 2013 IEEE/RSJ international conference on intelligent robots and systems, 2013, pp. 3923-3929: IEEE.
- [30] (26/07/2020). BTS Bioengineering [Web Page]. Available: <https://www.btsbioengineering.com/products/smart-dx/>
- [31] (26/07/2020). Ubuntu [Web Page]. Available: <https://ubuntu.com/>
- [32] (26/07/2020). Robot Operating System (ROS) [Web Page]. Available: <https://www.ros.org/>

CHAPITRE 5 MÉTHODES ET RÉSULTATS COMPLÉMENTAIRES

Des détails complémentaires à la méthodologie et aux résultats présentés dans l'article de la section précédente sont donnés ici. Ces spécifications ont pour but de permettre de reproduire les résultats obtenus et de justifier certains choix de conception. Plus spécifiquement, l'architecture logicielle utilisée sur le quadrirotor, l'algorithme pour obtenir l'état du véhicule ainsi que des détails sur la conception mécanique sont présentés.

5.1 Architecture logicielle

ROS est un système de mét-exploitation libre roulant sur Unix utilisée dans la communauté robotique. C'est un outil offrant des fonctionnalités comme le passage de messages entre les processus et la gestion de paquet utile lorsqu'on veut implémenter une solution logicielle sur un robot. Un aperçu des différents concepts de cette structure logicielle est donné dans les sous-sections suivantes.

5.1.1 Paquets

Les paquets permettent d'organiser le logiciel dans ROS. Chaque paquet a une utilité dépendamment de ce que l'utilisateur veut en faire. Il contient des processus d'exécution, les nœuds, des bibliothèques, de fichiers de configurations, etc. En combinant plusieurs paquets, il est possible de développer des solutions logicielles plus complexes.

5.1.2 Nœuds

Les nœuds sont les processus qui font les calculs. Ils sont des codes que le développeur robotique peut écrire en différents langages comme le C++ pour accomplir un calcul ou une tâche spécifique sur son robot comme la navigation, la localisation, etc. Un nœud est connecté au réseau de ROS et peut lire les informations qui sont publiées sur le réseau ou lui-même en publier.

5.1.3 Messages

Les nœuds peuvent communiquer entre eux en se passant des messages. Un message est une structure de donnée contenant des champs de différents types comme des entiers, des doubles, des tableaux, etc. Il est également possible de créer des messages avec des types personnalisés.

5.1.4 Sujets

Les sujets ou *topics* sont des messages publiés par les nœuds. Un sujet est unique, mais deux sujets différents peuvent avoir le même type de message. Un nœud peut publier un sujet sur le réseau de ROS pour que d'autres nœuds utilisent le message ou il peut souscrire à un sujet pour utiliser le message qui le contient pour ses calculs.

5.1.5 Implémentation

L'architecture complète du réseau ROS déployée sur le prototype du quadrirotor est illustrée dans la Fig. 5.1.

Les nœuds sont encerclés et les sujets permettant la communication entre les nœuds sont encadrés. Par exemple, `/laird_receiver_node` est un nœud qui envoie les messages `/hummingbird/mav_ilqr_control/trajectory`, `/laird/dataReceived_odo` et `/laird/dataReceived`.

Le réseau contient donc 5 nœuds avec comme tâches de :

- `/laird_receiver_node` : récupérer les données de positionnement fournies par le système BTS envoyées par télémétrie,
- `/hummingbird/mav_ilqr_control` : calculer les commandes à envoyer au contrôleur bas-niveau à partir du positionnement du drone,
- `/msf_vinconpos_sensor` : fusionner l'orientation fournie par le système BTS et l'orientation fournie par l'IMU placé sur le quadrirotor,
- `/servo_control_node` : exécuter la commande choisie par le contrôleur haut-niveau sur le servomoteur permettant d'incliner les bras du quadrirotor,
- `/mavros` : exécuter les commandes fournies par le contrôleur haut-niveau et publier des messages sur l'état du contrôleur de vol PX4 comme le voltage de la batterie. La

communication entre ROS et le PX4 est assurée par ce noeud qui utilise le protocole de communication MAVlink.

Une manette peut également envoyer un signal au PX4 en tout temps permettant de désarmer les moteurs au besoin par mesure de sécurité et sans passer par le réseau ROS.

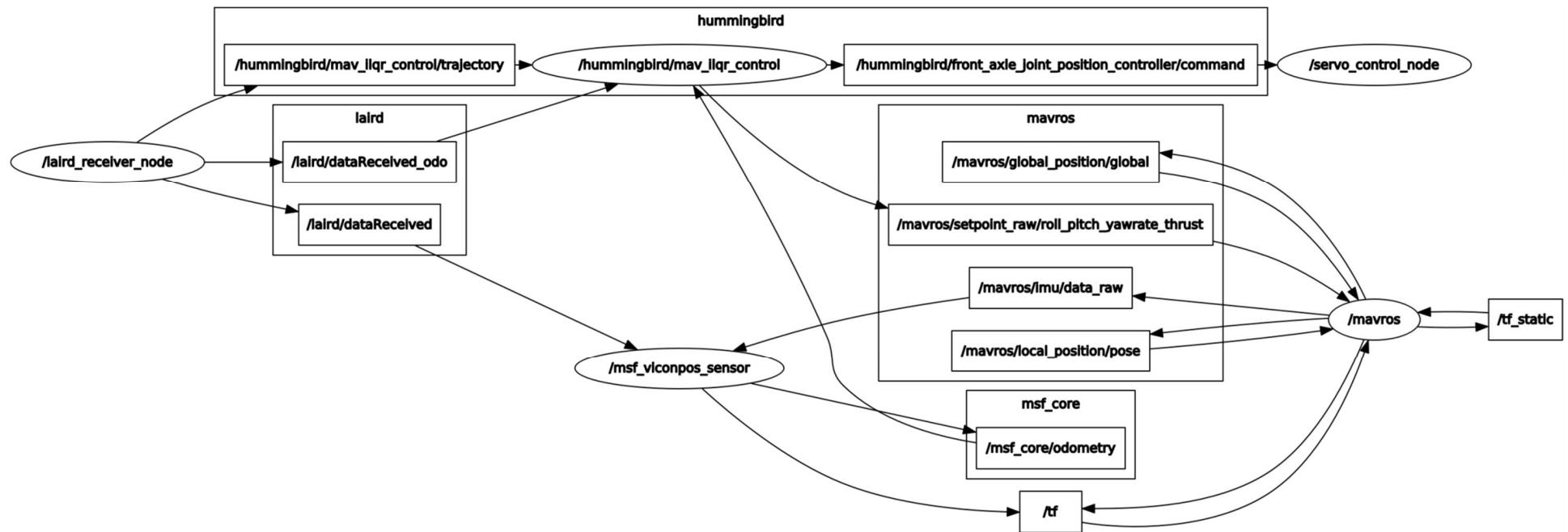


Fig. 5.1. : Architecture ROS sur le quadrirotor

5.2 Estimation de l'état du quadrirotor

Le système de capture BTS² peut fournir la position cartésienne d'un point réfléchissant en utilisant 2 caméras et le principe de stéréovision en temps réel (250fps) et avec une précision inférieure à 0.1mm. La méthode adoptée pour retrouver l'état du quadrirotor consiste en premier lieu à déposer un minimum de trois capteurs réfléchissants sur celui-ci, voir Fig. 5.2. Avec un capteur, la position du centre de gravité du drone peut être trouvée et avec 3 capteurs il est possible de retrouver son orientation.

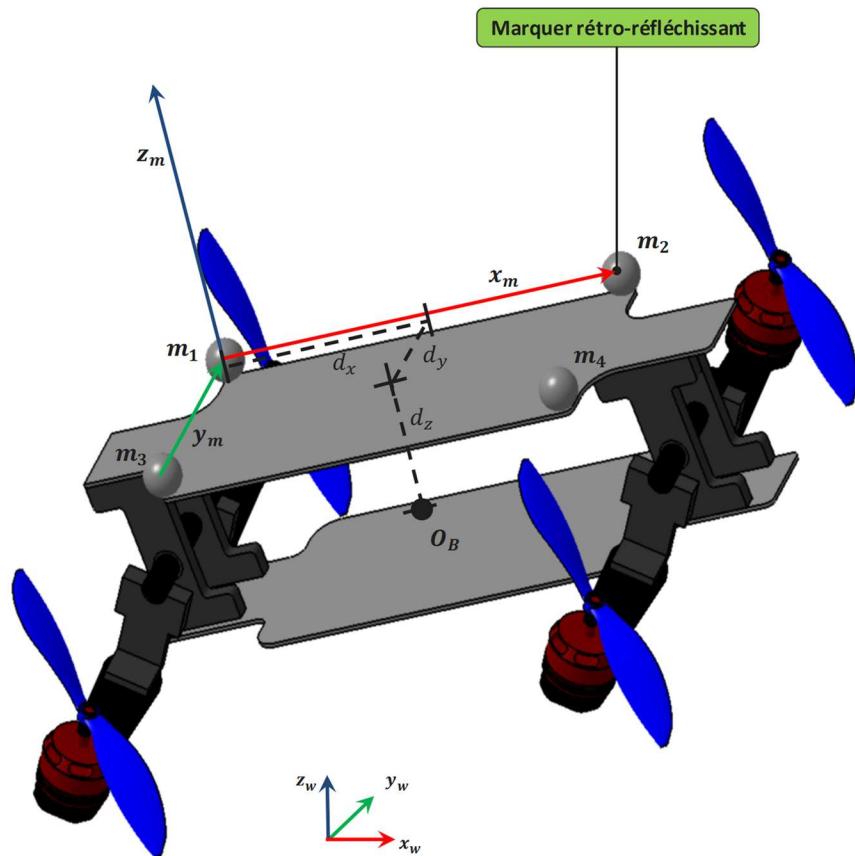


Fig. 5.2. : Schéma représentant la position des marqueurs rétro-réfléchissants et les vecteurs permettant de retrouver l'orientation du drone

² Système de capture BTS Smart-DX 700: <https://www.btsbioengineering.com/products/smart-dx-motion-capture/>

En définissant $\mathbf{m}_i(x_{m_i}, y_{m_i}, z_{m_i})$ la position cartésienne d'un marqueur donné par le système de capture BTS, les vecteurs définissant les axes de \mathcal{F}_B , mais placés au centre du capteur 1 peuvent être retrouvés :

$$\begin{aligned}\mathbf{x}_m &= \mathbf{m}_2 - \mathbf{m}_1, \\ \mathbf{y}_m &= \mathbf{m}_1 - \mathbf{m}_3, \\ \mathbf{z}_m &= \mathbf{x}_m \times \mathbf{y}_m.\end{aligned}\tag{5.1}$$

La matrice de rotation associée à l'orientation du véhicule par rapport à \mathcal{F}_W peut ensuite être défini par les 3 vecteurs unitaires linéairement indépendants \mathbf{e}_{x_m} , \mathbf{e}_{y_m} et \mathbf{e}_{z_m}

$${}^W\mathbf{R}_B = [\mathbf{e}_{x_m}^T \quad \mathbf{e}_{y_m}^T \quad \mathbf{e}_{z_m}^T] = \begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{bmatrix},\tag{5.2}$$

où $\mathbf{e}_{j_m} = [j_{m_x}/\|\mathbf{j}_m\| \quad j_{m_y}/\|\mathbf{j}_m\| \quad j_{m_z}/\|\mathbf{j}_m\|]$ avec $j = x, y$ ou z . En utilisant l'équation (4.1), les angles d'Euler définissant l'orientation du quadrirotor peuvent être retrouvés à partir de la matrice de rotation actuelle défini dans l'équation (5.2). Plus spécifiquement, les angles sont

$$\theta = -\sin^{-1}(R_{31}),$$

$$\varphi = \tan^{-1}\left(\frac{R_{32}}{\cos(\theta)}, \frac{R_{33}}{\cos(\theta)}\right),\tag{5.3}$$

$$\psi = \tan^{-1}\left(\frac{R_{21}}{\cos(\theta)}, \frac{R_{11}}{\cos(\theta)}\right),$$

si l'on considère que ceux-ci sont toujours inférieur à 90 deg [42]. Finalement, la position du centre de gravité \mathbf{O}_B peut être retrouvée en connaissant la position exacte où un marqueur a été placé sur le quadrirotor. Par exemple, à partir du point \mathbf{m}_1 on trouve que

$$\mathbf{O}_B = \mathbf{m}_1 + d_x \mathbf{e}_{x_m} - d_y \mathbf{e}_{y_m} - d_z \mathbf{e}_{z_m}.\tag{5.4}$$

Si un marqueur est perdu, on peut retrouver les vecteurs définissant les axes de \mathcal{F}_B avec 3 autres points, mais les signes des équations (5.1) et (5.4) devront être ajustées en fonction des positions des 3 marqueurs disponibles.

5.3 Conception mécanique et limitations

Le corps du quadrirotor est composé de 3 niveaux de cadre en fibre de carbone de 3mm d'épaisseur. La fibre de carbone est à la fois rigide et légère ce qui l'empêche de se briser lors des impacts au sol et de réduire la transmission des vibrations des moteurs vers l'IMU tout en minimisant la masse totale du véhicule aérien. Les bras sont également en fibre de carbone avec la même épaisseur. Les autres pièces sont soit des pièces standards (écrous, boulons, entretoises, etc.), des composants électriques ou des pièces personnalisées imprimés en 3D. Ces dernières sont en PLA qui est à la fois léger, rigide et peu coûteux. Étant donné que la conception mécanique des bras inclinables est une nouveauté, elle est expliquée en détail dans cette section.

Chaque bras est fixé à un essieu en fibre de carbone qui est lui-même fixé à une poulie de distribution, voir la Fig. 5.3. Cette dernière est enroulée à une courroie de distribution qui relie les 3 poulies composantes le système, voir la Fig. 5.4. C'est la poulie de distribution reliée au servomoteur qui permet d'actionner tout le système et d'incliner les bras à un angle spécifique.

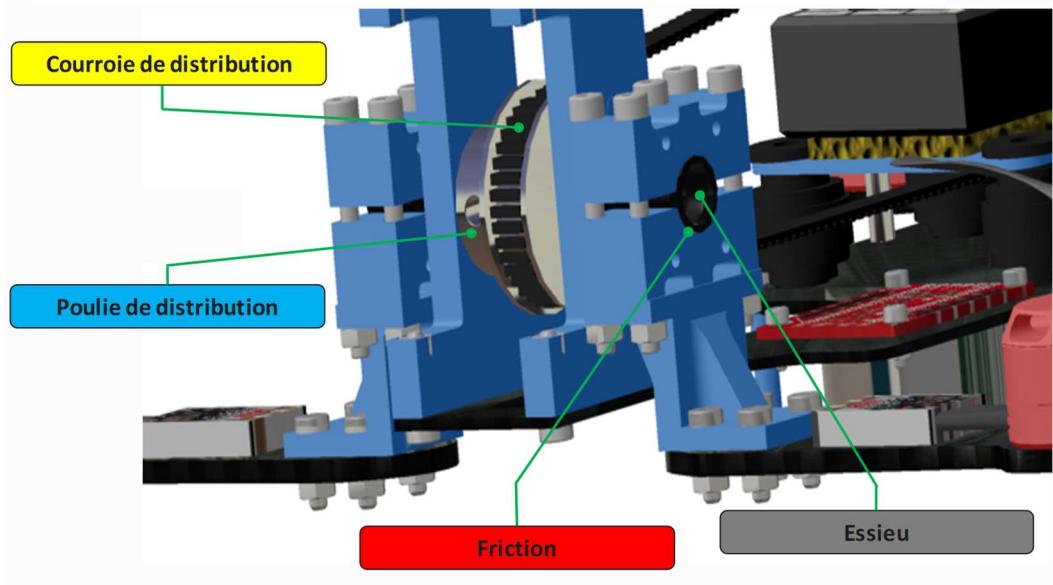


Fig. 5.3. : Liaison du bras avec l'essieu le système de distribution

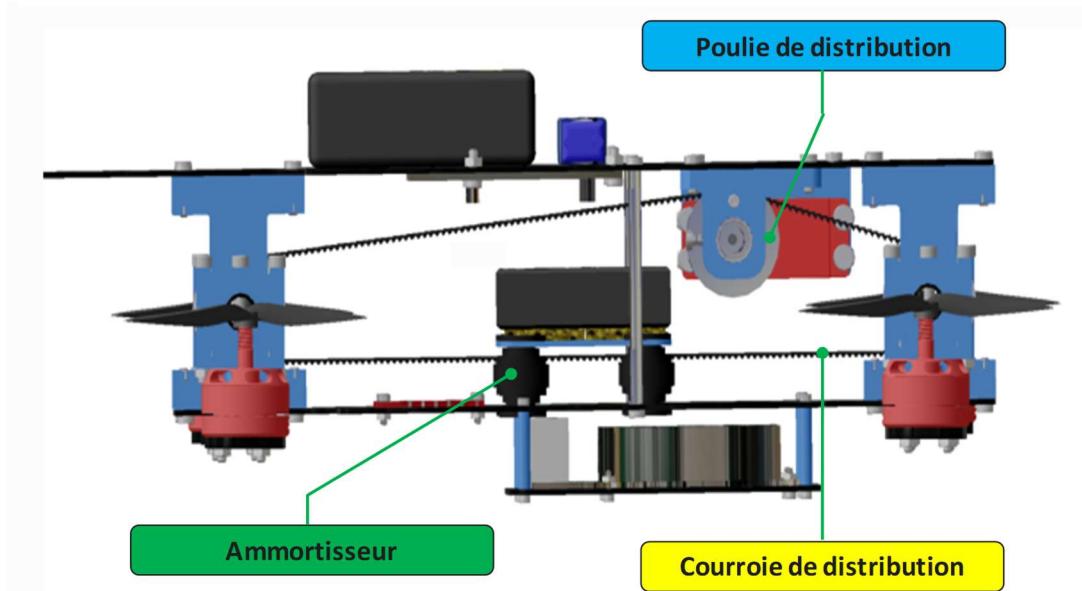


Fig. 5.4. : Principaux composants du système de bras inclinables

Ainsi, un seul servomoteur est nécessaire pour changer l'orientation de la force générée par 4 moteurs ce qui réduit le poids total du quadrirotor et permet de s'assurer que tous les bras ont les mêmes angles au même moment puisqu'ils sont reliés mécaniquement.

Néanmoins, ce système de distribution possède quelques désavantages. Tout d'abord, la fixation du bras à l'essieu est assurée uniquement par de la friction, voir la Fig. 5.3, en serrant deux pièces imprimées en 3D autour de l'essieu avec des boulons. D'une part, l'inclinaison d'un bras par rapport aux autres peut changer en cas d'impact. D'autre part, cette fixation est plus propice à la propagation des vibrations du moteur ce qui nuit au contrôleur de vol.

Ensuite, la courroie en caoutchouc est flexible ce qui empêche d'avoir une précision absolue sur les angles des moteurs tout en créant des vibrations supplémentaires nuisibles. L'ajout de quelques tendeurs métalliques à ressort de torsion sur la courroie permet diminuer ce problème.

Puis, pour permettre à la poulie de distribution de tourner sans friction, un jeu de chaque côté de celle-ci est laissé, voir la Fig. 5.5. Puisqu'uniquement la tension dans la courroie maintient la poulie centrée, il arrive qu'elle se déplace, ce qui désaxe les bras. L'emplacement des forces générées par les moteurs par rapport au centre de masse du quadrirotor n'est plus le même. Le modèle dynamique n'est plus tout à fait fidèle à la réalité ce qui peut nuire à la performance globale du

véhicule. En pratique, la robustesse du contrôleur PID de la boucle de contrôle d'orientation permet d'atténuer ce problème.

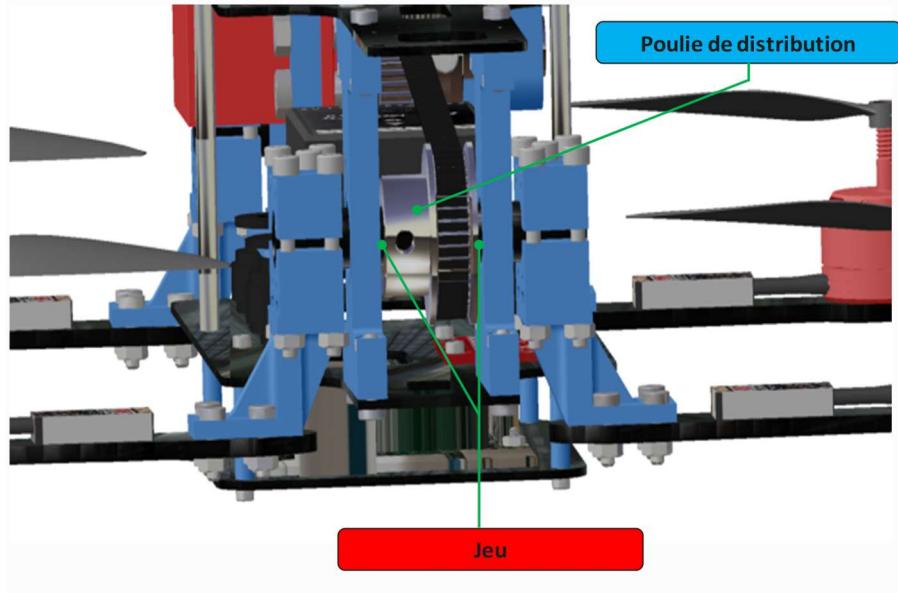


Fig. 5.5. : Jeu autour d'une poulie de distribution

Finalement, malgré la présence d'amortisseur sous le contrôleur de vol, voir la Fig. 5.4, les vibrations qui arrivent aux contrôleurs de vol sont élevé ce qui augmente le bruit perçu par l'IMU. En vol stationnaire, les accélérations brutes perçues par le contrôleur de vol sont de $\pm 4 \text{ m/s}^2$ en x, y et de $\pm 5 \text{ m/s}^2$ en z, voir la Fig. 5.6.

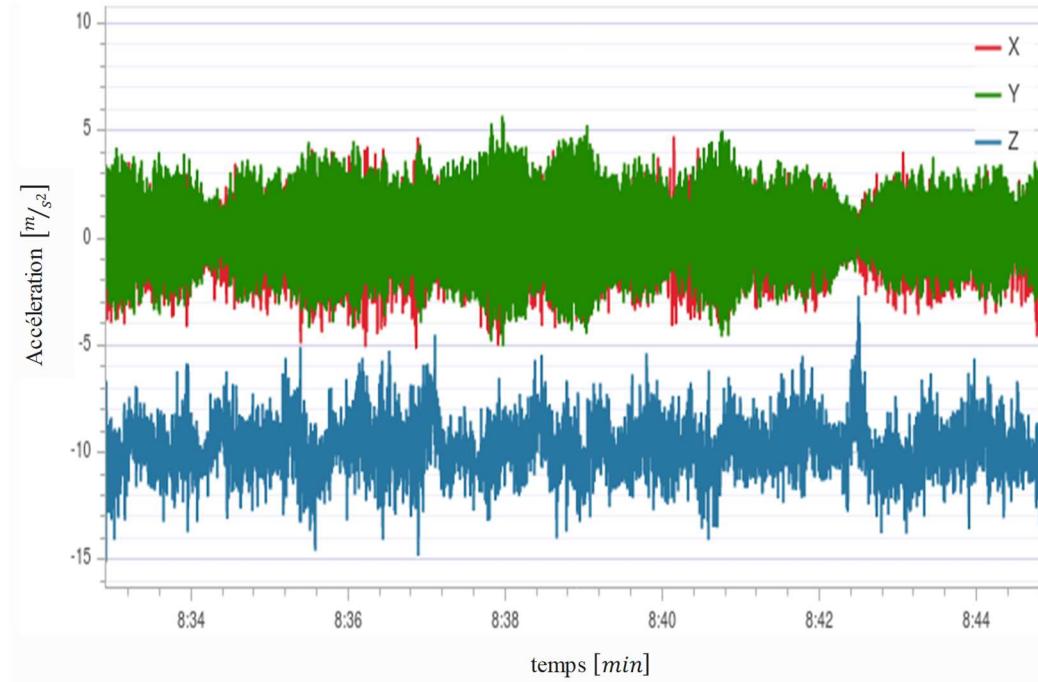


Fig. 5.6. : Accélérations brutes perçues par le PX4 en vol stationnaire

Or, des accélérations brutes de $\pm 2 \text{ m/s}^2$ sont conseillées pour une performance optimale [43]. Ce niveau élevé de bruit requiert de diminuer la fréquence des filtres passe-bas ce qui empêche l'IMU de capter des changements soudains d'angles et de vitesses angulaires. La boucle de contrôle des angles est affectée et des angles élevés ne peuvent pas être atteints aussi rapidement que voulue. Il devient également impossible d'intégrer l'accéléromètre et de fusionner la position trouvée avec celui-ci avec la position fournie par le système de capture et ainsi augmenter la fréquence de l'estimation d'état. En somme, les vibrations réduisent l'agressivité qu'il est possible d'atteindre avec le quadrirotor.

CHAPITRE 6 DISCUSSION GÉNÉRALE

L'objectif général de ce mémoire était de développer un nouveau design de quadrirotor avec son contrôleur capable de suivre des trajectoires agressives dans des environnements encombrés. Des accélérations dépassant 3m/s^2 et des vitesses de plus de 1m/s devaient être atteintes sur de courtes distances pour qu'une manœuvre soit considéré comme agressive et exploitant pleinement les capacités du véhicule aérien. Cet objectif a été atteint puisque, d'une part, un quadrirotor avec bras inclinables a été développé lui permettant d'accélérer dans une direction donnée presque instantanément. D'autre part, un contrôleur LQR itératif et dépendant de l'état du robot a été développé, implémenté puis testé permettant de mieux négocier des manœuvres agressives tout en décélérant rapidement pour éviter des obstacles. Le présent chapitre présente un résumé de la discussion de l'article présenté au Chapitre 4. Puis les limites du projet sont discutées et des pistes d'améliorations sont données.

6.1 Synthèse de l'article

Dans l'article présenté au Chapitre 4, un quadrirotor avec des moteurs inclinables dans un axe est présenté puis un contrôleur LQR et ILQR est développée pour contrôler ce système dynamique. Des essais en simulation en utilisant VREP et expérimentaux sur un prototype fabriqué au laboratoire sont présentés. Nous discutons donc de ces deux lots de tests.

6.1.1 Simulation

Pour démontrer l'utilité d'ajouter un actionneur, mais aussi d'adopter une loi de contrôle prédictive, 4 configurations ont été testées, montrées dans le Tableau 6.1.

Tableau 6.1. : Configurations du quadrirotor mises à l'essai en simulation

Configuration	Design	Contrôle
1	Conventionnel	LQR
2	Conventionnel	ILQR
3	Moteurs inclinables	LQR
4	Moteurs inclinables	ILQR

Chacune des configurations a été testées sur un slalom requérant de grandes vitesses et accélérations dans l'axe des moteurs inclinables. Avec un design conventionnel, le ILQR a une erreur en suivi de trajectoire inférieur tant en position qu'en vitesse. Il anticipe mieux les virages en atteignant des angles d'inclinaison élevés plus tôt que le LQR ce qui lui permet de ralentir à temps. En comparant le contrôleur LQR sur un design conventionnel ou à moteurs inclinables, la même hausse en performance est observée puisque les moteurs inclinables permettent d'accélérer plus rapidement sans incliner tout le quadrirotor. Néanmoins, l'impact sur les erreurs en suivi de trajectoire est plus important en incluant des moteurs inclinables qu'en utilisant une loi de contrôle prédictive. L'erreur pour suivre la position désirée en x diminue de 1023% avec le changement de design alors qu'elle diminue de 122% avec la commande prédictive. Sur des trajectoires nécessitant plus de 10m/s^2 d'accélération en x , pouvoir orienter le vecteur de force généré par les actionneurs instantanément est avantageux. Puis, la meilleure performance est obtenue avec un quadrirotor à moteurs inclinables et commande prédictive surtout pour l'erreur sur les vitesses désirées. L'erreur sur v_x par exemple est de 1084% inférieur comparativement au design et contrôle conventionnel.

6.1.2 Expérimental

Le quadrirotor développé au laboratoire a été mis à l'essai sur une trajectoire nécessitant d'éviter un obstacle tout en circulant rapidement. Cette fois-ci seulement un contrôleur iLQR a été utilisé puisqu'il a été démontré en simulation qu'il est plus performant pour des trajectoires agressives. Toutefois, la capacité d'incliner les moteurs a été désactivée puis activée pour tirer des conclusions sur l'impact du design. Pour atteindre une accélération de 5.89m/s^2 en x , le design conventionnel a dû atteindre un angle de tangage de 32° . Avec l'ajout de moteurs inclinables, le quadrirotor atteint une accélération plus élevée, de 8.24 m/s^2 et les erreurs sur x et v_x désirées sont inférieures de 26% et 37% respectivement. Le véhicule arrive également à la position finale plus rapidement qu'avec un design conventionnel tout en évitant l'obstacle. La même trajectoire a également été testée en simulation pour voir l'impact des imprécisions apparaissant dans le système réel. La tendance observée est la même, c'est-à-dire que l'utilisation de moteurs inclinables améliore d'environ le même pourcentage, 38%, le suivi en x , bien que la trajectoire est complétée plus rapidement en simulation et des accélérations plus élevées sont atteintes.

6.2 Limites et perspectives

L'objectif de ce mémoire était d'augmenter l'agilité d'un quadrirotor. Deux approches ont été utilisés, utiliser du contrôle prédictif et changer le design conventionnel du robot en ajoutant un actionneur lui permettant d'accélérer plus rapidement dans la direction souhaitée. Les résultats obtenus ont permis de confirmer que ces deux modifications au quadrirotor conventionnel permettent en effet d'augmenter ces capacités à suivre une trajectoire agressive. Or, plusieurs simplifications limitent la performance atteignable par le véhicule et dans le but d'effectuer des manœuvres encore plus agiles tout en diminuant les erreurs en suivi, plusieurs améliorations sont envisageables. De plus, les développements nécessaires à la plateforme expérimentale pour avoir un produit fini qui puisse effectuer des opérations de secours ou circuler dans les forêts ou mines à haute vitesse sont discutés.

6.2.1 Design

La conception mécanique est un aspect important dans le développement d'un produit robotique. Le design doit être optimisé pour la tâche que le robot doit accomplir [44-47]. Il peut toujours être amélioré surtout pour un quadrirotor qui doit voler dans les airs et par conséquent être le plus léger possible. La première amélioration possible est donc l'optimisation des dimensions des pièces et le choix des matériaux pour rendre le quadrirotor le plus léger possible tout en le gardant robuste pour résister aux chutes. Plusieurs pistes sont envisageables. D'une part, certaines pièces imprimées en 3D devraient être usinés dans un matériau plus rigide, mais léger comme de l'aluminium. La pièce la plus critique et qui a souvent brisé durant les impacts au sol est la jonction entre le bras et l'essieu. En rigidifiant cette pièce, les vibrations provenant du moteur diminueraient aussi ce qui est important pour les performances du quadrirotor comme il a été vu dans la section 5.3. Les vibrations seraient également atténues si la fixation à l'essieu n'était pas uniquement basée sur le frottement par contact. Un double bras incluant l'essieu central conçu comme une seule pièce rigide en fibre de carbone pourrait être envisageable. D'autre part, les différents jeux présents dans le système mécanique de la distribution de la rotation du servomoteur devraient être éliminés. Plus spécifiquement, la poulie de distribution devrait être en contact direct avec les parois et pouvoir tourner en s'assurant que les matériaux sur les extrémités glissent. La courroie devrait être

remplacée par un matériau non-extensible pour s'assurer de la précision de l'angle d'inclinaison commandé. Utiliser une tige métallique se déplaçant horizontalement et reliant les deux essieux est envisageable.

La cause principale des vibrations arrivant au contrôleur de vol provient de la faiblesse de la jonction du moteur au corps principal du quadrirotor. Cette jonction est faible dû au décalage entre le bras supportant le moteur et l'essieu pour que l'hélice arrive dans l'axe de l'essieu. Cette modification au design conventionnel du quadrirotor n'a pas été observée dans la littérature jusqu'à présent même pour des multirotors avec des moteurs inclinables. Néanmoins, cela permet d'éviter que le point d'application de la force générée par chaque moteur change lorsque le bras s'incline et par le fait même change le centre de gravité du véhicule. Or, il serait pertinent d'étudier l'impact réel de cette modification sur les performances du quadrirotor et de voir si le compromis en vaut la peine ou s'il vaut mieux avoir moins de vibrations pour une performance globale du quadrirotor accrue.

Finalement, utiliser des composantes électroniques moins encombrantes permettrait d'avoir plus d'espace au centre de la plateforme tout en diminuant la masse pour une plus grande agilité. L'ordinateur de bord pourrait être remplacé par un modèle plus coûteux, mais moins lourd et plus performant. L'espace verticale au centre du quadrirotor devrait également être diminuée en rapprochant les deux plaques en fibre de carbone. Cela isolerait mieux le contrôleur de vol vis-à-vis des vibrations. De plus, les composantes devraient être protégées en rajoutant des plaques en fibre de carbone sur les côtés.

Globalement, nous pensons que le quadrirotor présenté dans ce mémoire pourraient être plus petit, plus robuste, moins lourd et plus performant en optimisant le design mécanique et le choix des composantes.

De plus, résoudre complètement le problème de sous-actuation du quadrirotor serait possible en rajoutant un actionneur pour incliner les rotors dans l'axe des y en plus de l'axe des x . Par exemple, 4 servo-moteurs pour chacun des bras pourraient faire tourner les rotors sur un deuxième axe, mais le design mécanique devra être différent en surélevant les moteurs pour éviter une collision avec le bras lors de leurs rotations par rapport à cet axe.

6.2.2 Modèle dynamique

La modélisation du système dynamique du quadrirotor présenté dans ce mémoire comporte plusieurs simplifications. Tout d'abord, la force de traîné n'est pas prise en compte. Les effets de la traînée induite par les rotors sur la dynamique du quadrirotor ont été étudiée dans [48] et [49]. Cette traînée provient des battements de l'hélice qui n'est pas parfaitement rigide et de la traînée induite par la force de portance qui permet d'ailleurs de contrôler l'angle de lacet du véhicule. Dû à leurs équivalents mathématiques, ces effets sont souvent modélisés comme un paramètre constant linéairement proportionnel à la vitesse du quadrirotor [50]. Les résultats obtenus dans [19] montrent qu'à une vitesse de 3 m/s, le quadrirotor peut être sujet à un accélération d'environ -1 m/s^2 . En tenant compte de ces effets dans le modèle dynamique du quadrirotor, l'erreur en suivi de trajectoire peut être atténuée surtout à des vitesses dépassant 2 m/s.

Puis, inclure la dynamique des actionneurs permettrait également d'accroître la performance du quadrirotor. D'une part, le servomoteur n'atteint pas l'angle désiré instantanément. Dans [51], en utilisant une stratégie de compensation de la latence standard, les erreurs en maintien de position diminue de plus de 6cm. Dans [40], il a été conclu que le délai des servomoteurs installés sur un quadrirotor à moteurs inclinables dégrade significativement la performance en boucle fermée. Un délai de plus de 18ms a été trouvé et compensé à l'aide d'un schéma de prédition de Smith [52]. Néanmoins, une mesure de l'angle des servomoteurs est disponible à haute fréquence ce qui n'est pas le cas dans le quadrirotor présenté dans ce mémoire. Cet ajout complexifie le schéma de communication et nécessiterait des pièces électroniques supplémentaires donc une étude détaillée de l'augmentation des performances devrait être mené. D'autre part, rajouter des encodeurs optiques pour connaître la vitesse réelle de rotation des moteurs en temps réel permet également d'améliorer la précision du suivi de trajectoire agressive [24]. En somme, les limites réelles des actionneurs devraient être prises en compte dans la modélisation du système dynamique.

6.2.3 Contrôle

L'algorithme ILQR choisi pour le contrôleur de la boucle externe s'est avéré avantageux pour des trajectoires agressives. Néanmoins, il n'a pas été mis à l'épreuve en cas de perturbations extérieures comme du vent ou un léger impact comme une roche lancée sur le quadrirotor. Ces scénarios sont

probables pour les applications possibles de ce produit présenté au Chapitre 1. Le contrôleur doit pouvoir gérer ces imprévus. Tout porte à croire que la capacité prédictive du contrôleur lui permettrait de réajuster ses prédictions futures en tenant compte des évènements au temps présent et de sa dynamique. En effet, lors du calcul de la commande optimale, le contrôleur peut prendre en compte l'état actuel du quadrirotor et calculer une trajectoire de correction pour rejoindre celle de référence si le quadrirotor en a dévié à cause de perturbations. Par exemple, le contrôleur prédictive présenté dans [30] réajuste sa trajectoire lorsqu'il est poussé d'un côté.

Comparativement au MPC, le ILQR est moins lourd quant à la puissance de calcul, mais il ne permet pas de tenir compte des contraintes sur les commandes ou sur les états à ne pas franchir. Par exemple, avec un MPC il est possible de spécifier qu'un angle de tangage de 3° ne doit pas être dépassé durant la trajectoire et les commandes effectués sur un horizon de temps vont être optimisé avec cette contrainte contrairement au fait de seulement bloquer les valeurs maximales envoyées aux actionneurs. Pour atténuer le problème de calcul du MPC, des chercheurs l'ont remplacé par un RNN (Réseaux de Neurones Récurrents) [53]. Une problème d'optimisation en temps réel n'a plus à être résolu et le temps de calcul est réduite [54].

Pour améliorer l'erreur en suivi de trajectoire, un piste de solution est d'ajouter à l'extérieur de la boucle de contrôle un DNN (Réseaux de Neurones Profonds) [55]. En étudiant la réaction du modèle réel du quadrirotor, les effets dynamiques non modélisés peuvent être pris en compte dans la loi de contrôle. L'erreur en suivi de trajectoire est réduite d'environ 50%. En fait, cette approche est reliée au contrôle adaptif qui est robuste vis-à-vis des variations dans les paramètres du modèle du robot comme son inertie. D'autres lois de contrôle adaptatives pourraient être envisagées.

Un autre aspect majeur affectant la performance du quadrirotor est sa fréquence de contrôle. Lors de trajectoires agressives, l'état du quadrirotor change rapidement d'où la nécessité de mettre à jour les commandes désirées à haute fréquence. La méthodologie expérimentale présentée dans ce mémoire utilise des fréquences de 100 Hz dans la boucle de contrôle en position. La fréquence utilisée pour le contrôle de l'inclinaison des moteurs est de 50 Hz. Or, dans [40] il a été démontré que l'erreur moyenne en position passe de 0.042 à 0.003m lorsque la fréquence de contrôle des servomoteurs est augmenté de 55 à 500 Hz. Ainsi, garder la fréquence de contrôle de la boucle externe à 100 Hz, mais en ayant une estimation d'état de la position et des vitesses à 100Hz

contrairement à 45 Hz et envoyer des commandes au servomoteur à 100 Hz pourrait être suffisant pour voir une différence dans la performance en suivi de trajectoire. Cependant, cela implique de changer de contrôleur de vol puisqu'il est limité quant à la fréquence de contrôle des actionneurs secondaires et d'augmenter la fréquence de l'estimation d'état, discuté dans la section 6.2.5.

6.2.4 Planification de la trajectoire

La méthode employée pour planifier la trajectoire à suivre est basé sur la propriété du système dynamique d'être différentiablement plat. Or, les trajectoires que le quadrirotor peut suivre sont limités à des polynomiales avec cette méthode. Avec un actionneur supplémentaire, le quadrirotor à moteurs inclinables peut accélérer presque qu'instantanément dans l'axe où les moteurs sont inclinables et par conséquent suivre des trajectoires non continues. Préférablement, la planification de trajectoire devrait tenir compte de cette particularité pour exploiter pleinement l'agressivité dont le véhicule est capable.

Aussi, la trajectoire planifiée est envoyée au contrôleur en début de manœuvre. Le quadrirotor devrait plutôt planifier sa trajectoire en temps réel en prenant en compte un vecteur d'état mis à jour pour atteindre les mêmes points d'intérêts. Ainsi, s'il dévie de sa trajectoire, il pourrait recalculer une trajectoire acceptable respectant les mêmes points d'intérêts.

Puis, le quadrirotor développé dans ce mémoire devrait également être mis à l'essai avec des points d'intérêt planifier sur tout son vecteur d'état. Présentement, seulement les positions cartésiennes et leurs dérivées peuvent être imposées comme points d'intérêts. Pour des manœuvres plus agiles, il serait pertinent de concevoir un algorithme de planification permettant d'imposer des angles à respecter aux points d'intérêts. Passer à travers des fenêtres encore plus étroites requérant le quadrirotor d'adopter des orientations non conventionnelles serait alors possible. Des méthodes d'optimisation peuvent être utilisées pour trouver ces trajectoires de référence à donner au contrôleur.

Toutefois, puisque seulement des points précis sur la trajectoire doivent être imposés, il serait pertinent de combiner la planification de trajectoire et le contrôle dans un cadre uniifié, comme dans [30]. Cette approche a l'avantage de permettre au quadrirotor de ne pas respecter pleinement la contrainte au point d'intérêt si le contrôle nécessaire pour y arriver est irréaliste. En plus, cette

méthode tiendrait compte de la dynamique complète du quadrirotor à rotors inclinables lorsque la trajectoire est planifiée et de l'optimiser en tenant compte des capacités dynamiques du robot.

6.2.5 Estimation d'état

L'estimation d'état est d'un part basé sur le système de capture BTS pour les positions et vitesses, puis de l'IMU embarqué sur le quadrirotor pour les angles. En fait, les angles sont fusionnés avec ceux mesurés par le système de capture pour plus de robustesse et pour les avoir à 100Hz plutôt qu'à 45Hz.

Les positions et vitesses sont fournis à trop basse fréquence pour du contrôle agressif précis. Une possibilité aurait été de fusionnée les positions avec les accélérations de l'IMU qui seraient intégrés pour avoir des mesures à 100Hz également. Les données du BTS servent alors à réajuster les positions lorsque les positions trouvées uniquement par l'accéléromètre devient excessivement ce qui est commun avec cette approche à cause de l'accumulation des erreurs causés par le bruit. Cependant, les vibrations excessives causent trop de bruit dans les mesures de l'accéléromètre ce qui rend ces mesures inutilisables. Donc, la structure mécanique doit préalablement être renforcée pour atteindre un seuil acceptable de vibrations, voir la section 5.3. Une autre possibilité serait tout simplement de fournir les mesures à 100 Hz avec le système BTS. Les principales limitations observées à ce niveau sont la communication par radio qui est de 60 Hz au plus et la communication UDP avec les caméras BTS qui sont le facteur limitant à 45 Hz. Cependant, nous croyons qu'il est possible d'atteindre une fréquence de 100 Hz puisque théoriquement les caméras BTS peuvent fournir les mesures de position des capteurs à plus de 400 Hz. Aussi, le module radio Laird [56] utilisé semble limité à 60 Hz comme suggéré dans [19], mais certains chercheurs sont parvenus à atteindre 100 Hz avec un module radio ZigBee [18]. Des trajectoires agressives ont également été accomplis dans [57] avec un module radio Crazyradio PA et un système de capture Vicon permettant d'avoir une estimation d'état à 100 Hz. Dans [18], la performance en maintien de position se dégrade seulement en dessous de 35 Hz et il est suggéré que des manœuvres agressives présentés peuvent être réalisés à plus faible fréquence, mais avec plus d'erreur en suivi de trajectoire. L'augmentation de la fréquence devrait donc seulement se faire si la précision du suivi doit être amélioré et un compromis doit être fait avec le coût des composantes et la charge de travail.

Finalement, un aspect important de l'estimation d'état est de compenser la latence totale à partir de la prise de mesure jusqu'à l'envoie des commandes. En effet, pour une latence de 50ms et une vitesse en cours de trajectoire de 3m/s, le quadrirotor pourrait avoir atteint une position différente de plus de 15cm par rapport à l'estimation d'état trouvé il y a 50ms. Dans [19], les états désuets sont intégrés sur la durée de temps de la latence. L'estimation d'état à jour du véhicule est alors utilisée dans la loi de contrôle. Une méthode plus robuste présentée dans [58] et utilisée dans [14] est de concevoir un filtre de Kalman étendu permettant de tenir compte de la dynamique du système et également de fusionner les capteurs disponibles pour les mesures. Ce fût la méthode employée pour les angles donnés au quadrirotor, mais pour les positions et les vitesses, il n'a pas été possible de fusionner les mesures à cause des vibrations comme discuté plus haut.

6.2.6 Autonomie

Les environnements inconnus tels que des mines ou des zones de sauvetage ne sont pas équipées de système de capture externe. L'estimation d'état doit provenir exclusivement de ressources disponibles seulement à bord du quadrirotor. La suite de capteurs minimale pour une localisation autonome se compose d'une caméra monoculaire et d'un IMU qui sont légers, peu coûteux et largement disponibles. Ainsi, plusieurs méthodes existent, mais l'état de l'art évolue présentement vers des algorithmes d'odométrie visuelle-inertielle monoculaire. L'utilisation d'un GPS n'est pas nécessaire ce qui est pratique dans des endroits isolés aux satellites comme l'intérieur d'un immeuble ou une mine. À l'aide d'une seule caméra et d'un IMU, l'état complet du quadrirotor peut être estimé. Ces algorithmes ont été testés sur un UpBoard, le même ordinateur de bord utilisé dans ce projet, et les erreurs moyennes se situent entre 2.02m et 0.05m par rapport à la position réelle [59]. Avec un ordinateur plus puissant, le *Qualcomm Snapdragon Flight*, des trajectoires agressives sont effectués avec des erreurs en localisation inférieur à 9cm comparativement à l'état calculé par un système de capture [25].

CHAPITRE 7 CONCLUSION ET RECOMMANDATIONS

Ce travail visait à développer un quadrirotor capable de suivre des trajectoires agressives dans des environnements encombrés. Le but principal était de s'affranchir des principales limitations du quadrirotor conventionnel pour exploiter pleinement son potentiel d'agilité. D'une part, le problème de sous-actuation a été atténué en rajoutant un actionneur dans un axe ce qui permet d'avoir des moteurs inclinables. Le quadrirotor devient alors complètement actionné dans un axe et reste sous-actionnée dans l'autre où les moteurs ne peuvent pas être inclinés. Cette modification au design conventionnel permet de générer des accélérations presqu'instantanément dans la direction voulue sans être limité par le temps de réponse de l'orientation commandée. D'autre part, une loi de contrôle prédictive, l'iLQR, a été développée sur le modèle dynamique modifié du quadrirotor. Les manœuvres agressives peuvent être mieux anticipées en tenant compte des implications futures du choix de commandes au temps présent et de la dynamique du quadrirotor pour prédire sa réaction dépendamment des commandes possibles. Le design et la loi de contrôle ont été testés en simulation où des manœuvres agressives avec des accélérations dépassant 10m/s^2 ont été effectuées. Il a été mis en évidence que les moteurs inclinables permettent de mieux négocier les changements de directions brusques. La modification au design impacte plus la performance du véhicule sur ces trajectoires que le contrôleur prédictif, mais celui-ci permet également de réduire l'erreur en suivi de trajectoire. Puis, un prototype expérimental du design proposé a été fabriqué en utilisant des pièces électroniques largement disponibles dans le domaine des UAVs, du prototypage rapide et des plaques en fibre de carbone machinées. Il a été mis à l'essai sur une trajectoire demandant d'éviter un obstacle à haute vitesse. L'estimation de l'état du véhicule a été effectuée en utilisant un système de capture BTS et l'IMU du contrôleur de vol. Comme en simulation, le design avec rotors inclinables et un contrôleur ILQR performe mieux que les autres. Sa capacité à freiner et accélérer rapidement sont mises en lumières. Néanmoins, l'erreur absolue par rapport aux positions et vitesses désirées est plus grande qu'en simulation ce qui est explicable à cause des limitations du prototype réel.

Les principales limitations affectant la performance du quadrirotor développé durant ce projet sont la fréquence de contrôle peu élevée comparativement à ce qui est accomplie dans la littérature pour des trajectoires agressives et les vibrations dues à la faiblesse de la conception mécanique. Ces

limitations devraient être adressées en premier lieu. Ensuite, un planificateur de trajectoire incorporant la dynamique de ce design de quadrirotor en particulier devrait être envisagé. Puis, pour développer un produit fini capable de circuler à haute vitesse dans des environnements encombrés comme des mines ou des immeubles en ruine, l'estimation d'état du quadrirotor devrait être effectuée à bord du véhicule avec les ressources disponibles. Une plateforme complètement autonome pourra alors être vendue commercialement.

RÉFÉRENCES

- [1] B. J. Emran et H. Najjaran, "A review of quadrotor: An underactuated mechanical system," vol. 46, p. 165-180, 2018, 165.
- [2] R. Clarke, "Understanding the drone epidemic," vol. 30, n°. 3, p. 230-246, 2014, 230.
- [3] S. Tang et V. Kumar, "Autonomous flight," vol. 1, p. 29-52, 2018.
- [4] P. Gohl *et al.*, "Towards autonomous mine inspection," communication présentée à Proceedings of the 2014 3rd International Conference on Applied Robotics for the Power Industry, 2014, p. 1-6.
- [5] D. Falanga, S. Kim et D. Scaramuzza, "How fast is too fast? the role of perception latency in high-speed sense and avoid," vol. 4, n°. 2, p. 1884-1891, 2019.
- [6] S. Karaman et E. Frazzoli, "High-speed flight in an ergodic forest," communication présentée à 2012 IEEE International Conference on Robotics and Automation, 2012, p. 2899-2906.
- [7] L. V. Santana *et al.*, "A trajectory tracking and 3d positioning controller for the ar. drone quadrotor," communication présentée à 2014 international conference on unmanned aircraft systems (ICUAS), 2014, p. 756-767.
- [8] R. Mahony, V. Kumar et P. Corke, "Multirotor aerial vehicles: Modeling, estimation, and control of quadrotor," vol. 19, n°. 3, p. 20-32, 2012.
- [9] P. E. Pounds, D. R. Bersak et A. M. Dollar, "Grasping from the air: Hovering capture and load stability," communication présentée à 2011 IEEE international conference on robotics and automation, 2011, p. 2491-2498.
- [10] L. Gentili, R. Naldi et L. Marconi, "Modeling and control of VTOL UAVs interacting with the environment," communication présentée à 2008 47th IEEE Conference on Decision and Control, 2008, p. 1231-1236.
- [11] K. Kondak *et al.*, "Closed-loop behavior of an autonomous helicopter equipped with a robotic arm for aerial manipulation tasks," vol. 10, n°. 2, p. 145, 2013.
- [12] M. Ryll *et al.*, "6D physical interaction with a fully actuated aerial robot," communication présentée à 2017 IEEE International Conference on Robotics and Automation (ICRA), 2017, p. 5190-5195.
- [13] P. Foehn et D. Scaramuzza, "Onboard state dependent lqr for agile quadrotors," communication présentée à 2018 IEEE International Conference on Robotics and Automation (ICRA), 2018, p. 6566-6572.
- [14] M. Kamel, M. Burri et R. Siegwart, "Linear vs nonlinear mpc for trajectory tracking applied to rotary wing micro aerial vehicles," vol. 50, n°. 1, p. 3463-3469, 2017.
- [15] M. Blösch *et al.*, "Vision based MAV navigation in unknown and unstructured environments," communication présentée à 2010 IEEE International Conference on Robotics and Automation, 2010, p. 21-28.
- [16] D. Mellinger et V. Kumar, "Minimum snap trajectory generation and control for quadrotors," communication présentée à 2011 IEEE International Conference on Robotics and Automation, 2011, p. 2520-2525.

- [17] M. Burri *et al.*, "A framework for maximum likelihood parameter identification applied on MAVs," vol. 35, n°. 1, p. 5-22, 2018.
- [18] D. Mellinger, N. Michael et V. Kumar, "Trajectory generation and control for precise aggressive maneuvers with quadrotors," vol. 31, n°. 5, p. 664-674, 2012.
- [19] M. Faessler, A. Franchi et D. Scaramuzza, "Differential flatness of quadrotor dynamics subject to rotor drag for accurate tracking of high-speed trajectories," vol. 3, n°. 2, p. 620-626, 2017.
- [20] J. A. Rossiter, *Model-based predictive control: a practical approach*: CRC press, 2003.
- [21] L. Li, L. Sun et J. Jin, "Survey of advances in control algorithms of quadrotor unmanned aerial vehicle," communication présentée à 2015 IEEE 16th International Conference on Communication Technology (ICCT), 2015, p. 107-111.
- [22] M. Islam et M. Okasha, "A Comparative Study of PD, LQR and MPC on Quadrotor Using Quaternion Approach," communication présentée à 2019 7th International Conference on Mechatronics Engineering (ICOM), 2019, p. 1-6.
- [23] E. C. Suicmez et A. T. Kutay, "Optimal path tracking control of a quadrotor UAV," communication présentée à 2014 International Conference on Unmanned Aircraft Systems (ICUAS), 2014, p. 115-125.
- [24] E. Tal et S. Karaman, "Accurate tracking of aggressive quadrotor trajectories using incremental nonlinear dynamic inversion and differential flatness," communication présentée à 2018 IEEE Conference on Decision and Control (CDC), 2018, p. 4282-4288.
- [25] G. Loianno *et al.*, "Estimation, control, and planning for aggressive flight with a small quadrotor with a single camera and IMU," vol. 2, n°. 2, p. 404-411, 2016.
- [26] D. Falanga *et al.*, "Aggressive quadrotor flight through narrow gaps with onboard sensing and computing using active vision," communication présentée à 2017 IEEE international conference on robotics and automation (ICRA), 2017, p. 5774-5781.
- [27] C. Richter, A. Bry et N. Roy, "Polynomial trajectory planning for aggressive quadrotor flight in dense indoor environments," dans *Robotics Research*: Springer, 2016, p. 649-666.
- [28] J.-J. E. Slotine et W. Li, *Applied nonlinear control* vol. 199: Prentice hall Englewood Cliffs, NJ, 1991.
- [29] M. Faessler *et al.*, "Automatic re-initialization and failure recovery for aggressive flight with a monocular vision-based quadrotor," communication présentée à 2015 IEEE International Conference on Robotics and Automation (ICRA), 2015, p. 1722-1729.
- [30] M. Neunert *et al.*, "Fast nonlinear model predictive control for unified trajectory optimization and tracking," communication présentée à 2016 IEEE international conference on robotics and automation (ICRA), 2016, p. 1398-1404.
- [31] D. Brescianini et R. D'Andrea, "Design, modeling and control of an omni-directional aerial vehicle," communication présentée à 2016 IEEE international conference on robotics and automation (ICRA), 2016, p. 3261-3266.
- [32] A. Bin Junaid *et al.*, "Design and implementation of a dual-axis tilting quadcopter," vol. 7, n°. 4, p. 65, 2018.
- [33] A. Alkamachi et E. Ercelebi, "Modelling and control of H-shaped racing quadcopter with tilting propellers," vol. 15, n°. 2, p. 201-215, 2017.

- [34] M. Kamel *et al.*, "Voliro: An omnidirectional hexacopter with tilttable rotors," 2018.
- [35] A. Oosedo *et al.*, "Large attitude change flight of a quad tilt rotor unmanned aerial vehicle," vol. 30, n°. 5, p. 326-337, 2016.
- [36] S. Rajappa *et al.*, "Modeling, control and design optimization for a fully-actuated hexarotor aerial vehicle with tilted propellers," communication présentée à 2015 IEEE International Conference on Robotics and Automation (ICRA), 2015, p. 4006-4013.
- [37] R. Kumar, "Position, Attitude, and Fault-Tolerant Control of Tilting-Rotor Quadcopter," University of Cincinnati, 2017.
- [38] R. Kumar *et al.*, "Position and attitude control by rotor tilt and rotor speed synchronization for single axis tilting-rotor quadcopter," communication présentée à ASME 2017 Dynamic Systems and Control Conference, 2017.
- [39] R. I. Popescu, "Design et développement d'un quadrirotor joueur de tennis de table avec des hélices inclinables," Génie mécanique, Polytechnique Montréal, Montréal, 2020. [En ligne]. Disponible: <https://publications.polymtl.ca/>
- [40] M. Ryall, H. H. Bülthoff et P. R. Giordano, "A novel overactuated quadrotor unmanned aerial vehicle: Modeling, control, and experimental validation," vol. 23, n°. 2, p. 540-556, 2014.
- [41] A. Laub, "A Schur method for solving algebraic Riccati equations," vol. 24, n°. 6, p. 913-921, 1979.
- [42] G. G. Slabaugh, "Computing Euler angles from a rotation matrix," vol. 6, n°. 2000, p. 39-63, 1999.
- [43] Log Analysis using Flight Review. [En ligne]. Disponible: https://docs.px4.io/v1.9.0/en/log/flight_review.html
- [44] A. Mohebbi, S. Achiche et L. J. T. o. t. C. S. f. M. E. Baron, "Design of a vision guided mechatronic quadrotor system using design for control methodology," vol. 40, n°. 2, p. 201-219, 2016.
- [45] A. Mohebbi, S. Achiche et L. J. E. A. o. A. I. Baron, "Integrated and concurrent detailed design of a mechatronic quadrotor system using a fuzzy-based particle swarm optimization," vol. 82, p. 192-206, 2019.
- [46] A. Mohebbi, S. Achiche et L. Baron, "Integrated Design of A Vision-Guided Quadrotor UAV: A Mechatronics Approach," communication présentée à CCToMM Symposium on Mechanisms, Machines, and Mechatronics, 2015.
- [47] A. Mohebbi *et al.*, "Integrated structure-control design optimization of an unmanned quadrotor helicopter (UGH) for object grasping and manipulation," communication présentée à DS 87-4 Proceedings of the 21st International Conference on Engineering Design (ICED 17) Vol 4: Design Methods and Tools, Vancouver, Canada, 21-25.08. 2017, 2017, p. 623-632.
- [48] P.-J. Bristeau *et al.*, "The role of propeller aerodynamics in the model of a quadrotor UAV," communication présentée à 2009 European Control Conference (ECC), 2009, p. 683-688.
- [49] P. Martin et E. Salaün, "The true role of accelerometer feedback in quadrotor control," communication présentée à 2010 IEEE International Conference on Robotics and Automation, 2010, p. 1623-1629.
- [50] R. Mahony, V. Kumar et P. Corke, "Multirotor aerial vehicles: Modeling, estimation, and control of quadrotor," vol. 19, n°. 3, p. 20-32, 2012.

- [51] C. Papachristos, K. Alexis et A. Tzes, "Dual-authority thrust–vectoring of a tri–tiltrotor employing model predictive control," vol. 81, n°. 3-4, p. 471-504, 2016.
- [52] J. E. Normey-Rico, *Control of dead-time processes*: Springer Science & Business Media, 2007.
- [53] Y. Pan et J. Wang, "Model predictive control of unknown nonlinear dynamical systems based on recurrent neural networks," vol. 59, n°. 8, p. 3089-3101, 2011.
- [54] S. S. P. Kumar *et al.*, "A deep learning architecture for predictive control," vol. 51, n°. 18, p. 512-517, 2018.
- [55] Q. Li *et al.*, "Deep neural networks for improved, impromptu trajectory tracking of quadrotors," communication présentée à 2017 IEEE International Conference on Robotics and Automation (ICRA), 2017, p. 5183-5189.
- [56] (2020) Laird Radio Module. [En ligne]. Disponible: <https://www.lairdconnect.com/wireless-modules/ramp-ism-modules/proprietary-24-ghz-zigbee-802154-modules/rm024-series>
- [57] D. Conover, "Trajectory Generation for a Quadrotor Unmanned Aerial Vehicle," École Polytechnique de Montréal, 2018.
- [58] S. Lynen *et al.*, "A robust and modular multi-sensor fusion approach applied to mav navigation," communication présentée à 2013 IEEE/RSJ international conference on intelligent robots and systems, 2013, p. 3923-3929.
- [59] J. Delmerico et D. Scaramuzza, "A benchmark comparison of monocular visual-inertial odometry algorithms for flying robots," communication présentée à 2018 IEEE International Conference on Robotics and Automation (ICRA), 2018, p. 2502-2509.
- [60] R. Clarke, "Understanding the drone epidemic," *Computer Law & Security Review*, vol. 30, no. 3, pp. 230-246, 2014.
- [61] B. J. Emran and H. Najjaran, "A review of quadrotor: An underactuated mechanical system," *Annual Reviews in Control*, vol. 46, pp. 165-180, 2018.
- [62] P. Gohl *et al.*, "Towards autonomous mine inspection," in *Proceedings of the 2014 3rd International Conference on Applied Robotics for the Power Industry*, 2014, pp. 1-6: IEEE.
- [63] D. Falanga, S. Kim, and D. Scaramuzza, "How fast is too fast? the role of perception latency in high-speed sense and avoid," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 1884-1891, 2019.
- [64] S. Karaman and E. Frazzoli, "High-speed flight in an ergodic forest," in *2012 IEEE International Conference on Robotics and Automation*, 2012, pp. 2899-2906: IEEE.
- [65] E. C. Suicmez and A. T. Kutay, "Optimal path tracking control of a quadrotor UAV," in *2014 International Conference on Unmanned Aircraft Systems (ICUAS)*, 2014, pp. 115-125: IEEE.
- [66] R. Mahony, V. Kumar, and P. Corke, "Multirotor aerial vehicles: Modeling, estimation, and control of quadrotor," *IEEE robotics & automation magazine*, vol. 19, no. 3, pp. 20-32, 2012.
- [67] E. Tal and S. Karaman, "Accurate tracking of aggressive quadrotor trajectories using incremental nonlinear dynamic inversion and differential flatness," in *2018 IEEE Conference on Decision and Control (CDC)*, 2018, pp. 4282-4288: IEEE.
- [68] G. Loianno, C. Brunner, G. McGrath, and V. Kumar, "Estimation, control, and planning for aggressive flight with a small quadrotor with a single camera and IMU," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 404-411, 2016.

- [69] D. Mellinger and V. Kumar, "Minimum snap trajectory generation and control for quadrotors," in *2011 IEEE International Conference on Robotics and Automation*, 2011, pp. 2520-2525: IEEE.
- [70] D. Falanga, E. Mueggler, M. Faessler, and D. Scaramuzza, "Aggressive quadrotor flight through narrow gaps with onboard sensing and computing using active vision," in *2017 IEEE international conference on robotics and automation (ICRA)*, 2017, pp. 5774-5781: IEEE.
- [71] C. Richter, A. Bry, and N. Roy, "Polynomial trajectory planning for aggressive quadrotor flight in dense indoor environments," in *Robotics Research*: Springer, 2016, pp. 649-666.
- [72] M. Kamel, M. Burri, and R. Siegwart, "Linear vs nonlinear mpc for trajectory tracking applied to rotary wing micro aerial vehicles," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 3463-3469, 2017.
- [73] M. Blösch, S. Weiss, D. Scaramuzza, and R. Siegwart, "Vision based MAV navigation in unknown and unstructured environments," in *2010 IEEE International Conference on Robotics and Automation*, 2010, pp. 21-28: IEEE.
- [74] R. Kumar, A. Nemati, M. Kumar, K. Cohen, and F. Cazaurang, "Position and attitude control by rotor tilt and rotor speed synchronization for single axis tilting-rotor quadcopter," in *ASME 2017 Dynamic Systems and Control Conference*, 2017: American Society of Mechanical Engineers Digital Collection.
- [75] A. Alkamachi and E. Ercelebi, "Modelling and control of H-shaped racing quadcopter with tilting propellers," *Facta Universitatis, Series: Mechanical Engineering*, vol. 15, no. 2, pp. 201-215, 2017.
- [76] A. Bin Junaid, A. Diaz De Cerio Sanchez, J. Betancor Bosch, N. Vitzilaios, and Y. Zweiri, "Design and implementation of a dual-axis tilting quadcopter," *Robotics*, vol. 7, no. 4, p. 65, 2018.
- [77] A. Zulu and S. J. a. p. a. John, "A review of control algorithms for autonomous quadrotors," 2016.
- [78] L. Li, L. Sun, and J. Jin, "Survey of advances in control algorithms of quadrotor unmanned aerial vehicle," in *2015 IEEE 16th International Conference on Communication Technology (ICCT)*, 2015, pp. 107-111: IEEE.
- [79] D. Mellinger, N. Michael, and V. Kumar, "Trajectory generation and control for precise aggressive maneuvers with quadrotors," *The International Journal of Robotics Research*, vol. 31, no. 5, pp. 664-674, 2012.
- [80] M. Faessler, A. Franchi, and D. Scaramuzza, "Differential flatness of quadrotor dynamics subject to rotor drag for accurate tracking of high-speed trajectories," *IEEE Robotics and Automation Letters*, vol. 3, no. 2, pp. 620-626, 2017.
- [81] A. Laub, "A Schur method for solving algebraic Riccati equations," *IEEE Transactions on automatic control*, vol. 24, no. 6, pp. 913-921, 1979.
- [82] Y. Tassa, N. Mansard, and E. Todorov, "Control-limited differential dynamic programming," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, 2014, pp. 1168-1175: IEEE.
- [83] E. Todorov and W. Li, "A generalized iterative LQG method for locally-optimal feedback control of constrained nonlinear stochastic systems," in *Proceedings of the 2005, American Control Conference, 2005.*, 2005, pp. 300-306: IEEE.
- [84] P. Geoffroy, N. Mansard, M. Raison, S. Achiche, and E. Todorov, "From inverse kinematics to optimal control," in *Advances in Robot Kinematics*: Springer, 2014, pp. 409-418.

- [85] P. Geoffroy, O. Bordron, N. Mansard, M. Raison, O. Stasse, and T. Bretl, "A two-stage suboptimal approximation for variable compliance and torque control," in *2014 European Control Conference (ECC)*, 2014, pp. 1151-1157: IEEE.
- [86] W. Li and E. Todorov, "Iterative linear quadratic regulator design for nonlinear biological movement systems," in *ICINCO (1)*, 2004, pp. 222-229.
- [87] (26/07/2020). *UP board specifications* [Web Page]. Available: <https://up-board.org/up/specifications/>
- [88] S. Lynen, M. W. Achtelik, S. Weiss, M. Chli, and R. Siegwart, "A robust and modular multi-sensor fusion approach applied to mav navigation," in *2013 IEEE/RSJ international conference on intelligent robots and systems*, 2013, pp. 3923-3929: IEEE.
- [89] (26/07/2020). *BTS Bioengineering* [Web Page]. Available: <https://www.btsbioengineering.com/products/smart-dx/>
- [90] (26/07/2020). *Ubuntu* [Web Page]. Available: <https://ubuntu.com/>
- [91] (26/07/2020). *Robot Operating System (ROS)* [Web Page]. Available: <https://www.ros.org/>