

الجمهورية الجزائرية الديمقراطية الشعبية

RÉPUBLIQUE ALGÉRIENNE DÉMOCRATIQUE ET POPULAIRE

وزارة

العلم العالي والبحث العلمي

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

جامعة محمد البشير الإبراهيمي - برج بوعريريج

Université de Mohamed El-Bachir El-Ibrahimi - Bordj Bou Arreridj

Faculté des Sciences et de la Technologie

Département d'Electromécanique

MÉMOIRE

Présenté pour l'obtention du **diplôme** de **MASTER**

En : Électromécanique

Spécialité :Électromécanique

Par : - Gherab Walid

- Guessabi Salah

Sujet

Commande avancée d'un véhicule aérien autonome (Quadcopter UAV)

Soutenu publiquement, le 29/06 /2026 , devant le jury composé de :

Mme.Bouguerra Zahira
M.Benheniche Abdelhak
Mme.Meradi Dounia

MCB
MCA
MCB

Univ-BBA
Univ-BBA
Univ-BBA

Président
Examineur
Encadrant

Remerciements

Je tiens tout d'abord à remercier Allah, le Tout-Puissant, de m'avoir accordé la santé, la force, la patience et la volonté nécessaires pour mener à bien ce travail.

J'adresse mes plus sincères remerciements à mon encadreuse, **Mme Dounia Meradi**, pour son accompagnement, sa disponibilité, ses précieux conseils ainsi que la confiance qu'elle m'a accordée tout au long de la réalisation de ce mémoire.

Je tiens également à exprimer ma profonde gratitude aux membres du jury qui ont accepté d'évaluer ce travail et de participer à sa soutenance.

Mes remerciements s'adressent aussi à l'ensemble des enseignants de l'Université de Bordj Bou Arréridj pour la qualité de leur enseignement et pour les connaissances qu'ils m'ont transmises durant mon cursus universitaire.

Dédicace 1

Je dédie humblement ce mémoire, fruit de plusieurs années d'efforts, de persévérance et d'apprentissage.

À mes très chers parents,

Pour leur amour inconditionnel, leurs sacrifices, leurs encouragements constants et leur soutien indéfectible tout au long de mon parcours. Aucun mot ne saurait exprimer ma profonde gratitude envers eux.

À ma famille,

Pour son affection, sa confiance et sa présence qui ont toujours été une source de motivation et de réconfort.

À mes chers amis,

Pour leur amitié sincère, leur soutien moral, leur esprit d'entraide et tous les moments inoubliables partagés durant ces années d'études. Leur présence a rendu ce parcours plus agréable et plus enrichissant.

À toutes les personnes qui ont contribué, de près ou de loin, à la réalisation de ce travail.

Walid Gherab

Dédicace 2

Je dédie ce modeste travail,

À mes très chers parents,

Pour leur amour infini, leur soutien inconditionnel, leurs sacrifices et leurs encouragements tout au long de mon parcours. Que Dieu les protège et les récompense pour tout ce qu'ils ont fait pour moi.

À mes frères et sœurs,

Pour leur affection, leur présence et leur soutien moral.

À mes chers amis : Yahia, Yaacoub, Bayazid et Anis,

En souvenir des moments de partage, d'entraide et d'amitié sincère qui ont marqué ces années d'études.

À tous ceux qui m'ont soutenu de près ou de loin dans la réalisation de ce travail.

Salleh Guessabi

Table des matières

Remerciements	i
Dédicace 1	ii
Dédicace 2	iii
Liste des figures	viii
Liste des tableaux	ix
Résumé	1
Introduction Générale	2
1 Modélisation Mathématique et Dynamique du Quadrotor	4
1.1 Généralités sur les systèmes VTOL et Quadrotors	5
1.1.1 Principes de vol et architecture matérielle d'un quadrotor	5
1.1.2 Défis de commande : système sous-actionné et fortement non linéaire	7
1.2 Cinématique Spatiale et Représentation de l'Attitude	7
1.2.1 Les repères d'étude : Repère inertiel (Terre) et Repère mobile (Corps)	8
1.2.2 Les limites des angles d'Euler et le phénomène de Blocage de Cardan (Gimbal Lock)	9
1.2.3 Algèbre des Quaternions unitaires : Propriétés et opérations fon- damentales	10
1.2.4 Modélisation de la cinématique d'attitude par quaternions . . .	12
1.3 Modélisation Dynamique par l'Approche de Newton-Euler	12
1.3.1 Modélisation des forces appliquées (Gravité et Poussée aérody- namique)	13
1.3.2 Modélisation des moments appliqués (Couples de contrôle et ef- fets gyroscopiques)	14
1.3.3 Justification et formulation du modèle d'état non linéaire complet	15

2	Génération de Trajectoire et Conception du Contrôleur NMPC	19
2.1	Planification de Trajectoire par Différentielle Plate	20
2.1.1	Concept mathématique de la différentielle plate appliquée au quadrotor	20
2.1.2	Choix des variables plates (Position $\mathbf{p}$ et angle de lacet ψ) . . .	21
2.1.3	Déduction algébrique des états de référence idéaux ($\mathbf{x}_{ref}$)	21
2.1.4	Calcul des commandes nominales anticipées (Feedforward $\mathbf{u}_{ref}$)	24
2.2	Formulation du Problème de Contrôle Optimal (OCP)	24
2.2.1	Principes fondamentaux de la Commande Prédictive Non Linéaire (NMPC) en horizon glissant	24
2.2.2	Évaluation de l'erreur d'attitude sur SO(3) via l'Application Logarithmique (Log Map)	25
2.2.3	Conception de la fonction de coût quadratique	28
2.3	Intégration des Contraintes du Système	29
2.3.1	Contrainte de la dynamique interne (Discrétisation par RK4) . .	29
2.3.2	Contraintes d'inégalité des actionneurs (Limites des efforts physiques)	30
2.3.3	Contrainte géométrique de normalisation stricte du quaternion .	30
2.4	Modélisation Mathématique pour l'Évitement d'Obstacle	31
2.4.1	Représentation géométrique d'un obstacle sphérique statique . .	31
2.4.2	Formulation de la contrainte d'inégalité de distance euclidienne	31
2.5	Formulation Globale du Problème de Contrôle Optimal (OCP)	33
3	Simulation Numérique et Analyse des Performances	35
3.1	Environnement de Résolution Numérique	36
3.1.1	Le framework logiciel CasADi pour la transcription de l'OCP . .	36
3.1.2	Le solveur IPOPT basé sur les méthodes de points intérieurs . .	36
3.2	Configuration de la Simulation	37
3.2.1	Paramètres physiques nominaux du quadrotor (Masse m_0 , Inertie J_0)	37
3.2.2	Réglages du NMPC	37
3.2.3	Présentation des profils de trajectoires cibles	38
3.3	Scénario 1 : Validation du Suivi de Trajectoire en Conditions Nominale	40
3.3.1	Analyse du comportement spatial (Trajectoire 3D)	40
3.3.2	Évolution temporelle des variables cinématiques et d'état	40
3.3.3	Évolution de l'erreur d'attitude (Log Map) et stabilité des quaternions	40
3.3.4	Profil des signaux de commande générés	43
3.4	Scénario 2 : Analyse de Robustesse face aux Incertitudes et Perturbations	43

3.4.1	Introduction d'une dynamique non modélisée ($t \in [5, 15]$ s) . . .	44
3.4.2	Perturbation externe : Rafale de vent latéral ($t \in [20, 25]$ s) . . .	44
3.4.3	Analyse de la capacité de rejet des perturbations par le contrôleur NMPC	45
3.5	Scénario 3 : Suivi de Trajectoire avec Évitement d'Obstacle	47
3.5.1	Analyse du profil de trajectoire Circulaire avec évitement d'obstacle	48
3.5.2	Analyse du profil de trajectoire Carrée avec évitement d'obstacle	51
3.6	Synthèse Globale des Performances	53
3.6.1	Évaluation de la charge calculatoire (Temps de calcul d'IPOPT)	53
3.6.2	Discussion sur la faisabilité matérielle et le déploiement en temps réel	54
	Conclusion du Chapitre	55
	Conclusion Générale et Perspectives	56
	Bibliographie	57

Table des figures

1.1	Illustration des mouvements du quadrirotor générés par la variation des vitesses angulaires de ses rotors [1].	6
1.2	Géométrie générale du quadrotor et définition des repères inertiel et mobile [1]	8
1.3	Représentation des rotations successives $Z - Y' - X''$ et changement de repères par les angles d'Euler [1].	9
1.4	Mécanisme aérodynamique de génération du moment de tangage par variation différentielle de vitesse (Source : [1]).	14
2.1	Représentation géométrique des sorties plates (position 3D et angle de lacet) du quadrotor le long d'une trajectoire spatiale.	22
2.2	Architecture générale du NMPC.	25
2.3	Illustration du principe de la commande prédictive : glissement de l'horizon d'optimisation aux instants successifs k et $k + 1$	26
2.4	Horizon de prédiction et de contrôle.	26
2.5	Principe du problème OCP.	28
2.6	Modélisation géométrique de l'évitement d'obstacle par la méthode des sphères englobantes.	32
3.1	Réponse spatiale tridimensionnelle du quadrotor en conditions nominales (Scénario 1).	41
3.2	Chronogrammes temporels des variables d'état du quadrotor en suivi de trajectoire circulaire (Scénario 1 : cas nominal).	42
3.3	Évolution temporelle de l'erreur d'angle d'attitude α (distance géodésique calculée par Log-Map) en conditions nominales.	43
3.4	Évolution temporelle des entrées de commande du quadrotor : force de poussée globale F_b et moments de couple aérodynamiques $(\tau_\phi, \tau_\theta, \tau_\psi)$ (Scénario 1 : cas nominal).	44
3.5	Trajectoire spatiale 3D du drone sous l'effet de la dynamique non modélisée (5 – 15 s) et de la rafale de vent (20 – 25 s).	45
3.6	Évolution temporelle des variables d'état (positions et quaternions) comparées aux consignes désirées.	46

3.7	Évolution de l'angle d'erreur d'attitude global α (échelle semi-logarithmique) mettant en évidence les deux phases de perturbation.	47
3.8	Signaux de commande optimisés $(F_b, \tau_\phi, \tau_\theta, \tau_\psi)$ générés par le NMPC au cours du scénario perturbé.	48
3.9	Trajectoire spatiale 3D du quadrotor pour le profil Circulaire avec évitement d'obstacle.	49
3.10	Évolution temporelle des positions cartésiennes (x, y, z) pour le profil Circulaire.	49
3.11	Évolution de l'attitude (Quaternions unitaires) pour le profil Circulaire.	50
3.12	Signaux de commande optimisés $(F_b, \boldsymbol{\tau})$ pour le profil Circulaire.	50
3.13	Trajectoire spatiale 3D du quadrotor pour le profil Carré avec évitement d'obstacle.	51
3.14	Évolution couplée des positions cartésiennes (à gauche) et de l'attitude quaternionique (à droite) pour le profil Carré.	52
3.15	Profils des vitesses linéaires (à gauche) et des vitesses angulaires (à droite) illustrant la dynamique transitoire.	52
3.16	Signaux de commande optimisés $(F_b, \boldsymbol{\tau})$ pour le profil Carré sous contrainte d'évitement.	53

Liste des Symboles et Abréviations

Nomenclature du Quadrotor

Symbole	Signification
x, y, z	Coordonnées de position du quadrotor dans le repère inertiel [m]
v_x, v_y, v_z	Vitesses linéaires du quadrotor dans le repère inertiel [m/s]
q_w, q_x, q_y, q_z	Composantes du quaternion représentant l'attitude du quadrotor
p, q, r	Vitesses angulaires dans le repère du corps (Roulis, Tangage, Lacet) [rad/s]
ψ, θ, ϕ	Angles d'Euler (Lacet, Tangage, Roulis) [rad]
m	Masse totale du quadrotor [kg]
g	Accélération de la pesanteur [m/s ²]
T	Poussée totale (Thrust) générée par les moteurs [N]
τ_x, τ_y, τ_z	Couples d'actionnement appliqués sur les axes du corps [N.m]
I_x, I_y, I_z	Moments d'inertie du quadrotor par rapport aux axes du corps [kg.m ²]
X	Vecteur d'état du système (13×1)
U	Vecteur d'entrée/commande du système (4×1)
Y	Vecteur des sorties plates du système

Paramètres de Commande (NMPC)

Abréviations

Symbole	Signification
N	Horizon de prédiction du contrôleur NMPC
dt	Pas d'échantillonnage temporel [s]
T_p	Temps de prédiction global ($N \times dt$) [s]
Q_p	Matrice de pondération des erreurs de position
Q_v	Matrice de pondération des erreurs de vitesse
Q_R	Matrice de pondération des erreurs d'attitude (quaternions)
Q_ω	Matrice de pondération des erreurs de vitesse angulaire
R_u	Matrice de pondération de l'effort de commande (actionneurs)
J	Fonction de coût du problème d'optimisation

Abréviation	Signification
NMPC	Nonlinear Model Predictive Control
RK4	Méthode de Runge-Kutta d'ordre 4
IPOPT	Interior Point Optimizer (Solveur)
CasADi	Computer algebra system for automatic differentiation
UAV	Unmanned Aerial Vehicle (Véhicule aérien sans pilote)

Liste des tableaux

3.1	Paramètres physiques nominaux et limites saturantes du quadrotor . .	37
3.2	Paramètres de réglage et matrices de pondération du contrôleur NMPC	38

الملخص

يقترح هذا البحث نظام تحكم إستشراقي غير خطي (NMPC) لتتبع المسار وتجنب العوائق لطائرة رباعية المروحيات (Quadrotor). تم بناء النموذج الديناميكي وفق نهج نيوتنأويلر باستخدام الكواتيرنيون (Quaternions) لتجنب ظاهرة قفل المحور (Gimbal Lock). كما استُخدم التسطيح التفاضلي (Differential Flatness) لتوليد المسارات المرجعية. تدج خوارزمية NMPC القيود الفيزيائية للنظام، وتقيّم خطأ الاتجاه باستخدام الخريطة اللوغاريتمية (Log Map) على المتنوعة $SO(3)$ ، مع تضمين قيد هندسي لتجنب العوائق الثابتة. أثبتت المحاكاة المبرمجة على MATLAB باستخدام أداتي CasADi و IPOPT دقة في التتبع، ومثانة عالية ضد الاضطرابات، وفعالية كبيرة في تجنب العوائق.

الكلمات المفتاحية : طائرة رباعية المروحيات (Quadrotor)، NMPC، كواتيرنيون (Quaternions)، التسطيح التفاضلي (Differential Flatness)، الخريطة اللوغاريتمية (Log Map)، تجنب العوائق، IPOPT، CasADi.

Résumé

Ce mémoire propose une commande prédictive non linéaire (NMPC) pour le suivi de trajectoire et l'évitement d'obstacles d'un quadrotor. Le modèle dynamique est établi par l'approche de Newton-Euler avec des quaternions unitaires pour éviter le phénomène de Gimbal Lock. La différentielle plate (Differential Flatness) est utilisée pour générer les trajectoires de référence. L'optimisation NMPC intègre les contraintes physiques, évalue l'erreur d'attitude via l'application logarithmique (Log Map) sur la variété $SO(3)$, et inclut une contrainte géométrique pour l'évitement d'obstacles statiques. Implémenté sous MATLAB avec CasADi et IPOPT, le contrôleur démontre en simulation un suivi précis, une forte robustesse aux perturbations et un évitement efficace.

Mots-clés : Quadrotor, NMPC, Quaternions, Differential Flatness, Log Map, Évitement d'obstacles, CasADi, IPOPT.

Abstract

This thesis proposes a Nonlinear Model Predictive Control (NMPC) strategy for trajectory tracking and obstacle avoidance of a quadrotor UAV. The dynamic model is derived using the Newton-Euler approach with unit quaternions to prevent the Gimbal Lock phenomenon. Differential Flatness is utilized to generate reference trajectories. The NMPC formulation incorporates physical constraints, evaluates attitude error via the logarithmic map (Log Map) on the $SO(3)$ manifold, and includes a geometric constraint for static obstacle avoidance. Implemented in MATLAB using CasADi and IPOPT, simulation results demonstrate precise tracking, robust disturbance rejection, and effective obstacle avoidance capabilities.

Keywords : Quadrotor, NMPC, Quaternions, Differential Flatness, Log Map, Obstacle Avoidance, CasADi, IPOPT.

Introduction Générale

Au cours de la dernière décennie, les Véhicules Aériens Sans Pilote (UAV), et plus particulièrement les quadrotors, ont connu un essor technologique et industriel sans précédent. Autrefois confinés au domaine militaire ou à l'aéromodélisme récréatif, ces systèmes occupent aujourd'hui une place centrale dans de nombreuses applications civiles et industrielles critiques. De l'inspection d'infrastructures inaccessibles à la surveillance agricole, en passant par la cartographie haute résolution, les missions de recherche et sauvetage, ou encore la livraison de matériel médical d'urgence, la polyvalence du quadrotor en fait un outil incontournable. Cet engouement s'explique par sa capacité à décoller et atterrir verticalement (VTOL), son vol stationnaire stable et son agilité exceptionnelle dans des environnements tridimensionnels restreints [2, 3].

Cependant, l'exploitation de cette grande agilité soulève des défis majeurs en matière d'automatique et de théorie du contrôle. Le quadrotor est un système par nature intrinsèquement instable, fortement non linéaire et sous-actionné, ne disposant que de quatre actionneurs (les rotors) pour contrôler ses six degrés de liberté spatiaux (position et orientation). De plus, lors de manœuvres agressives ou de suivis de trajectoires complexes, les représentations classiques de l'attitude, telles que les angles d'Euler, révèlent de graves limites mathématiques en provoquant des singularités cinématiques connues sous le nom de blocage de cardan (*Gimbal Lock*) [4]. Par ailleurs, les contrôleurs linéaires traditionnels de type PID peinent à anticiper les dynamiques futures du système et s'avèrent incapables d'intégrer nativement les contraintes physiques strictes des moteurs (saturation) ou les contraintes de l'environnement (obstacles) [5].

Face à ces limitations, l'objectif principal de ce mémoire est de concevoir, développer et valider une architecture de contrôle avancée, robuste et prédictive, capable d'exploiter tout le potentiel dynamique du quadrotor tout en garantissant un vol sûr. Pour ce faire, notre méthodologie s'articule autour de trois axes majeurs. Premièrement, nous adoptons l'algèbre des quaternions unitaires associée à l'application logarithmique (*Log Map*) pour modéliser l'erreur d'attitude de manière globale et sans singularité. Deuxièmement, nous exploitons la propriété de différentielle plate (*Differential Flatness*) du quadrotor pour générer algébriquement des trajectoires de référence physiquement réalisables [6, 7]. Troisièmement, nous formulons le problème de suivi de trajectoire et d'évitement d'obstacles sous la forme d'une Commande Prédictive Non

Linéaire (NMPC) centralisée, permettant d’optimiser les commandes en temps réel tout en respectant l’ensemble des contraintes du système [8, 9, 10, 11].

Pour exposer notre démarche, ce manuscrit est organisé en trois chapitres complémentaires :

Le Chapitre 1 est consacré à la modélisation mathématique du quadrotor. Il détaille la représentation cinématique par les quaternions unitaires pour pallier les limites des angles d’Euler, et établit le modèle dynamique non linéaire complet en utilisant le formalisme de Newton-Euler.

Le Chapitre 2 constitue le cœur théorique de notre approche. Il présente d’abord la planification de trajectoire par différentielle plate, puis détaille la formulation mathématique rigoureuse du problème d’optimisation NMPC. Ce chapitre inclut la conception de la fonction de coût basée sur l’erreur de quaternion, l’intégration des contraintes physiques et la modélisation géométrique pour l’évitement d’obstacles.

Le Chapitre 3 est dédié à l’implémentation numérique et à la validation des performances. À l’aide du framework CasADi et du solveur IPOPT, nous évaluons l’efficacité de notre loi de commande à travers divers scénarios de simulation incluant le vol nominal, l’introduction de fortes perturbations paramétriques et environnementales (rafales de vent), ainsi que des manœuvres dynamiques d’évitement d’obstacles.

Chapitre 1

Modélisation Mathématique et Dynamique du Quadrotor

La modélisation mathématique constitue une étape essentielle dans l'étude et la commande des systèmes aériens sans pilote, notamment les quadrotors. En raison de leur structure sous-actionnée et de leur dynamique fortement non linéaire, ces véhicules nécessitent une description rigoureuse de leurs mouvements afin de comprendre leur comportement et de développer des stratégies de commande performantes. Une modélisation précise permet également de prédire l'évolution du système et de garantir la stabilité ainsi que la précision du suivi de trajectoire.

Dans ce contexte, ce chapitre est consacré à l'établissement du modèle mathématique et dynamique du quadrotor. Dans un premier temps, les caractéristiques générales des systèmes VTOL et les principes de fonctionnement d'un quadrotor seront présentés. Ensuite, les notions fondamentales de cinématique spatiale seront introduites à travers la définition des repères de référence et l'étude des différentes représentations de l'attitude. Une attention particulière sera accordée aux quaternions unitaires, qui offrent une représentation robuste et exempte des singularités géométriques associées aux angles d'Euler.

Enfin, la modélisation dynamique du quadrotor sera développée en utilisant l'approche de Newton–Euler. Cette démarche permettra de déterminer les équations du mouvement en tenant compte des forces et des moments appliqués au véhicule, conduisant ainsi à l'obtention d'un modèle d'état non linéaire complet à treize dimensions. Ce bloc d'équations servira de socle mathématique à la conception du contrôleur NMPC dans les chapitres suivants.

1.1 Généralités sur les systèmes VTOL et Quadrotors

Les véhicules aériens sans pilote (UAV), communément appelés drones, se déclinent sous diverses architectures aérodynamiques. Parmi celles-ci, les systèmes à décollage et atterrissage verticaux (VTOL : *Vertical Take-Off and Landing*) se distinguent par leur capacité à s'affranchir des pistes de lancement conventionnelles. Cette flexibilité opérationnelle les rend indispensables dans les environnements confinés ou urbains. Au sein de la famille des VTOL, le quadrotor occupe une place prépondérante en raison de sa grande maniabilité, de sa simplicité mécanique et de son rapport coût-performance optimal [12].

Contrairement aux voilures fixes orientées vers des missions de longue endurance, la configuration multirotor à quatre hélices offre l'aptitude essentielle au vol stationnaire et aux évolutions omnidirectionnelles à basse vitesse, jetant ainsi les bases du problème de commande traité dans ce manuscrit.

1.1.1 Principes de vol et architecture matérielle d'un quadrotor

Le principe fondamental du vol d'un quadrotor repose sur la génération d'une poussée verticale par chacun de ses quatre rotors, disposés symétriquement autour d'un châssis rigide en configuration croisée (en « $\times$ » ou en « $+$ »). Afin de compenser et d'équilibrer les couples de réaction aérodynamiques naturellement générés par la rotation des moteurs, deux hélices tournent dans le sens horaire (CW) tandis que les deux autres tournent dans le sens antihoraire (CCW) [13].

Le contrôle complet du véhicule est obtenu en modifiant de façon coordonnée les vitesses de rotation de ces rotors, ce qui permet d'engendrer les forces et les moments requis pour guider l'aéronef (voir la figure 1.1) :

- **Vol stationnaire et translation verticale (Gaz / Lift) :** Obtenus en augmentant ou en diminuant simultanément et uniformément la vitesse des quatre moteurs. Lorsque la poussée totale équilibre exactement le poids du drone, celui-ci se maintient en vol stationnaire [14].
- **Mouvement de roulis (Roll) et de tangage (Pitch) :** Générés en introduisant une variation différentielle de vitesse entre deux moteurs opposés. Une dissymétrie entre les rotors gauche et droit crée un moment de roulis, tandis qu'une dissymétrie entre les rotors avant et arrière produit un moment de tangage.
- **Mouvement de lacet (Yaw) :** Obtenu en créant un déséquilibre entre la somme des couples des moteurs tournant dans le sens horaire et celle des moteurs

tournant dans le sens antihoraire, induisant une rotation du drone autour de son axe vertical.

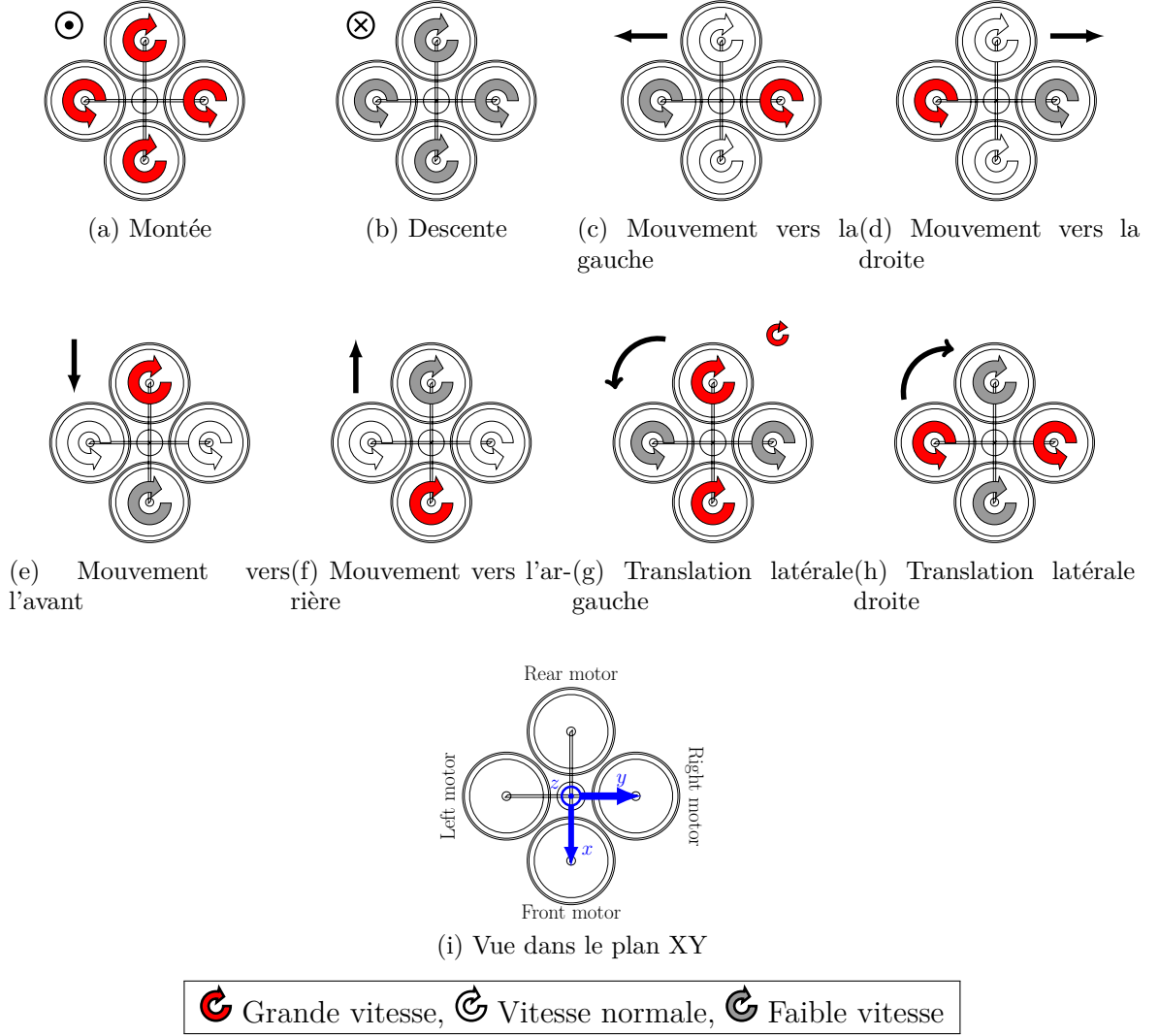


Figure 1.1 – Illustration des mouvements du quadrirotor générés par la variation des vitesses angulaires de ses rotors [1].

Sur le plan matériel, l'architecture physique d'un quadrotor articule plusieurs composants critiques indispensables à son automatisation : un châssis léger et rigide, quatre moteurs électriques de type *brushless* pilotés par des contrôleurs électroniques de vitesse (ESC), une batterie d'alimentation à forte densité énergétique (LiPo), et une unité de calcul embarquée. Cette dernière centralise les mesures d'une Unité de Mesure Inertielle (IMU) — combinant accéléromètres et gyromètres — complétée éventuellement par des capteurs de positionnement (GPS, Lidars) dépendamment des contraintes de l'application visée [13].

1.1.2 Défis de commande : système sous-actionné et fortement non linéaire

Malgré une apparente simplicité mécanique, la stabilisation et le guidage d'un quadrotor soulèvent des défis méthodologiques complexes en automatique quantitative. Le premier obstacle théorique réside dans la nature sous-actionnée du système. En effet, l'aéronef évolue dans un espace à six degrés de liberté (DoF) — définis par trois coordonnées de translation spatiale et trois angles d'orientation — alors qu'il ne dispose structurellement que de quatre entrées de commande indépendantes issues des forces de poussée des quatre moteurs [13].

Cette configuration sous-actionnée impose un couplage dynamique strict entre les mouvements de translation et les mouvements de rotation. Le quadrotor est incapable de générer une accélération latérale ou longitudinale directe sans incliner préalablement son châssis. Pour se déplacer horizontalement, il doit modifier son attitude afin de projeter une composante de son vecteur de poussée principal selon l'axe de déplacement désiré. Par conséquent, l'asservissement cinématique en position est entièrement subordonné à la dynamique interne de la boucle d'attitude.

De surcroît, les équations du mouvement sont caractérisées par de fortes non-linéarités, exacerbées par les termes de couplage gyroscopique, les matrices de rotation trigonométriques et les produits croisés des vitesses angulaires. Ces phénomènes non linéaires dégradent significativement les performances des régulateurs linéaires classiques dès que le véhicule s'éloigne de son point d'équilibre nominal ou réalise des manœuvres à haute dynamique. Ce constat justifie le recours à des approches de commande avancées, à l'instar de la Commande Prédictive Non Linéaire (NMPC), capable de gérer explicitement ces non-linéarités tout en garantissant le respect strict des contraintes physiques et opérationnelles du système [15].

1.2 Cinématique Spatiale et Représentation de l'Attitude

La description du mouvement d'un corps rigide évoluant dans l'espace tridimensionnel requiert la distinction stricte entre sa cinématique de translation et sa cinématique de rotation (attitude). Alors que la position linéaire s'exprime de manière intuitive dans un espace euclidien classique, la modélisation de l'orientation pose des défis topologiques majeurs. Cette section formalise les repères d'étude nécessaires, met en évidence les limitations géométriques des représentations paramétriques minimales, et pose les bases fondamentales de l'algèbre des quaternions unitaires retenue pour ce travail [16].

1.2.1 Les repères d'étude : Repère inertiel (Terre) et Repère mobile (Corps)

Pour décrire sans ambiguïté les forces, les moments, ainsi que les variables d'état cinématiques du quadrotor, il est indispensable de définir deux repères orthogonaux directs conformes aux conventions aéronautiques internationales (illustrés sur la figure 1.2) :

1. **Le repère inertiel** (ou repère fixe) $\mathcal{R}_i = \{O_i, \mathbf{X}_i, \mathbf{Y}_i, \mathbf{Z}_i\}$: Ce repère est solidaire de la Terre et est considéré comme galiléen, ce qui permet l'application directe des lois de la mécanique newtonienne. L'origine O_i est fixée à un point de départ arbitraire au sol. Selon la convention *East-North-Up* (ENU) adoptée dans ce travail, l'axe $\mathbf{X}_i$ pointe vers l'Est, l'axe $\mathbf{Y}_i$ vers le Nord, et l'axe $\mathbf{Z}_i$ est dirigé verticalement vers le haut, de façon colinéaire mais opposée au champ de gravité terrestre. La position linéaire du centre de masse du drone dans ce repère est définie par le vecteur [17] :

$$\mathbf{p} = [x, y, z]^T \in \mathbb{R}^3 \quad (1.1)$$

La vitesse linéaire par rapport à ce repère fixe est obtenue par dérivation temporelle :

$$\mathbf{v} = [v_x, v_y, v_z]^T = [\dot{x}, \dot{y}, \dot{z}]^T \in \mathbb{R}^3 \quad (1.2)$$

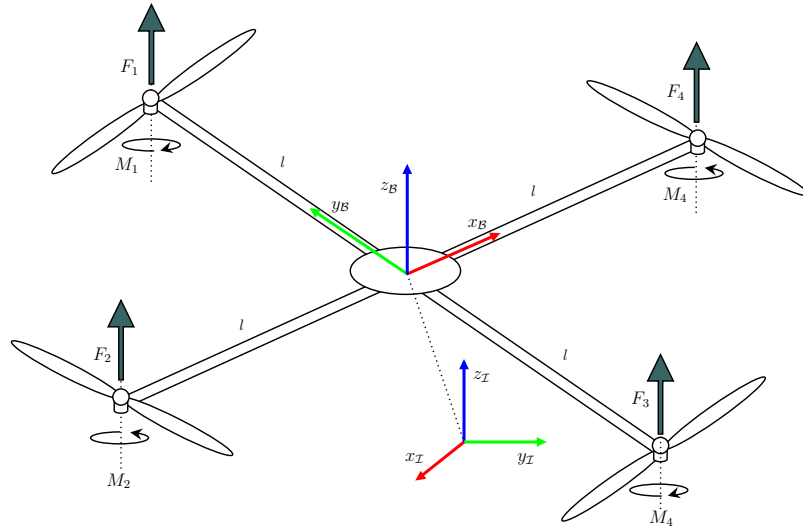


Figure 1.2 – Géométrie générale du quadrotor et définition des repères inertiel et mobile [1]

2. **Le repère mobile** (ou repère corps) $\mathcal{R}_b = \{O_b, \mathbf{X}_b, \mathbf{Y}_b, \mathbf{Z}_b\}$: Ce repère est intrinsèquement lié à la structure rigide du quadrotor. Son origine O_b coïncide en permanence avec le centre de masse de l'appareil. Les axes $\mathbf{X}_b$ et $\mathbf{Y}_b$ sont alignés avec les bras de la structure supportant les moteurs, définissant le plan de sustentation. L'axe $\mathbf{Z}_b$ est orthogonal à ce plan et est orienté vers le haut, correspondant au sens

de la poussée aérodynamique totale. Les capteurs embarqués, notamment l'unité de mesure inertielle (IMU), mesurent les grandeurs physiques directement dans ce repère. La vitesse angulaire du drone y est définie par le vecteur [18] :

$$\boldsymbol{\omega} = [\omega_x, \omega_y, \omega_z]^T \in \mathbb{R}^3 \quad (1.3)$$

où ω_x , ω_y et ω_z représentent respectivement les vitesses de roulis (*roll rate*), de tangage (*pitch rate*) et de lacet (*yaw rate*).

L'orientation du repère mobile $\mathcal{R}_b$ par rapport au repère de référence $\mathcal{R}_i$ est modélisée par une matrice de rotation $\mathbf{R} \in \text{SO}(3)$, où $\text{SO}(3)$ désigne le groupe spécial orthogonal d'ordre 3, satisfaisant les propriétés suivantes :

$$\mathbf{R}^T \mathbf{R} = \mathbf{I}_3, \quad \det(\mathbf{R}) = +1 \quad (1.4)$$

1.2.2 Les limites des angles d'Euler et le phénomène de Blocage de Cardan (Gimbal Lock)

La représentation de l'attitude par les angles d'Euler (Roulis ϕ , Tangage θ , Lacet ψ) repose sur une séquence de trois rotations successives effectuées autour des axes mobiles, suivant la convention aéronautique Tait-Bryan $Z - Y' - X''$. Bien que ce paramétrage minimal à trois variables soit visuellement intuitif, il présente une défaillance topologique critique lors de l'établissement des équations cinématiques [17], comme le schématise la figure 1.3.

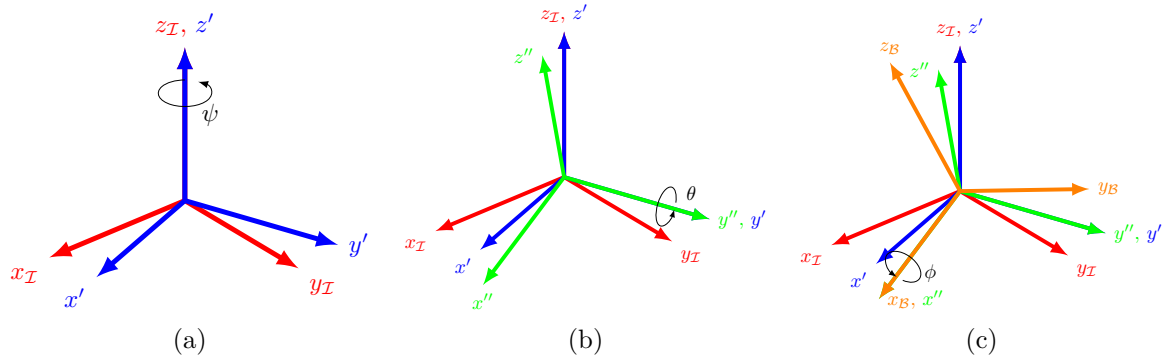


Figure 1.3 – Représentation des rotations successives $Z - Y' - X''$ et changement de repères par les angles d'Euler [1].

La relation cinématique liant le vecteur des vitesses angulaires physiques $\boldsymbol{\omega}$ aux dérivées temporelles des angles d'Euler s'écrit :

$$\begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} = \begin{bmatrix} 1 & 0 & -\sin \theta \\ 0 & \cos \phi & \sin \phi \cos \theta \\ 0 & -\sin \phi & \cos \phi \cos \theta \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \quad (1.5)$$

Pour exprimer l'évolution temporelle des angles d'Euler en fonction des vitesses mesurées par les gyromètres, l'inversion de cette matrice cinématique donne :

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & \sin \phi \tan \theta & \cos \phi \tan \theta \\ 0 & \cos \phi & -\sin \phi \\ 0 & \frac{\sin \phi}{\cos \theta} & \frac{\cos \phi}{\cos \theta} \end{bmatrix} \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} \quad (1.6)$$

L'analyse mathématique de cette formulation met en évidence une singularité incontournable lorsque le pas de tangage approche les valeurs critiques $\theta \rightarrow \pm \frac{\pi}{2}$. À ces points, les fonctions $\tan \theta$ et $\frac{1}{\cos \theta}$ divergent vers l'infini, provoquant l'effondrement des algorithmes de calcul numérique embarqués.

Physiquement, ce phénomène est connu sous le nom de *Blocage de Cardan* (*Gimbal Lock*) : l'axe du mouvement de roulis se retrouve confondu avec l'axe du mouvement de lacet. La représentation perd instantanément un degré de liberté géométrique, rendant le système incapable de dissocier une rotation de roulis d'une rotation de lacet. Pour un drone autonome devant effectuer des trajectoires complexes à forte dynamique ou subissant des perturbations sévères, cette singularité est intolérable. Cela justifie le choix d'un paramétrage global et non singulier : les quaternions unitaires.

1.2.3 Algèbre des Quaternions unitaires : Propriétés et opérations fondamentales

Définition mathématique

Les quaternions constituent une extension hypercomplexe des nombres complexes permettant de modéliser les rotations dans l'espace tridimensionnel sans recourir aux fonctions trigonométriques. Ils ont été introduits en 1843 par le mathématicien William Rowan Hamilton [19]. Un quaternion général $\mathbf{q}$ est défini par :

$$\mathbf{q} = q_0 + q_1 i + q_2 j + q_3 k \quad (1.7)$$

où $q_0 \in \mathbb{R}$ représente la partie réelle (scalaire), $q_1, q_2, q_3 \in \mathbb{R}$ désignent les composantes de la partie imaginaire (vectorielle), et i, j, k sont les unités imaginaires fondamentales d'Hamilton satisfaisant les relations :

$$i^2 = j^2 = k^2 = ijk = -1 \quad (1.8)$$

$$ij = k, \quad jk = i, \quad ki = j, \quad ji = -k, \quad kj = -i, \quad ik = -j \quad (1.9)$$

De manière compacte, le quaternion est représenté par le couple formé de sa partie scalaire et de son vecteur vectoriel :

$$\mathbf{q} = \begin{bmatrix} q_0 \\ \mathbf{q}_v \end{bmatrix}, \quad \mathbf{q}_v = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \end{bmatrix} \in \mathbb{R}^3 \quad (1.10)$$

Quaternion de rotation

D'après le théorème de rotation d'Euler, toute orientation peut être décrite par une rotation unique d'un angle θ autour d'un axe unitaire fixe $\mathbf{u} = [u_x, u_y, u_z]^T$. Le quaternion unitaire correspondant est formalisé par [20] :

$$\mathbf{q} = \begin{bmatrix} \cos\left(\frac{\theta}{2}\right) \\ \mathbf{u} \sin\left(\frac{\theta}{2}\right) \end{bmatrix} = \begin{bmatrix} \cos(\theta/2) \\ u_x \sin(\theta/2) \\ u_y \sin(\theta/2) \\ u_z \sin(\theta/2) \end{bmatrix} \quad (1.11)$$

La norme d'un quaternion est définie par $\|\mathbf{q}\| = \sqrt{q_0^2 + q_1^2 + q_2^2 + q_3^2}$. Pour représenter une orientation physique valide, le quaternion doit impérativement respecter la contrainte de normalisation suivante :

$$\|\mathbf{q}\| = 1 \quad (1.12)$$

Opérations algébriques élémentaires

La manipulation de l'attitude requiert la définition des opérateurs fondamentaux suivants :

- **Conjugaison** : Le conjugué d'un quaternion $\mathbf{q}$ inversant le sens de la rotation est donné par :

$$\mathbf{q}^* = \begin{bmatrix} q_0 \\ -\mathbf{q}_v \end{bmatrix} = [q_0, -q_1, -q_2, -q_3]^T \quad (1.13)$$

- **Inverse** : L'inverse algébrique est défini par $\mathbf{q}^{-1} = \frac{\mathbf{q}^*}{\|\mathbf{q}\|^2}$. Pour les quaternions unitaires ($\|\mathbf{q}\| = 1$), l'inverse coïncide avec le conjugué :

$$\mathbf{q}^{-1} = \mathbf{q}^* \quad (1.14)$$

- **Produit quaternionique** : La composition de deux rotations successives déterminée par les quaternions $\mathbf{q}$ et $\mathbf{p}$ s'effectue via le produit de Hamilton, noté

$\otimes$ [18] :

$$\mathbf{q} \otimes \mathbf{p} = \begin{bmatrix} q_0 p_0 - q_1 p_1 - q_2 p_2 - q_3 p_3 \\ q_0 p_1 + q_1 p_0 + q_2 p_3 - q_3 p_2 \\ q_0 p_2 - q_1 p_3 + q_2 p_0 + q_3 p_1 \\ q_0 p_3 + q_1 p_2 - q_2 p_1 + q_3 p_0 \end{bmatrix} \quad (1.15)$$

Changement de repère d'un vecteur par quaternion

Pour appliquer une rotation à un vecteur physique $\mathbf{v} \in \mathbb{R}^3$, ce dernier est transposé sous la forme d'un quaternion pur $\mathbf{v}_q = [0, \mathbf{v}^T]^T$. Le vecteur pivoté $\mathbf{v}'$ exprimé dans le nouveau repère est extrait de l'opération [16] :

$$\begin{bmatrix} 0 \\ \mathbf{v}' \end{bmatrix} = \mathbf{q} \otimes \mathbf{v}_q \otimes \mathbf{q}^{-1} \quad (1.16)$$

1.2.4 Modélisation de la cinématique d'attitude par quaternions

La cinématique d'attitude décrit la relation mathématique liant le taux de variation temporel de l'orientation (la dérivée du quaternion) au vecteur de la vitesse angulaire $\boldsymbol{\omega}$ mesuré par la centrale inertielle dans le repère mobile. Contrairement aux approches basées sur les angles d'Euler, cette formulation est globalement régulière sur l'ensemble de la sphère unité $\mathbb{S}^3$ [16].

L'évolution différentielle du quaternion d'attitude est régie par le système d'équations linéaires suivant :

$$\dot{\mathbf{q}} = \frac{1}{2} \Omega(\boldsymbol{\omega}) \mathbf{q} \quad (1.17)$$

où $\Omega(\boldsymbol{\omega})$ représente la matrice cinématique antisymétrique construite à partir des composantes de la vitesse angulaire du corps $\boldsymbol{\omega} = [\omega_x, \omega_y, \omega_z]^T$ [20] :

$$\Omega(\boldsymbol{\omega}) = \begin{bmatrix} 0 & -\omega_x & -\omega_y & -\omega_z \\ \omega_x & 0 & \omega_z & -\omega_y \\ \omega_y & -\omega_z & 0 & \omega_x \\ \omega_z & \omega_y & -\omega_x & 0 \end{bmatrix} \quad (1.18)$$

1.3 Modélisation Dynamique par l'Approche de Newton-Euler

La modélisation dynamique consiste à établir les équations mathématiques formelles qui relient les forces et les moments appliqués au quadrotor à ses accélérations linéaires et angulaires. L'approche de Newton-Euler est particulièrement adaptée pour décrire

le comportement d'un corps rigide évoluant dans un espace tridimensionnel, en découplant explicitement le mouvement de translation (traité dans le repère inertiel via le formalisme de Newton) du mouvement de rotation (traité dans le repère mobile via le formalisme d'Euler) [21].

1.3.1 Modélisation des forces appliquées (Gravité et Poussée aérodynamique)

Le quadrotor est soumis à deux forces extérieures principales agissant sur son centre de masse : la force gravitationnelle et la force de poussée aérodynamique globale générée par la rotation des quatre hélices.

- **Force gravitationnelle $\mathbf{F}_g$** : Elle s'exerce en permanence selon l'axe vertical descendant du repère inertiel $\mathcal{R}_i$ et s'exprime par :

$$\mathbf{F}_g = \begin{bmatrix} 0 \\ 0 \\ -mg \end{bmatrix} \quad (1.19)$$

où m représente la masse totale du drone, et g désigne l'accélération de la pesanteur terrestre.

- **Force de poussée aérodynamique totale $\mathbf{F}_b$** : Chaque hélice i (pour $i \in \{1, 2, 3, 4\}$) tournant à une vitesse angulaire ω_i génère une force de poussée aérodynamique orientée suivant l'axe vertical positif $\mathbf{Z}_b$ du repère mobile [22]. Exprimée dans ce repère mobile, la force de poussée globale est la somme algébrique des poussées individuelles des quatre rotors :

$$\mathbf{F}_b = \begin{bmatrix} 0 \\ 0 \\ \sum_{i=1}^4 b\omega_i^2 \end{bmatrix} \quad (1.20)$$

où $b > 0$ désigne le coefficient de poussée aérodynamique dépendant de la géométrie des pales et de la densité de l'air.

En appliquant le principe fondamental de la dynamique de Newton dans le repère inertiel, le mouvement de translation du centre de masse est régi par la relation suivante [23] :

$$m\ddot{\mathbf{p}} = \mathbf{R}(\mathbf{q})\mathbf{F}_b + \mathbf{F}_g \quad (1.21)$$

où $\mathbf{p} = [x, y, z]^T$ est le vecteur position du drone et $\mathbf{R}(\mathbf{q}) \in \text{SO}(3)$ représente la matrice de rotation cinématique tridimensionnelle paramétrée par le quaternion d'attitude $\mathbf{q}$.

La matrice de rotation $\mathbf{R}(\mathbf{q}) \in \text{SO}(3)$, qui permet de projeter les forces exprimées dans le repère mobile du corps $\mathcal{R}_b$ vers le repère inertiel $\mathcal{R}_i$, est explicitement définie en

fonction des composantes du quaternion unitaire $\mathbf{q} = [q_0, q_1, q_2, q_3]^T$ (où q_0 représente la partie scalaire et $[q_1, q_2, q_3]^T$ la partie vectorielle) par la relation suivante :

$$\mathbf{R}(\mathbf{q}) = \begin{bmatrix} 1 - 2(q_2^2 + q_3^2) & 2(q_1q_2 - q_0q_3) & 2(q_1q_3 + q_0q_2) \\ 2(q_1q_2 + q_0q_3) & 1 - 2(q_1^2 + q_3^2) & 2(q_2q_3 - q_0q_1) \\ 2(q_1q_3 - q_0q_2) & 2(q_2q_3 + q_0q_1) & 1 - 2(q_1^2 + q_2^2) \end{bmatrix} \quad (1.22)$$

Cette représentation matricielle directe possède la propriété fondamentale d'être purement algébrique, évitant ainsi le calcul de fonctions trigonométriques coûteuses en temps de calcul (cos, sin) indispensables lors de l'usage des angles d'Euler.

1.3.2 Modélisation des moments appliqués (Couples de contrôle et effets gyroscopiques)

Les moments agissant sur le quadrotor déterminent l'intégralité de sa dynamique rotationnelle. Ils se composent des couples de contrôle utiles générés par la variation volontaire de la vitesse des moteurs et des moments perturbateurs induits par les effets gyroscopiques structurels [22] (voir la figure 1.4).

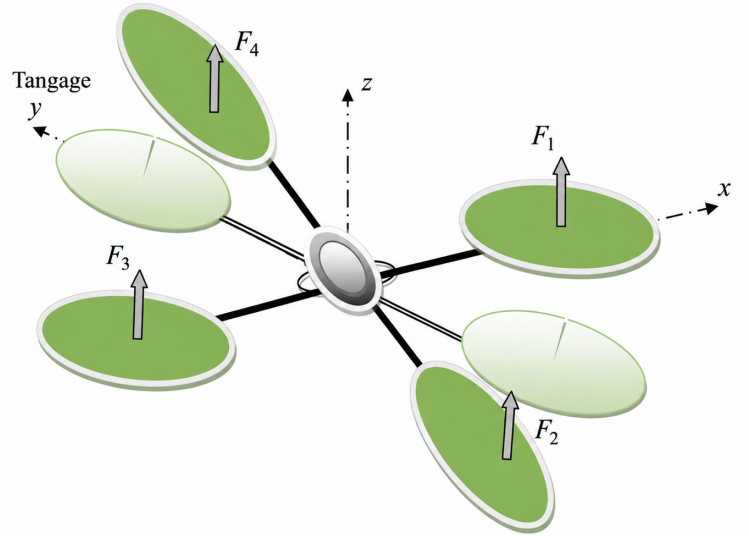


Figure 1.4 – Mécanisme aérodynamique de génération du moment de tangage par variation différentielle de vitesse (Source : [1]).

- **Couples de contrôle (τ_c)** : Ils permettent d'orienter le drone selon ses trois degrés de liberté en rotation (roulis, tangage, lacet) en modifiant de manière

différentielle les vitesses des hélices selon la configuration structurelle :

$$\boldsymbol{\tau}_c = \begin{bmatrix} \tau_\phi \\ \tau_\theta \\ \tau_\psi \end{bmatrix} = \begin{bmatrix} bl(\omega_4^2 - \omega_2^2) \\ bl(\omega_3^2 - \omega_1^2) \\ d(\omega_1^2 - \omega_2^2 + \omega_3^2 - \omega_4^2) \end{bmatrix} \quad (1.23)$$

où τ_ϕ , τ_θ , τ_ψ sont les moments respectifs de roulis (*Roll*), de tangage (*Pitch*) et de lacet (*Yaw*), l est la distance séparant le centre de gravité de l'axe de rotation de chaque moteur (longueur du bras), et d est le coefficient de traînée aérodynamique du rotor.

- **Effets gyroscopiques des rotors (τ_g)** : Lors des changements d'attitude rapides, la rotation à haute vitesse des hélices combinée au mouvement de précession du châssis engendre un couple gyroscopique perturbateur, agissant de manière croisée sur les axes de roulis et de tangage [21] :

$$\boldsymbol{\tau}_g = \begin{bmatrix} \omega_y J_r \Omega_r \\ -\omega_x J_r \Omega_r \\ 0 \end{bmatrix} \quad (1.24)$$

où J_r désigne le moment d'inertie des pièces mobiles tournantes (moteur et hélice), et $\Omega_r = \omega_1 - \omega_2 + \omega_3 - \omega_4$ représente la vitesse angulaire résiduelle globale des rotors.

En appliquant le théorème d'Euler pour les corps rigides en rotation exprimé directement dans le repère mobile du corps, la dynamique de rotation s'écrit :

$$\mathbf{J}\dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \times (\mathbf{J}\boldsymbol{\omega}) = \boldsymbol{\tau}_c + \boldsymbol{\tau}_g \quad (1.25)$$

où $\mathbf{J} = \text{diag}(I_x, I_y, I_z)$ représente la matrice d'inertie principale du quadrotor selon ses axes de symétrie.

1.3.3 Justification et formulation du modèle d'état non linéaire complet

Justification mathématique de la représentation d'état

Le quadrotor constitue un système mécanique fortement non linéaire, multivariable et par nature sous-actionné. Il évolue dans un espace à six degrés de liberté (DoF) mais ne dispose que de quatre entrées physiques indépendantes. De surcroît, sa dynamique de translation est intrinsèquement couplée à sa dynamique d'attitude (la modification des variables de position requiert impérativement une inclinaison préalable du châssis).

Pour concevoir des lois de commande optimales robustes à l'instar de la commande

prédictive non linéaire (NMPC), il est indispensable d'établir un modèle d'état complet capable de capturer ces couplages et non-linéarités sans recourir à des approximations de linéarisation locale. L'intégration des quaternions unitaires en lieu et place des angles d'Euler classiques permet d'aboutir à un espace d'état global, régulier et exempt de singularité géométrique (*Gimbal Lock*), de dimension treize.

Définition des vecteurs d'état et de commande

Pour caractériser de manière unique et exhaustive l'évolution temporelle de la trajectoire et de l'orientation du drone, le vecteur d'état $\mathbf{x} \in \mathbb{R}^{13}$ est structuré en assemblant les coordonnées de position, de vitesse linéaire, le quaternion d'attitude et les vitesses angulaires :

$$\mathbf{x} = [\mathbf{p}^T, \mathbf{v}^T, \mathbf{q}^T, \boldsymbol{\omega}^T]^T = [x, y, z, v_x, v_y, v_z, q_0, q_1, q_2, q_3, \omega_x, \omega_y, \omega_z]^T \quad (1.26)$$

Le vecteur de commande $\mathbf{u} \in \mathbb{R}^4$ regroupe les entrées énergétiques indépendantes du système. Ces entrées sont directement définies par le carré des vitesses de rotation de chacun des quatre moteurs électriques, notées ω_i (pour $i \in \{1, 2, 3, 4\}$) :

$$\mathbf{u} = [u_1, u_2, u_3, u_4]^T = [\omega_1^2, \omega_2^2, \omega_3^2, \omega_4^2]^T \quad (1.27)$$

Ces vitesses de rotation aérodynamiques permettent de générer la force de poussée totale du drone, notée F_b (dirigée selon l'axe vertical du repère mobile $\mathbf{b}_z$), ainsi que le vecteur des moments de commande (couples aérodynamiques) noté $\boldsymbol{\tau}_b = [\tau_x, \tau_y, \tau_z]^T$ selon les axes de roulis (*roll*), de tangage (*pitch*) et de lacet (*yaw*).

La relation mathématique qui relie ces efforts physiques (F_b et $\boldsymbol{\tau}_b$) au vecteur des commandes de commande $\mathbf{u}$ est une transformation statique linéaire. Elle dépend explicitement du coefficient de poussée (ou de portance) b , du coefficient de traînée aérodynamique d , et de la longueur du bras du quadrotor l (la distance séparant le centre de gravité du drone de l'axe de rotation de chaque moteur).

En adoptant la configuration géométrique du système, cette relation d'allocation de puissance s'exprime sous la forme matricielle suivante :

$$\begin{bmatrix} F_b \\ \tau_x \\ \tau_y \\ \tau_z \end{bmatrix} = \begin{bmatrix} b & b & b & b \\ 0 & l \cdot b & 0 & -l \cdot b \\ -l \cdot b & 0 & l \cdot b & 0 \\ d & -d & d & -d \end{bmatrix} \begin{bmatrix} \omega_1^2 \\ \omega_2^2 \\ \omega_3^2 \\ \omega_4^2 \end{bmatrix} \quad (1.28)$$

Ce qui se traduit de manière développée par le système d'équations algébriques

suivant :

$$F_b = b \left(\omega_1^2 + \omega_2^2 + \omega_3^2 + \omega_4^2 \right) \quad (1.29a)$$

$$\tau_x = l \cdot b \left(\omega_2^2 - \omega_4^2 \right) \quad (1.29b)$$

$$\tau_y = l \cdot b \left(\omega_3^2 - \omega_1^2 \right) \quad (1.29c)$$

$$\tau_z = d \left(\omega_1^2 - \omega_2^2 + \omega_3^2 - \omega_4^2 \right) \quad (1.29d)$$

Où les paramètres physiques intervenant dans ce modèle sont définis comme suit :

- $b > 0$ représente le coefficient de poussée aérodynamique des hélices ($N \cdot rad^{-2} \cdot s^2$).
- $d > 0$ représente le coefficient de traînée aérodynamique s'opposant au couple moteur ($N \cdot m \cdot rad^{-2} \cdot s^2$).
- l désigne la longueur structurelle du bras du quadrotor (m).

Formulation générale du modèle d'état non linéaire

En combinant les équations cinématiques issues de la section 1.2 et les équations dynamiques de Newton-Euler, le système différentiel non linéaire du premier ordre s'organise sous la forme d'état standardisé $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u})$:

$$\dot{\mathbf{x}} = \begin{bmatrix} \dot{\mathbf{p}} \\ \dot{\mathbf{v}} \\ \dot{\mathbf{q}} \\ \dot{\boldsymbol{\omega}} \end{bmatrix} = \begin{bmatrix} \mathbf{v} \\ \frac{1}{m} \mathbf{R}(\mathbf{q}) \mathbf{F}_b + \mathbf{F}_g \\ \frac{1}{2} \boldsymbol{\Omega}(\boldsymbol{\omega}) \mathbf{q} \\ \mathbf{J}^{-1} (\boldsymbol{\tau}_c + \boldsymbol{\tau}_g - \boldsymbol{\omega} \times (\mathbf{J} \boldsymbol{\omega})) \end{bmatrix} \quad (1.30)$$

En développant explicitement chaque bloc vectoriel en fonction des composantes d'état et des entrées de commande u_i , la formulation non linéaire explicite complète se

déploie comme suit :

$$\begin{bmatrix} \dot{\mathbf{p}} \\ \dot{\mathbf{v}} \\ \dot{\mathbf{q}} \\ \dot{\boldsymbol{\omega}} \end{bmatrix} = \begin{bmatrix} v_x \\ v_y \\ v_z \\ \frac{1}{m} \left(\mathbf{R}(\mathbf{q}) \begin{bmatrix} 0 \\ 0 \\ b(u_1 + u_2 + u_3 + u_4) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ -mg \end{bmatrix} \right) \\ \frac{1}{2} \Omega(\boldsymbol{\omega}) \mathbf{q} \\ \mathbf{J}^{-1} \left(\begin{bmatrix} lb(u_4 - u_2) \\ lb(u_3 - u_1) \\ d(u_1 - u_2 + u_3 - u_4) \end{bmatrix} + \begin{bmatrix} \omega_y J_r \Omega_r \\ -\omega_x J_r \Omega_r \\ 0 \end{bmatrix} - \boldsymbol{\omega} \times (\mathbf{J} \boldsymbol{\omega}) \right) \end{bmatrix} \quad (1.31)$$

Conclusion du chapitre

Ce premier chapitre a permis d'établir les fondements théoriques et géométriques indispensables à l'étude cinématique et dynamique du drone quadrotor. Après une mise en contexte globale situant le quadrotor au sein des différentes architectures de systèmes VTOL, ses défis opérationnels et ses fortes contraintes physiques ont été mis en relief.

Afin de s'affranchir définitivement de la singularité topologique du blocage de Cardan (*Gimbal Lock*), inhérente aux représentations minimales par angles d'Euler, le formalisme rigoureux des quaternions unitaires a été introduit. En s'appuyant sur ces concepts hypercomplexes, la cinématique d'attitude ainsi que les équations de Newton-Euler régissant les mouvements de translation et de rotation ont été entièrement développées.

Finalement, la synthèse de ces formulations a abouti à la construction d'un modèle d'état non linéaire compact de dimension treize. Ce modèle global, exempt de singularités et capturant l'ensemble des couplages du système, constitue le socle analytique rigoureux sur lequel sera synthétisée, dans le chapitre suivant, la stratégie de Commande Prédictive Non Linéaire (NMPC) pour assurer la stabilisation robuste et le guidage de l'aéronef.

Chapitre 2

Génération de Trajectoire et Conception du Contrôleur NMPC

Introduction

Suite à l'établissement du modèle dynamique du quadrotor dans le chapitre précédent, ce deuxième chapitre est consacré aux étapes fondamentales permettant d'assurer son vol autonome : la génération d'une trajectoire réalisable et la conception d'une loi de commande robuste, optimale et sécurisée.

Dans un premier temps, nous aborderons la planification de trajectoire en exploitant la propriété de la différentielle plate (differential flatness) du quadrotor. Cette approche mathématique élégante nous permettra, à partir d'un choix judicieux de variables plates (la position spatiale et l'angle de lacet), de déduire algébriquement les états de référence idéaux ainsi que les commandes nominales anticipées (feedforward), garantissant ainsi une trajectoire physiquement réalisable par le drone.

Une fois cette trajectoire de référence générée, la deuxième partie du chapitre sera dédiée à la formulation du Problème de Contrôle Optimal (OCP) à travers la Commande Prédictive Non Linéaire (NMPC) en horizon glissant. Nous définirons une fonction de coût quadratique visant à trouver le meilleur compromis entre la précision du suivi et la minimisation de l'effort de commande. Une attention particulière sera portée à la représentation de l'orientation : pour éviter les singularités tout en mesurant rigoureusement l'écart directionnel, l'erreur d'attitude sera évaluée directement sur la variété différentiable $SO(3)$ via l'application logarithmique (Log Map).

Pour que cette commande soit applicable dans un contexte réel, il est impératif d'y intégrer les contraintes intrinsèques du système. Ainsi, nous détaillerons la discrétisation de la dynamique interne par la méthode de Runge-Kutta 4, la gestion des limites physiques des actionneurs (saturation de la poussée et des couples), ainsi que la préservation géométrique de l'attitude par la contrainte stricte de normalisation du

quaternion.

Enfin, pour doter le quadrotor d'une capacité de navigation évoluée, la dernière section introduira la modélisation mathématique pour l'évitement d'obstacles. Nous modéliserons des obstacles statiques sous forme de sphères géométriques, qui seront intégrés au problème d'optimisation sous forme de contraintes d'inégalité basées sur la distance euclidienne.

L'objectif ultime de ce chapitre est de formuler un contrôleur prédictif complet, capable non seulement de stabiliser le drone et de suivre une trajectoire complexe avec une haute précision, mais aussi d'anticiper et d'éviter les collisions dans son environnement.

2.1 Planification de Trajectoire par Différentielle Plate

2.1.1 Concept mathématique de la différentielle plate appliquée au quadrotor

La propriété de platitude différentielle (*differential flatness*), introduite par Fliess et al. en 1995 [6], constitue une caractéristique géométrique puissante des systèmes dynamiques non linéaires commandés. Un système est dit différentiellement plat s'il existe un ensemble de variables indépendantes, appelées *sorties plates*, permettant d'exprimer l'intégralité des variables d'état et des entrées de commande par des relations purement algébriques, sans qu'il soit nécessaire d'intégrer numériquement les équations différentielles du modèle.

Mathématiquement, pour un système défini par un vecteur d'état $\mathbf{x}$ et un vecteur de commande $\mathbf{u}$, il existe deux applications lisses Φ et Ψ telles que :

$$\mathbf{x} = \Phi(\mathbf{y}, \dot{\mathbf{y}}, \ddot{\mathbf{y}}, \dots, \mathbf{y}^{(q)}) \quad (2.1)$$

$$\mathbf{u} = \Psi(\mathbf{y}, \dot{\mathbf{y}}, \ddot{\mathbf{y}}, \dots, \mathbf{y}^{(q)}) \quad (2.2)$$

Où :

- $\mathbf{x} \in \mathbb{R}^n$: Le vecteur d'état complet du système.
- $\mathbf{u} \in \mathbb{R}^m$: Le vecteur des entrées de commande.
- $\mathbf{y} \in \mathbb{R}^m$: Le vecteur des sorties plates, dont la dimension est strictement égale au nombre d'entrées m .
- $\mathbf{y}^{(k)}$: La dérivée temporelle d'ordre k du vecteur des sorties plates.
- Φ, Ψ : Des applications algébriques lisses (continûment différentiables).

Pour le quadrotor, cette propriété fondamentale permet de transposer une planification

de trajectoire complexe dans l'espace d'état en un simple problème d'interpolation polynomiale ou géométrique au sein de l'espace des sorties plates.

2.1.2 Choix des variables plates (Position $\mathbf{p}$ et angle de lacet ψ)

Le quadrotor étant un système sous-actionné possédant six degrés de liberté pour seulement quatre actionneurs indépendants, le choix des sorties plates se porte judicieusement sur la position cartésienne du centre de masse et l'angle de lacet (*yaw*). Nous définissons ainsi le vecteur des sorties plates $\mathbf{y} \in \mathbb{R}^4$ par :

$$\mathbf{y} = [x, y, z, \psi]^T \quad (2.3)$$

Où les variables sont explicitées de la manière suivante [7] :

- $\mathbf{p} = [x, y, z]^T \in \mathbb{R}^3$: Le vecteur position du centre de gravité du drone, exprimé dans le repère fixe terrestre (inertiel) $\mathcal{R}_I$.
- x, y, z : Les coordonnées cartésiennes associées aux mouvements de translation (longitudinale, latérale et verticale).
- $\psi \in [-\pi, \pi[$: L'angle de lacet, définissant la rotation de l'aéronef autour de son axe vertical corps.

Ce choix est physiquement et cinématiquement justifié : la trajectoire spatiale tridimensionnelle et l'orientation du nez du drone définissent entièrement l'exécution de sa mission. Les mouvements de roulis (*roll*) et de tangage (*pitch*) ne sont que des variables intermédiaires induites, nécessaires pour orienter le vecteur de poussée totale dans la direction désirée.

2.1.3 Dédution algébrique des états de référence idéaux ($\mathbf{x}_{ref}$)

Le vecteur d'état complet du quadrotor, basé sur la formulation rigoureuse par quaternions établie au Chapitre 1, est structuré comme suit :

$$\mathbf{x} = [\mathbf{p}^T, \mathbf{v}^T, \mathbf{q}^T, \boldsymbol{\omega}^T]^T \in \mathbb{R}^{13} \quad (2.4)$$

Où les composantes cinématiques et dynamiques sont :

- $\mathbf{v}$: Le vecteur vitesse linéaire du drone exprimé dans le repère inertiel $\mathcal{R}_I$.
- $\mathbf{q}$: Le quaternion d'attitude unitaire représentant l'orientation du repère lié au corps $\mathcal{R}_B$ par rapport au repère $\mathcal{R}_I$.
- $\boldsymbol{\omega}$: Le vecteur vitesse angulaire exprimé dans le repère mobile $\mathcal{R}_B$.

À partir de la trajectoire nominale de la sortie plate $\mathbf{y}(t)$ et de ses dérivées successives jusqu'au jerk ($\dot{\mathbf{a}} = \mathbf{p}^{(3)}$), l'état de référence complet peut être reconstruit de

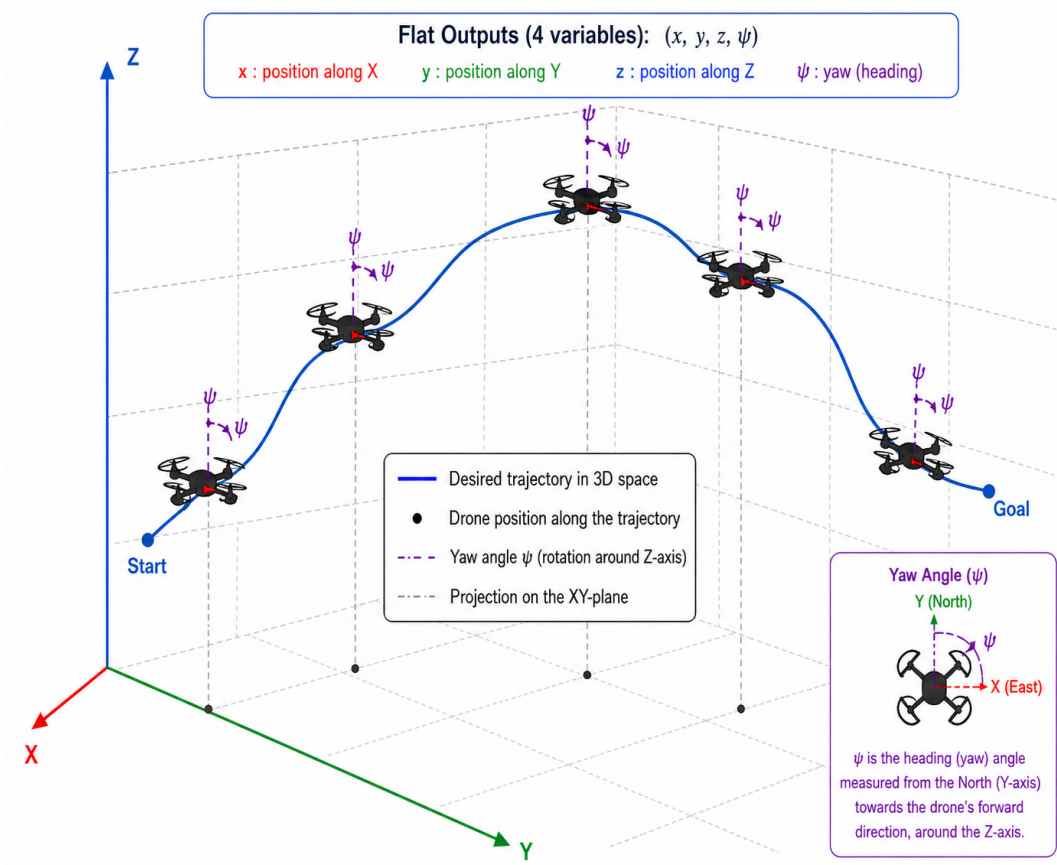


Figure 2.1 – Représentation géométrique des sorties plates (position 3D et angle de lacet) du quadrotor le long d’une trajectoire spatiale.

manière purement analytique.

Position et vitesse linéaire de référence :

$$\mathbf{p}_{ref} = [x, y, z]^T \quad (2.5)$$

$$\mathbf{v}_{ref} = \dot{\mathbf{p}}_{ref} = [\dot{x}, \dot{y}, \dot{z}]^T \quad (2.6)$$

Attitude de référence (Quaternion $\mathbf{q}_{ref}$) : En exploitant la dynamique de translation du quadrotor isolée dans le repère inertiel [7] :

$$m(\ddot{\mathbf{p}} + g\mathbf{e}_3) = F_b \cdot \mathbf{R}(\mathbf{q})\mathbf{e}_3 = F_b \cdot \mathbf{z}_B \quad (2.7)$$

Où m est la masse totale du quadrotor, g désigne l’accélération de la pesanteur, $\mathbf{e}_3 = [0, 0, 1]^T$ représente le vecteur unitaire vertical de la base inertielle, F_b correspond à la force de poussée totale générée par les rotors, $\mathbf{R}(\mathbf{q})$ est la fonction de rotation cinématique associée au quaternion $\mathbf{q}$, et $\mathbf{z}_B \in \mathbb{R}^3$ est l’axe vertical unitaire du repère corps $\mathcal{R}_B$.

De cette relation physique fondamentale, nous extrayons l’orientation de l’axe ver-

tical de référence $\mathbf{z}_{B_{ref}}$:

$$\mathbf{z}_{B_{ref}} = \frac{\ddot{\mathbf{p}}_{ref} + g\mathbf{e}_3}{\|\ddot{\mathbf{p}}_{ref} + g\mathbf{e}_3\|} \quad (2.8)$$

Afin de déterminer l'ensemble des axes du repère corps de référence, un vecteur directionnel intermédiaire $\mathbf{x}_C$ lié à l'angle de lacet planifié ψ est introduit :

$$\mathbf{x}_C = [\cos \psi, \sin \psi, 0]^T \quad (2.9)$$

Les axes restants de la base mobile de référence $(\mathbf{x}_{B_{ref}}, \mathbf{y}_{B_{ref}})$ sont alors calculés géométriquement par des opérations de produits vectoriels successifs [7] :

$$\mathbf{y}_{B_{ref}} = \frac{\mathbf{z}_{B_{ref}} \times \mathbf{x}_C}{\|\mathbf{z}_{B_{ref}} \times \mathbf{x}_C\|} \quad (2.10)$$

$$\mathbf{x}_{B_{ref}} = \mathbf{y}_{B_{ref}} \times \mathbf{z}_{B_{ref}} \quad (2.11)$$

Où $\times$ désigne l'opérateur du produit vectoriel. Pour exprimer ces axes de manière explicite via leurs composantes cartésiennes respectives, nous posons :

$$\mathbf{x}_{B_{ref}} = \begin{bmatrix} x_x \\ x_y \\ x_z \end{bmatrix}, \quad \mathbf{y}_{B_{ref}} = \begin{bmatrix} y_x \\ y_y \\ y_z \end{bmatrix}, \quad \mathbf{z}_{B_{ref}} = \begin{bmatrix} z_x \\ z_y \\ z_z \end{bmatrix} \quad (2.12)$$

En appliquant l'algorithme d'extraction directe dérivé de la méthode de Sheppard pour garantir une stabilité numérique absolue et éliminer tout risque de division par zéro, la partie scalaire $q_{0,ref}$ et la partie vectorielle du quaternion de référence $\mathbf{q}_{ref} = [q_{0,ref}, q_{1,ref}, q_{2,ref}, q_{3,ref}]^T$ sont formellement calculées par :

$$q_{0,ref} = \frac{1}{2} \sqrt{1 + x_x + y_y + z_z} \quad (2.13)$$

$$\begin{bmatrix} q_{1,ref} \\ q_{2,ref} \\ q_{3,ref} \end{bmatrix} = \frac{1}{4q_{0,ref}} \begin{bmatrix} y_z - z_y \\ z_x - x_z \\ x_y - y_x \end{bmatrix} \quad (2.14)$$

Cette formulation close permet d'alimenter directement le vecteur d'état de référence, restant ainsi en parfaite adéquation avec le formalisme par quaternions unitaires choisi pour ce travail.

Vitesse angulaire de référence ($\boldsymbol{\omega}_{ref}$) : En dérivant analytiquement les projections temporelles des axes du repère corps, on fait apparaître le jerk $\dot{\mathbf{a}} = \mathbf{p}^{(3)}$ et la vitesse de lacet $\dot{\psi}$. La vitesse angulaire s'obtient directement par la projection suivante :

$$\boldsymbol{\omega}_{ref} = [-\mathbf{y}_{B_{ref}}^T \cdot \mathbf{z}_{B_{ref}} \dot{\psi}, \mathbf{x}_{B_{ref}}^T \cdot \mathbf{z}_{B_{ref}} \dot{\psi}, \mathbf{y}_{B_{ref}}^T \cdot \mathbf{x}_{B_{ref}} \dot{\psi}]^T \quad (2.15)$$

Où $\dot{\mathbf{x}}_{B_{ref}}$ et $\dot{\mathbf{z}}_{B_{ref}}$ représentent les dérivées temporelles premières des axes géométriques de référence du repère corps.

2.1.4 Calcul des commandes nominales anticipées (Feedforward $\mathbf{u}_{ref}$)

Les entrées de commande nominales $\mathbf{u}_{ref} = [F_{b,ref}, \tau_x, \tau_y, \tau_z]^T \in \mathbb{R}^4$ définissent les efforts théoriques requis pour forcer le système à suivre idéalement la trajectoire planifiée en l'absence de perturbations. Leur calcul requiert la connaissance des dérivées de la position jusqu'à l'ordre 4 (*snap* $\mathbf{p}^{(4)}$) ainsi que l'accélération du lacet $\ddot{\psi}$ [8].

Poussée nominale de référence ($F_{b,ref}$) :

$$F_{b,ref} = m \|\ddot{\mathbf{p}}_{ref} + g\mathbf{e}_3\| \quad (2.16)$$

Couples nominaux de référence ($\boldsymbol{\tau}_{ref}$) : En inversant les équations d'Euler régissant la dynamique de rotation du quadrotor, nous obtenons :

$$\boldsymbol{\tau}_{ref} = [\tau_x, \tau_y, \tau_z]^T = \mathbf{J}\dot{\boldsymbol{\omega}}_{ref} + \boldsymbol{\omega}_{ref} \times (\mathbf{J}\boldsymbol{\omega}_{ref}) \quad (2.17)$$

Où $\boldsymbol{\tau}_{ref}$ est le vecteur des couples nominaux de commande (roulis τ_x , tangage τ_y , lacet τ_z), $\mathbf{J} \in \mathbb{R}^{3 \times 3}$ représente la matrice d'inertie principale du drone, et $\dot{\boldsymbol{\omega}}_{ref}$ désigne le vecteur d'accélération angulaire de référence, obtenu par dérivation analytique directe de $\boldsymbol{\omega}_{ref}$ (faisant intervenir le *snap* $\mathbf{p}^{(4)}$ et $\ddot{\psi}$).

Ce calcul algébrique direct de $\mathbf{x}_{ref}$ et $\mathbf{u}_{ref}$ décharge le contrôleur prédictif de l'obligation de générer en ligne une trajectoire d'attitude admissible. Le terme $\mathbf{u}_{ref}$ agit ainsi comme une action anticipée (*feedforward*), permettant au NMPC de focaliser son action sur le rejet des perturbations dynamiques et la compensation des incertitudes de modèle.

2.2 Formulation du Problème de Contrôle Optimal (OCP)

2.2.1 Principes fondamentaux de la Commande Prédictive Non Linéaire (NMPC) en horizon glissant

La Commande Prédictive Non Linéaire (NMPC) est une stratégie de commande avancée basée sur l'optimisation en temps réel d'un critère de performance sur un horizon temporel futur fini. Le principe fondamental repose sur le concept de l'horizon glissant (*receding horizon principle*). À chaque instant d'échantillonnage t_k , l'état actuel

du quadrotor $\mathbf{x}(t_k)$ est mesuré ou estimé. En utilisant le modèle dynamique non linéaire du système, le contrôleur résout en ligne un Problème de Contrôle Optimal (OCP) pour prédire le comportement futur du drone sur un horizon de prédiction déterminé [24].

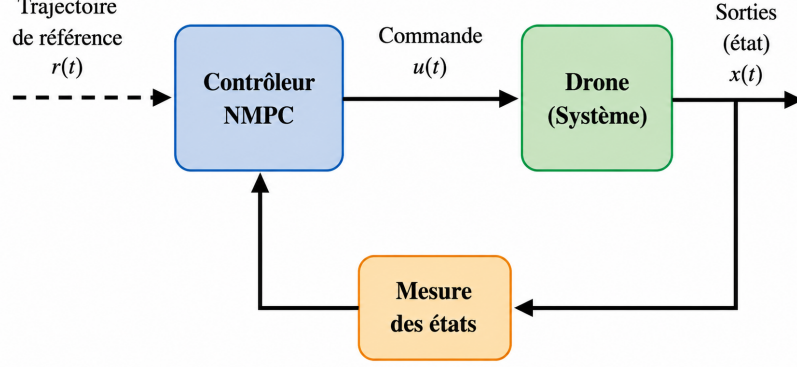


Figure 2.2 – Architecture générale du NMPC.

Ce calcul génère une séquence optimale d'entrées de commande sur cet horizon. Cependant, selon la philosophie de l'horizon glissant, seule la première commande de cette séquence est effectivement appliquée aux moteurs du quadrotor durant l'intervalle d'échantillonnage $[t_k, t_{k+1}]$ (Figure 2.3). Au pas de temps suivant t_{k+1} , l'horizon est décalé d'une unité, un nouveau problème d'optimisation est formulé à partir du nouvel état mesuré, et le processus se répète. Cette boucle fermée implicite confère au système une robustesse intrinsèque face aux incertitudes de modèle et aux perturbations extérieures [24].

Les paramètres temporels clés de cette formulation sont définis comme suit :

- T_s : Le pas d'échantillonnage temporel (en secondes), définissant la discrétisation du temps.
- N : L'horizon de prédiction (nombre de pas), fixant la fenêtre temporelle future $T_p = N \cdot T_s$ sur laquelle le système anticipe sa trajectoire.
- $\mathbf{x}_{k+i|k}$: Le vecteur d'état prédit au pas $k + i$, calculé à partir des informations disponibles au pas k .
- $\mathbf{u}_{k+i|k}$: Le vecteur de commande prédit à appliquer au pas $k + i$ [24].

2.2.2 Évaluation de l'erreur d'attitude sur $\text{SO}(3)$ via l'Application Logarithmique (Log Map)

Pour un quadrotor effectuant des manœuvres dynamiques à grands angles, l'utilisation des représentations classiques d'attitude pose des verrous mathématiques majeurs. Les angles d'Euler souffrent de singularités cinématiques (*Gimbal lock*), tandis que les quaternions, bien que sans singularité, souffrent du problème de double recouvrement

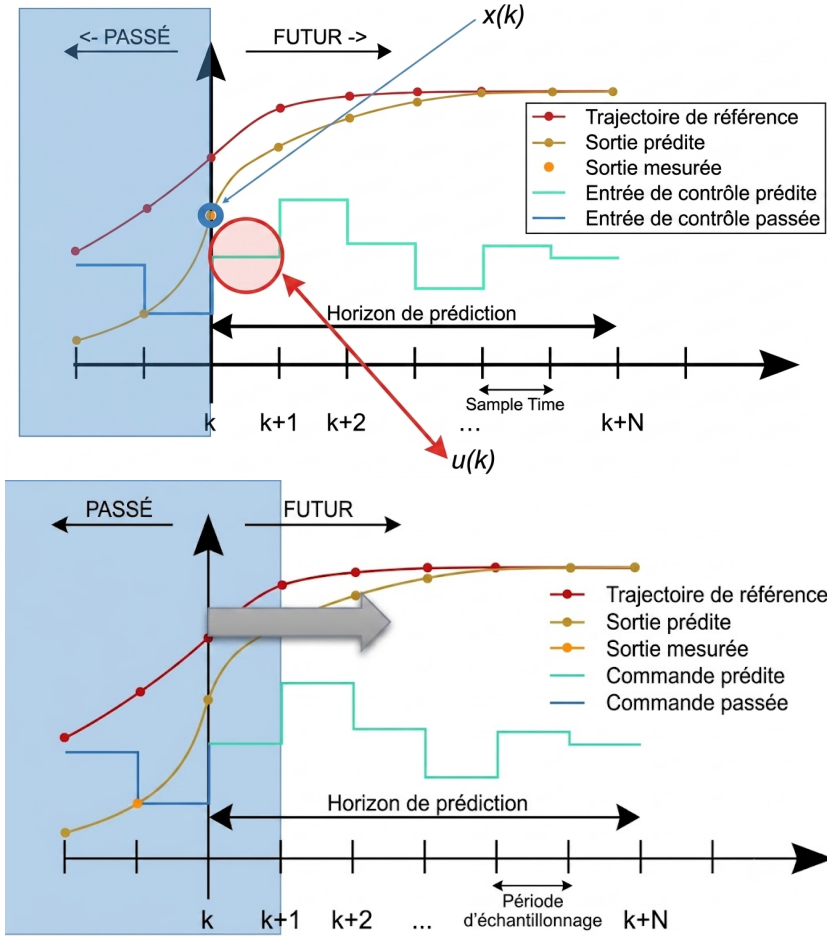


Figure 2.3 – Illustration du principe de la commande prédictive : glissement de l’horizon d’optimisation aux instants successifs k et $k + 1$.

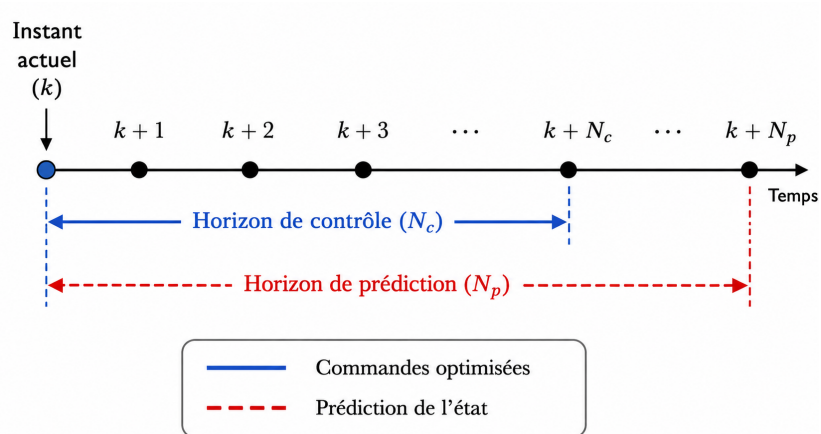


Figure 2.4 – Horizon de prédiction et de contrôle.

(les quaternions $\mathbf{q}$ et $-\mathbf{q}$ représentent la même orientation physique). Cette dualité peut induire des phénomènes de déroulement indésirables (*Unwinding*) dans un algorithme d’optimisation quadratique, forçant le drone à effectuer une rotation complète superflue [25].

Pour surmonter ces limitations tout en exploitant les propriétés globales des quaternions, nous évaluons l'erreur d'orientation en calculant le quaternion d'erreur, puis en le projetant vers l'espace tangent euclidien $\mathbb{R}^3$ via l'application logarithmique (*Log Map*) [26].

Soit $\mathbf{q} = [q_0, q_1, q_2, q_3]^T \in \mathbb{S}^3$ le quaternion unitaire représentant l'attitude actuelle du quadrotor, et $\mathbf{q}_{ref} \in \mathbb{S}^3$ le quaternion d'attitude de référence idéale issu de la différentielle plate. Nous définissons le quaternion d'erreur relative $\mathbf{q}_e = [q_{e,0}, q_{e,1}, q_{e,2}, q_{e,3}]^T \in \mathbb{S}^3$, représentant la rotation relative entre ces deux repères, par le produit de Hamilton :

$$\mathbf{q}_e = \mathbf{q}_{ref}^* \otimes \mathbf{q} \quad (2.18)$$

où $\mathbf{q}_{ref}^*$ désigne le conjugué du quaternion de référence. Pour extraire explicitement sa structure, nous isolons sa partie scalaire $q_{e,0}$ et sa partie vectorielle $\mathbf{q}_{e,v} = [q_{e,1}, q_{e,2}, q_{e,3}]^T \in \mathbb{R}^3$.

L'espace des quaternions unitaires étant une variété courbe, une simple différence algébrique n'a pas de sens géométrique. Afin de lever l'ambiguïté du double recouvrement et de garantir que le contrôleur choisisse toujours le chemin de rotation le plus court, une correction de signe est systématiquement appliquée : si la partie scalaire $q_{e,0} < 0$, le quaternion d'erreur est inversé tel que $\mathbf{q}_e \leftarrow -\mathbf{q}_e$.

L'application logarithmique permet d'aplatir cette erreur de la sphère $\mathbb{S}^3$ vers son espace tangent $\mathbb{R}^3$. Le vecteur d'erreur d'attitude géométrique tridimensionnel $\mathbf{e}_R \in \mathbb{R}^3$ est ainsi calculé directement par :

$$\mathbf{e}_R = 2\text{Log}(\mathbf{q}_e) \quad (2.19)$$

L'expression analytique de cette application logarithmique, corrigée pour éviter les singularités numériques, est donnée par :

$$\mathbf{e}_R = \begin{cases} \mathbf{0}_{3 \times 1} & \text{si } \|\mathbf{q}_{e,v}\| \leq \epsilon \\ 2 \arccos(q_{e,0}) \frac{\mathbf{q}_{e,v}}{\|\mathbf{q}_{e,v}\|} & \text{si } \|\mathbf{q}_{e,v}\| > \epsilon \end{cases} \quad (2.20)$$

Où :

- $q_{e,0}$ et $\mathbf{q}_{e,v}$: Représentent respectivement la partie scalaire et la partie vectorielle du quaternion d'erreur.
- $\arccos(q_{e,0})$: Représente la moitié de l'angle d'erreur de rotation total ($\theta/2$) autour de l'axe propre.
- $\frac{\mathbf{q}_{e,v}}{\|\mathbf{q}_{e,v}\|}$: Représente l'axe de rotation unitaire de l'erreur.
- ϵ : Une constante numérique très petite (10^{-6}) introduite pour éviter la singularité mathématique de division par zéro lorsque le drone est parfaitement aligné avec sa référence.

Le vecteur résultant $\mathbf{e}_R = [e_{R,x}, e_{R,y}, e_{R,z}]^T$ constitue un vecteur d'erreur d'attitude géométrique minimal isomorphe à l'espace des vecteurs de rotation, strictement exempt de singularité et d'ambiguïté.

2.2.3 Conception de la fonction de coût quadratique

L'objectif de performance du contrôleur NMPC est formulé mathématiquement à travers la minimisation d'une fonction de coût quadratique J . Cette fonction pénalise l'écart entre les états prédits et les états de référence, tout en minimisant l'effort énergétique des actionneurs et les variations brusques de commande pour éviter la fatigue mécanique des moteurs [27].

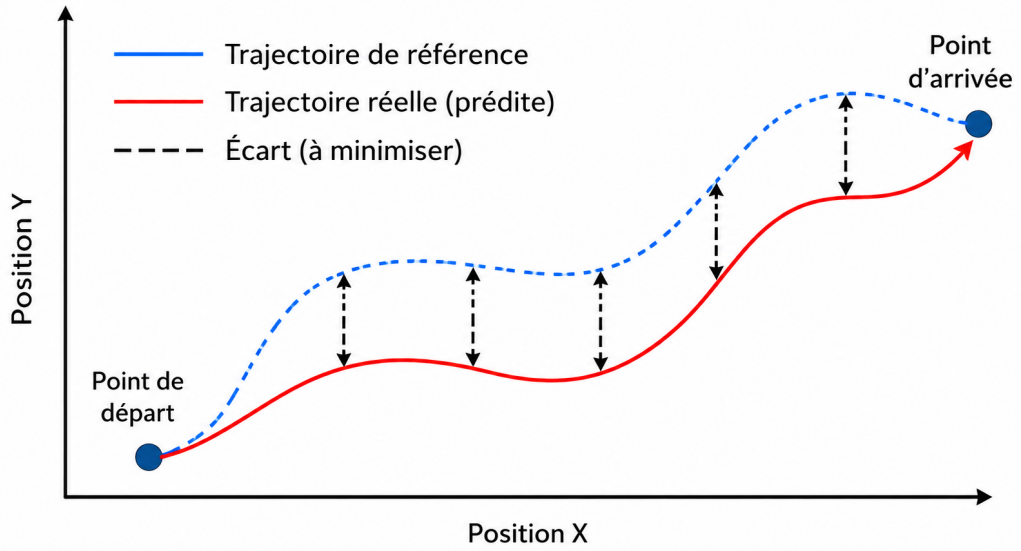


Figure 2.5 – Principe du problème OCP.

Au pas de temps k , la fonction de coût discrète J évaluée sur l'horizon de prédiction de longueur N est structurée selon la formulation standard suivante :

$$J = \sum_{i=0}^{N-1} \left(\mathbf{e}_{\mathbf{x},k+i|k}^T \mathbf{Q} \mathbf{e}_{\mathbf{x},k+i|k} + \mathbf{e}_{\mathbf{u},k+i|k}^T \mathbf{R}_m \mathbf{e}_{\mathbf{u},k+i|k} \right) + \mathbf{e}_{\mathbf{x},k+N|k}^T \mathbf{P} \mathbf{e}_{\mathbf{x},k+N|k} \quad (2.21)$$

Où les vecteurs d'erreur d'état $\mathbf{e}_{\mathbf{x}} \in \mathbb{R}^{12}$ et d'erreur de commande $\mathbf{e}_{\mathbf{u}} \in \mathbb{R}^4$ sont explicitement définis par :

$$\mathbf{e}_{\mathbf{x}} = [\mathbf{e}_{\mathbf{p}}^T, \mathbf{e}_{\mathbf{v}}^T, \mathbf{e}_R^T, \mathbf{e}_{\omega}^T]^T \in \mathbb{R}^{12} \quad (2.22)$$

$$\mathbf{e}_{\mathbf{u}} = \mathbf{u} - \mathbf{u}_{ref} \in \mathbb{R}^4 \quad (2.23)$$

La nomenclature des matrices de pondération et des variables de la fonction de coût est la suivante :

- $J \in \mathbb{R}$: La valeur scalaire du coût global à minimiser par le solveur d'optimisation.
- $\mathbf{e}_x \in \mathbb{R}^{12}$: Le vecteur d'erreur d'état généralisé (composé des vecteurs d'erreur de position $\mathbf{e}_p$, de vitesse linéaire $\mathbf{e}_v$, d'attitude géométrique par Log Map $\mathbf{e}_R$, et de vitesse angulaire $\mathbf{e}_\omega$).
- $\mathbf{e}_u \in \mathbb{R}^4$: Le vecteur d'écart de commande par rapport aux commandes nominales anticipées (*feedforward*).
- $\mathbf{Q} \in \mathbb{R}^{12 \times 12}$: La matrice de pondération des états intermédiaires. Elle est choisie diagonale et semi-définie positive ($\mathbf{Q} \succeq 0$). Ses coefficients permettent de régler l'importance relative du suivi de chaque variable.
- $\mathbf{R}_m \in \mathbb{R}^{4 \times 4}$: La matrice de pondération des commandes. Elle est diagonale et strictement définie positive ($\mathbf{R}_m \succ 0$). Ses coefficients pénalisent l'utilisation excessive de la force de poussée F_b et du vecteur des couples $\boldsymbol{\tau}$.
- $\mathbf{P} \in \mathbb{R}^{12 \times 12}$: La matrice de coût terminal, garantissant la stabilité asymptotique du système en boucle fermée à la fin de l'horizon de prédiction.

Grâce à cette formulation quadratique intégrant l'erreur géométrique vectorielle $\mathbf{e}_R$ issue du quaternion, le problème d'optimisation (OCP) reste lisse et localement convexe au voisinage de la trajectoire de référence, ce qui assure des temps de calcul rapides compatibles avec les exigences du contrôle en temps réel du quadrotor.

2.3 Intégration des Contraintes du Système

Pour que la commande générée par le contrôleur NMPC soit physiquement applicable et géométriquement cohérente, le Problème de Contrôle Optimal (OCP) doit être soumis à un ensemble de contraintes strictes, modélisant la dynamique du drone et les limites de ses actionneurs [28].

2.3.1 Contrainte de la dynamique interne (Discrétisation par RK4)

Le modèle dynamique du quadrotor est décrit par un système d'équations différentielles ordinaires (EDO) non linéaires en temps continu de la forme [28] :

$$\dot{\mathbf{x}}(t) = \mathbf{f}_c(\mathbf{x}(t), \mathbf{u}(t)) \quad (2.24)$$

Cependant, le solveur d'optimisation du contrôleur NMPC opère en temps discret sur un horizon fini. Pour garantir une haute précision numérique lors de la prédiction des états, nous utilisons la méthode de Runge-Kutta d'ordre 4 (RK4). À chaque intervalle d'échantillonnage T_s , et pour chaque pas i de l'horizon de prédiction, le passage de

l'état prédit $\mathbf{x}_{k+i|k}$ à l'état suivant $\mathbf{x}_{k+i+1|k}$ est évalué par quatre incréments vectoriels intermédiaires [24] :

$$\begin{cases} \mathbf{k}_1 = \mathbf{f}_c(\mathbf{x}_{k+i|k}, \mathbf{u}_{k+i|k}) \\ \mathbf{k}_2 = \mathbf{f}_c(\mathbf{x}_{k+i|k} + \frac{T_s}{2}\mathbf{k}_1, \mathbf{u}_{k+i|k}) \\ \mathbf{k}_3 = \mathbf{f}_c(\mathbf{x}_{k+i|k} + \frac{T_s}{2}\mathbf{k}_2, \mathbf{u}_{k+i|k}) \\ \mathbf{k}_4 = \mathbf{f}_c(\mathbf{x}_{k+i|k} + T_s\mathbf{k}_3, \mathbf{u}_{k+i|k}) \end{cases} \quad (2.25)$$

L'état futur discrétisé est alors calculé par la moyenne pondérée de ces incréments, définissant l'application de transition discrète $\mathbf{f}_d$:

$$\mathbf{x}_{k+i+1|k} = \mathbf{x}_{k+i|k} + \frac{T_s}{6}(\mathbf{k}_1 + 2\mathbf{k}_2 + 2\mathbf{k}_3 + \mathbf{k}_4) \equiv \mathbf{f}_d(\mathbf{x}_{k+i|k}, \mathbf{u}_{k+i|k}) \quad (2.26)$$

Dans le problème d'optimisation OCP, cette relation se traduit par une contrainte d'égalité vectorielle stricte (méthode de tir multiple), garantissant que la trajectoire prédite respecte rigoureusement le modèle physique à chaque pas de la fenêtre de prédiction :

$$\mathbf{x}_{k+i+1|k} - \mathbf{f}_d(\mathbf{x}_{k+i|k}, \mathbf{u}_{k+i|k}) = \mathbf{0}_{13 \times 1}, \quad \forall i \in \{0, \dots, N-1\} \quad (2.27)$$

2.3.2 Contraintes d'inégalité des actionneurs (Limites des efforts physiques)

Dans un système réel, la force de poussée aérodynamique et les couples de torsion directionnels sont bornés par les capacités physiques maximales du système motopropulseur. Il est donc impératif d'imposer des contraintes d'inégalité saturantes au vecteur de commande au sein de l'OCP [28] :

$$\mathbf{u}_{\min} \leq \mathbf{u}_{k+i|k} \leq \mathbf{u}_{\max}, \quad \forall i \in \{0, \dots, N-1\} \quad (2.28)$$

Où les vecteurs limites $\mathbf{u}_{\min} = [F_{b,\min}, \tau_{x,\min}, \tau_{y,\min}, \tau_{z,\min}]^T$ et $\mathbf{u}_{\max} = [F_{b,\max}, \tau_{x,\max}, \tau_{y,\max}, \tau_{z,\max}]^T$ représentent les frontières physiques admissibles. Ces contraintes garantissent la faisabilité des solutions calculées en ligne et empêchent toute commande irréalisable par les actionneurs [24].

2.3.3 Contrainte géométrique de normalisation stricte du quaternion

Pour que le quaternion d'attitude prédit $\mathbf{q}_{k+i|k} = [q_0, q_1, q_2, q_3]_{k+i|k}^T$ représente une orientation physique valide appartenant à la variété sphérique $\mathbb{S}^3$, il doit impérativement respecter la propriété mathématique d'unitaire [28]. En raison des approxima-

tions inhérentes à l'intégration numérique répétée du RK4, la norme du quaternion peut s'écarter de l'unité, dégradant la cohérence géométrique du modèle.

Pour pallier cette dérive et imposer un respect absolu de la structure algébrique de l'attitude, nous formulons cette propriété géométrique comme une contrainte d'égalité quadratique directement intégrée à l'OCP pour chaque pas de temps :

$$\|\mathbf{q}_{k+i|k}\|^2 - 1 = 0, \quad \forall i \in \{1, \dots, N\} \quad (2.29)$$

Cette formulation force le solveur d'optimisation à projeter et à maintenir la trajectoire d'attitude sur la variété $SO(3)$, éliminant tout besoin de normalisation artificielle a posteriori.

2.4 Modélisation Mathématique pour l'Évitement d'Obstacle

2.4.1 Représentation géométrique d'un obstacle sphérique statique

Pour garantir l'efficacité algorithmique et réduire le temps de calcul lors de l'optimisation en ligne, les obstacles tridimensionnels présents dans l'environnement sont modélisés par des volumes englobants simplifiés. Dans ce travail, chaque obstacle statique est représenté par une sphère géométrique caractérisée par [29] :

- Un vecteur de position centrale fixe : $\mathbf{p}_{\text{obs}} = [x_{\text{obs}}, y_{\text{obs}}, z_{\text{obs}}]^T \in \mathbb{R}^3$.
- Un rayon physique minimal : $R_{\text{obs}} > 0$.

Afin de prendre en compte l'encombrement spatial et les dimensions physiques du quadrotor, ce dernier est encapsulé virtuellement dans une sphère de sécurité de rayon R_{drone} . Par conséquent, la distance de sécurité minimale admissible, notée R_{safe} , est définie par la relation suivante :

$$R_{\text{safe}} = R_{\text{obs}} + R_{\text{drone}} + \epsilon_s \quad (2.30)$$

Où $\epsilon_s > 0$ représente une marge de tolérance ou de sécurité supplémentaire face aux incertitudes de suivi de trajectoire.

2.4.2 Formulation de la contrainte d'inégalité de distance euclidienne

La mission d'évitement d'obstacle est mathématiquement formulée en imposant que la distance euclidienne séparant le centre de masse du drone et le centre de la sphère de

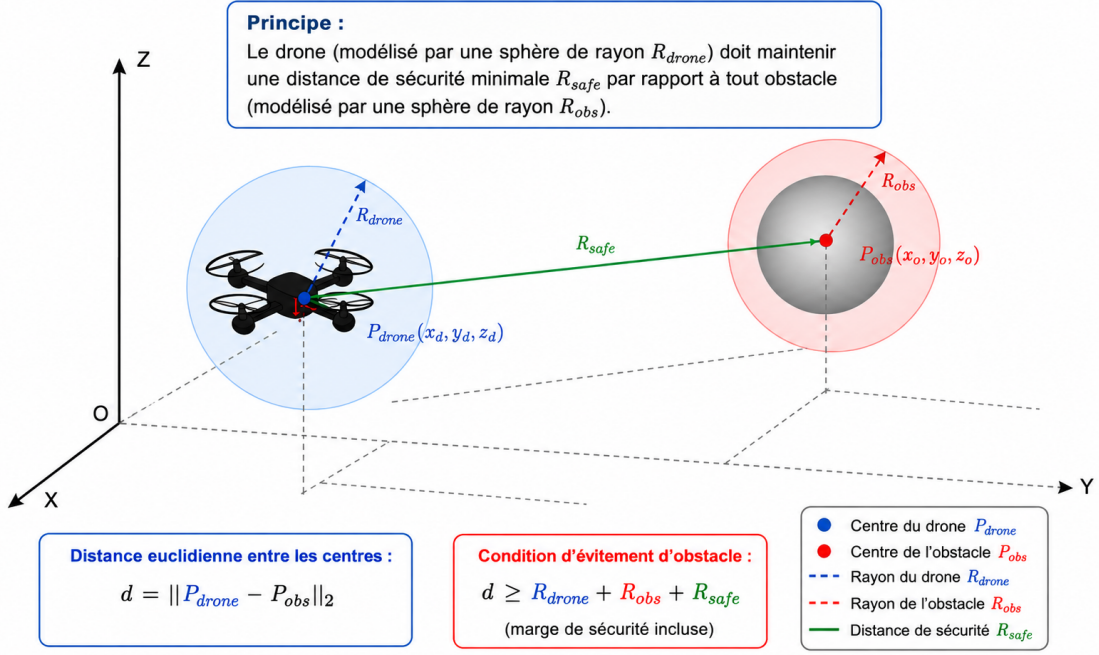


Figure 2.6 – Modélisation géométrique de l'évitement d'obstacle par la méthode des sphères englobantes.

l'obstacle demeure strictement supérieure ou égale au rayon de sécurité R_{safe} à chaque pas de prédiction [29].

Soit $\mathbf{p}_{k+i|k} = [x_{k+i|k}, y_{k+i|k}, z_{k+i|k}]^T \in \mathbb{R}^3$ le vecteur de position prédit du drone à l'horizon i , extrait directement des trois premières composantes du vecteur d'état global $\mathbf{x}_{k+i|k}$. La contrainte géométrique d'exclusion s'exprime par :

$$\|\mathbf{p}_{k+i|k} - \mathbf{p}_{obs}\|^2 \geq R_{safe}^2, \quad \forall i \in \{1, \dots, N\} \quad (2.31)$$

D'un point de vue numérique, l'évaluation directe de la norme euclidienne implique l'usage d'une racine carrée, une opération non linéaire coûteuse et potentiellement non différentiable en zéro pour le solveur d'optimisation. Pour s'en affranchir, la contrainte est implémentée sous sa forme quadratique développée [30]. Elle est intégrée dans l'OCP global sous l'aspect d'une contrainte d'inégalité algébrique $\mathcal{C}_{obs}(\mathbf{x}_{k+i|k}) \geq 0$:

$$\mathcal{C}_{obs}(\mathbf{x}_{k+i|k}) = (x_{k+i|k} - x_{obs})^2 + (y_{k+i|k} - y_{obs})^2 + (z_{k+i|k} - z_{obs})^2 - R_{safe}^2 \geq 0, \quad \forall i \in \{1, \dots, N\} \quad (2.32)$$

Cette inégalité informe dynamiquement l'algorithme NMPC que l'espace délimité par le rayon de sécurité de la sphère constitue une zone interdite (*Keep-Out Zone*), forçant le solveur à modifier la séquence des commandes $\mathbf{U}_k$ pour contourner l'obstacle.

2.5 Formulation Globale du Problème de Contrôle Optimal (OCP)

Afin de synthétiser la stratégie de commande prédictive non linéaire (NMPC) développée, cette section regroupe le critère de performance (fonction de coût) ainsi que l'ensemble des contraintes physiques, dynamiques et géométriques explicitées précédemment.

À chaque instant d'échantillonnage T_s , le contrôleur résout en ligne un Problème de Contrôle Optimal (OCP) sur un horizon de prédiction fini N . Ce problème d'optimisation sous contraintes s'exprime de manière compacte sous la forme suivante :

$$\min_{\mathbf{U}_k, \mathbf{X}_k} J(\mathbf{x}_k, \mathbf{U}_k) = \sum_{i=0}^{N-1} L(\mathbf{e}_{x,k+i|k}, \mathbf{e}_{u,k+i|k}) + V_f(\mathbf{e}_{x,k+N|k}) \quad (2.33a)$$

$$\text{s.c. } \mathbf{x}_{k|k} = \mathbf{x}(t_k), \quad (2.33b)$$

$$\mathbf{x}_{k+i+1|k} = \mathbf{f}_d(\mathbf{x}_{k+i|k}, \mathbf{u}_{k+i|k}), \quad \forall i \in \{0, \dots, N-1\}, \quad (2.33c)$$

$$\mathbf{u}_{\min} \leq \mathbf{u}_{k+i|k} \leq \mathbf{u}_{\max}, \quad \forall i \in \{0, \dots, N-1\}, \quad (2.33d)$$

$$\|\mathbf{q}_{k+i|k}\|^2 - 1 = 0, \quad \forall i \in \{1, \dots, N\}, \quad (2.33e)$$

$$\mathcal{C}_{\text{obs}}(\mathbf{x}_{k+i|k}) \geq 0, \quad \forall i \in \{1, \dots, N\}. \quad (2.33f)$$

Où la **fonction de coût d'étape** (ou *running function*) $L(\mathbf{e}_x, \mathbf{e}_u)$ et la **fonction de coût terminale** $V_f(\mathbf{e}_x)$ respectent la fonction J définie précédemment :

$$\begin{aligned} L(\mathbf{e}_{x,k+i|k}, \mathbf{e}_{u,k+i|k}) &= \mathbf{e}_{x,k+i|k}^T \mathbf{Q} \mathbf{e}_{x,k+i|k} + \mathbf{e}_{u,k+i|k}^T \mathbf{R}_m \mathbf{e}_{u,k+i|k} \\ V_f(\mathbf{e}_{x,k+N|k}) &= \mathbf{e}_{x,k+N|k}^T \mathbf{P} \mathbf{e}_{x,k+N|k} \end{aligned}$$

Dans cette formulation :

- L'équation (2.33a) représente la fonction de coût totale J à minimiser.
- (2.33b) est la condition initiale basée sur l'état réel actuel mesuré ou estimé $\mathbf{x}(t_k)$.
- (2.33c) représente la contrainte d'égalité dynamique discrète issue de l'intégration numérique par la méthode RK4.
- (2.33d) et (2.33e) matérialisent respectivement les contraintes d'inégalités de saturation des actionneurs et la contrainte géométrique de normalisation stricte du quaternion d'attitude.
- (2.33f) correspond à l'incorporation de la contrainte d'évitement d'obstacle par sphère englobante développée précédemment.

Les variables et structures de pondération intervenant dans le problème (2.33) sont explicitées comme suit :

- $\mathbf{U}_k = \{\mathbf{u}_{k|k}, \mathbf{u}_{k+1|k}, \dots, \mathbf{u}_{k+N-1|k}\} \in \mathbb{R}^{4 \times N}$ est la séquence des vecteurs de commande optimisés sur l'horizon, constituant la variable de décision principale du solveur.
- $\mathbf{X}_k = \{\mathbf{x}_{k|k}, \mathbf{x}_{k+1|k}, \dots, \mathbf{x}_{k+N|k}\} \in \mathbb{R}^{13 \times (N+1)}$ est la trajectoire des états prédits par le modèle, où chaque état individuel $\mathbf{x} \in \mathbb{R}^{13}$ inclut la représentation par quaternion.
- $\mathbf{e}_x = \mathbf{x} \boxminus \mathbf{x}^{\text{ref}} \in \mathbb{R}^{12}$ et $\mathbf{e}_u = \mathbf{u} - \mathbf{u}^{\text{ref}} \in \mathbb{R}^4$ sont les vecteurs d'erreur calculés par rapport aux trajectoires de référence issues du planificateur. L'écart d'attitude au sein de $\mathbf{e}_x$ est réduit à 3 dimensions via l'application logarithmique.
- $\mathbf{Q} \in \mathbb{R}^{12 \times 12}$ et $\mathbf{P} \in \mathbb{R}^{12 \times 12}$ sont les matrices de pondération diagonales semi-définies positives ($\mathbf{Q}, \mathbf{P} \succeq 0$) associées aux erreurs d'état intermédiaires et terminales.
- $\mathbf{R}_m \in \mathbb{R}^{4 \times 4}$ est la matrice de pondération diagonale strictement définie positive ($\mathbf{R}_m \succ 0$) pénalisant l'amplitude de l'effort de commande.

Une fois le problème d'optimisation (2.33) résolu en ligne à l'instant t_k , seule la première commande optimale de la séquence, notée $\mathbf{u}_{k|k}^*$, est effectivement appliquée aux moteurs du quadrotor durant l'intervalle $[t_k, t_{k+1}]$. Au pas de temps suivant t_{k+1} , l'horizon est décalé d'une unité, une nouvelle mesure de l'état $\mathbf{x}(t_{k+1})$ est acquise, et l'ensemble du processus est réitéré selon le principe de l'horizon glissant.

Conclusion

Dans ce chapitre, nous avons développé l'intégralité de l'architecture de commande NMPC dédiée au quadrotor. En commençant par la génération de trajectoire basée sur la **platitude différentielle**, nous avons garanti la production de références dynamiquement admissibles pour le système. Ensuite, la formulation du problème de contrôle optimal (OCP) a été rigoureusement établie en exploitant la géométrie des **quaternions unitaires** ($\mathbb{S}^3$), permettant une gestion élégante et sans singularité de l'attitude. L'intégration minutieuse des contraintes liées à la dynamique du drone (discretisée par la méthode RK4), aux limitations physiques des actionneurs, ainsi qu'au respect strict de la norme du quaternion assure la viabilité physique de la commande calculée. Enfin, la modélisation mathématique de la distance euclidienne a permis d'incorporer une contrainte d'inégalité algébrique, dotant le quadrotor d'une **capacité d'évitement d'obstacles en temps réel**. Le chapitre suivant sera consacré à l'implémentation numérique de cette architecture de commande sous environnement de simulation, en s'appuyant sur des outils d'optimisation numérique avancés.

Chapitre 3

Simulation Numérique et Analyse des Performances

Dans les chapitres précédents, les modèles mathématiques régissant la dynamique du quadrotor ainsi que la formulation théorique du contrôleur prédictif non linéaire (NMPC) ont été rigoureusement développés. Ce troisième chapitre est dédié à l'implémentation de ces formulations au sein d'un environnement de simulation numérique sous MATLAB, afin d'évaluer quantitativement et qualitativement les performances du système dans des conditions de vol réalistes.

Après une présentation détaillée des outils de résolution numérique et des paramètres de configuration de la simulation, plusieurs scénarios d'évaluation seront étudiés pour analyser le comportement dynamique du drone :

- **Le premier scénario** concerne le suivi de trajectoires de référence complexes en conditions nominales, permettant de valider l'exactitude du couplage entre la planéité différentielle et la commande prédictive.
- **Le deuxième scénario** introduit des perturbations exogènes et endogènes, telles que l'action de rafales de vent ou des variations subites de la masse embarquée, afin d'éprouver la robustesse intrinsèque du contrôleur.
- **Enfin, le troisième scénario** porte sur la navigation en environnement contraint avec évitement d'obstacles statiques, vérifiant la capacité du NMPC à garantir la sécurité absolue du système tout en maintenant un suivi optimal.

Les résultats graphiques et statistiques obtenus permettront de démontrer l'efficacité de la stratégie de commande proposée en termes de précision, de robustesse face aux incertitudes et de respect strict des contraintes physiques imposées aux actionneurs.

3.1 Environnement de Résolution Numérique

3.1.1 Le framework logiciel CasADi pour la transcription de l'OCP

La mise en œuvre pratique d'une stratégie NMPC requiert la transformation du problème de contrôle optimal (OCP) continu en un problème d'optimisation non linéaire (NLP) de dimension finie, une étape cruciale appelée transcription directe. Pour accomplir cette tâche complexe, le framework logiciel **CasADi** a été exploité.

CasADi est un environnement open-source de pointe dédié à l'optimisation numérique et au contrôle optimal des systèmes dynamiques non linéaires [31]. Sa force majeure réside dans ses outils de *différentiation automatique* algorithmique, qui permettent le calcul exact, rapide et sans erreur analytique des gradients, des matrices jacobiennes et hessiennes requis par les algorithmes de programmation non linéaire. Grâce à son interface transparente avec MATLAB et Python, CasADi facilite la formulation symbolique de l'architecture NMPC et accélère considérablement le temps de calcul, ce qui en fait un standard incontournable dans les domaines de la robotique aérienne et du contrôle avancé.

3.1.2 Le solveur IPOPT basé sur les méthodes de points intérieurs

Une fois le problème de contrôle optimal discrétisé et transcrit sous forme de programme non linéaire (NLP) par CasADi, sa résolution numérique en temps réel est confiée au solveur de grande envergure **IPOPT** (*Interior Point OPTimizer*).

IPOPT est un progiciel open-source hautement optimisé, conçu spécifiquement pour résoudre les problèmes d'optimisation non linéaire de grande dimension soumis à des contraintes d'égalité et d'inégalité directes [32]. Le cœur algorithmique d'IPOPT repose sur une méthode primal-dual de points intérieurs (barrière logarithmique), couplée à une stratégie de recherche linéaire de type filtre (*filter line-search*) pour garantir la convergence globale. En exploitant les dérivées exactes du premier et du second ordre fournies en amont par CasADi, IPOPT parvient à converger vers l'optimum local avec une vitesse de calcul remarquable à chaque pas d'échantillonnage, assurant ainsi la viabilité temporelle du contrôleur embarqué.

3.2 Configuration de la Simulation

3.2.1 Paramètres physiques nominaux du quadrotor (Masse m_0 , Inertie J_0)

Pour garantir la fidélité de la simulation par rapport à la dynamique intrinsèque d'un drone réel, le modèle mathématique a été paramétré avec les constantes physiques, géométriques et aérodynamiques résumées dans le Tableau 3.1. Les limites opérationnelles des actionneurs (poussée maximale et couples maximaux) y sont également intégrées, agissant comme des contraintes physiques directes au sein du solveur d'optimisation.

Table 3.1 – Paramètres physiques nominaux et limites saturantes du quadrotor

Paramètre	Symbole	Valeur	Unité
Masse totale nominale	m_0	0.486	kg
Accélération gravitationnelle	g	9.806	m/s ²
Longueur du bras	l	0.25	m
Inertie nominale selon l'axe X	I_{x0}	3.8278×10^{-3}	kg·m ²
Inertie nominale selon l'axe Y	I_{y0}	3.8288×10^{-3}	kg·m ²
Inertie nominale selon l'axe Z	I_{z0}	7.6566×10^{-3}	kg·m ²
Coefficient de poussée	b	2.9842×10^{-5}	N·s ²
Coefficient de traînée	d	3.232×10^{-7}	N·m·s ²
Limites de la poussée totale	T_{min}, T_{max}	[0, 14.357]	N
Limites de couple (Roulis/Tangage)	$\tau_{x,y}^{max}$	± 1.6526	N·m
Limites de couple (Lacet)	τ_z^{max}	± 0.9008	N·m

3.2.2 Réglages du NMPC

La synthèse et la performance en temps réel du contrôleur prédictif non linéaire (NMPC) reposent fondamentalement sur le paramétrage rigoureux de sa structure d'optimisation (*Tuning*). À chaque pas d'échantillonnage dt , le contrôleur résout un Problème de Contrôle Optimal (OCP) à horizon fuyant. En combinant les états dynamiques planifiés, l'OCP s'énonce par la minimisation de la fonction de coût quadratique globale suivante [33, 34] :

$$\min_{\mathbf{U}, \mathbf{X}} \sum_{k=0}^N \left(\|\mathbf{p}_k - \mathbf{p}_{d,k}\|_{\mathbf{Q}_p}^2 + \|\mathbf{v}_k - \mathbf{v}_{d,k}\|_{\mathbf{Q}_v}^2 + \|\mathbf{e}_{R,k}\|_{\mathbf{Q}_R}^2 + \|\omega_k - \omega_{d,k}\|_{\mathbf{Q}_\omega}^2 \right) + \sum_{k=0}^{N-1} \|\mathbf{u}_k\|_{\mathbf{R}_u}^2 \quad (3.1)$$

Dans cette formulation, l'horizon de prédiction a été calibré à $N = 15$ pas avec un pas d'échantillonnage de $dt = 0.1$ s, définissant ainsi une fenêtre de prédiction

temporelle de 1.5 seconde. Ce choix algorithmique offre un compromis optimal entre la capacité d’anticipation des courbures géométriques et le coût computationnel requis par le solveur IPOPT pour une exécution déterministe en temps réel.

Les matrices de pondération diagonales semi-définies positives ($\mathbf{Q}_p, \mathbf{Q}_v, \mathbf{Q}_R, \mathbf{Q}_\omega$) et définies positives ($\mathbf{R}_u$) ont été syntonisées par une méthode itérative d’essais et erreurs (tâtonnements) afin de privilégier la précision géométrique globale du suivi tout en pénalisant les sollicitations énergétiques à haute fréquence des actionneurs. Les valeurs numériques retenues sont explicitées dans le Tableau 3.2.

Table 3.2 – Paramètres de réglage et matrices de pondération du contrôleur NMPC

Paramètre NMPC	Symbole	Configuration numérique
Horizon de prédiction	N	15
Pas d’échantillonnage	dt	0.1 s
Poids sur la position (x, y, z)	$\mathbf{Q}_p$	diag(250, 250, 400)
Poids sur la vitesse (v_x, v_y, v_z)	$\mathbf{Q}_v$	diag(20, 20, 50)
Poids sur l’attitude (Log Map)	$\mathbf{Q}_R$	diag(50, 50, 20)
Poids sur la vitesse angulaire (p, q, r)	$\mathbf{Q}_\omega$	diag(10, 10, 1)
Poids sur l’actionnement $(T, \tau_x, \tau_y, \tau_z)$	$\mathbf{R}_u$	diag(0.1, 0.4, 0.4, 0.5)

Remarque : Le choix de l’horizon de prédiction $N = 15$ (associé à un pas d’échantillonnage $dt = 0.1$ s) définit une fenêtre de prédiction temporelle de 1.5 seconde ($T_p = N \times dt$). Ce choix représente un compromis optimal : il est suffisamment grand pour permettre au contrôleur d’anticiper les variations de la trajectoire et les obstacles (bonne vision future), tout en restant suffisamment petit pour garantir que le temps de calcul du solveur IPOPT ne dépasse pas la contrainte du temps réel (temps de calcul inférieur à dt).

3.2.3 Présentation des profils de trajectoires cibles

Afin de valider l’autorité de commande, le découplage des axes et la robustesse du NMPC développé, deux profils de trajectoires tridimensionnelles distincts et hautement exigeants ont été implémentés.

Trajectoire 1 : Profil Sinusoidal/Exp (Cercle adouci)

Ce premier profil est conçu pour évaluer l’aptitude du système à s’ancrer sur une orbite périodique tout en gérant une phase d’initialisation stabilisée. Le vecteur de position désirée $\mathbf{p}_d(t) = [x_d(t), y_d(t), z_d(t)]^T \in \mathbb{R}^3$ est régi par le système cinématique suivant :

$$\mathbf{p}_d(t) = \begin{cases} x_d(t) = 0.8 \sin(\omega_n t) (1 - e^{-0.1t^3}) \\ y_d(t) = 0.8 \cos(\omega_n t) (1 - e^{-0.1t^3}) \\ z_d(t) = 0.9 + 0.2 \sin(\omega_n t) \end{cases} \quad (3.2)$$

où $\omega_n = \frac{2\pi}{10}$ rad/s représente la pulsation nominale du mouvement planifié.

L'introduction du terme multiplicatif d'atténuation exponentielle $(1 - e^{-0.1t^3})$ sur les axes horizontaux permet de générer une transition dynamique douce qui élimine toute discontinuité initiale de type échelon. Sans cette précaution mathématique, le solveur d'optimisation ferait face à des erreurs initiales massives induisant des saturations immédiates des actionneurs [7].

La consigne de cap $\psi_{ref}(t)$ est générée de façon découplée via un polynôme de transition d'ordre 5 durant une phase transitoire de 5 secondes pour garantir une continuité de classe $\mathcal{C}^2$ [35] :

$$\psi_{ref}(t) = \frac{\pi}{3} (10\tau^3 - 15\tau^4 + 6\tau^5) + \psi_0 \quad (3.3)$$

où $\tau = \frac{\min(t,5)}{5}$ désigne le temps normalisé.

Trajectoire 2 : Profil Hélicoïdal (Cylindre)

Le second profil consiste en une trajectoire cylindrique ascendante pure, exploitée pour analyser les performances du contrôleur sous de fortes accélérations centripètes directionnelles variables. Les équations de guidage spatial s'articulent selon le modèle :

$$\mathbf{p}_{cyl}(t) = \begin{cases} x_c(t) = R_c \cos(\omega_c t) \\ y_c(t) = R_c \sin(\omega_c t) \\ z_c(t) = v_z t + z_0 \end{cases} \quad (3.4)$$

où $R = 1.0$ m désigne le rayon du cylindre, $\omega_n = \frac{2\pi}{10}$ rad/s la pulsation du mouvement circulaire (soit un tour complet en 10 s), $v_z = 0.1$ m/s la vitesse d'ascension constante, et $z_0 = 0.5$ m la hauteur initiale de départ.

Contrairement au premier profil où le lacet évoluait de façon découplée, la consigne de cap $\psi_{ref}(t)$ est ici calculée dynamiquement afin que le vecteur d'orientation du drone reste en permanence tangentiel à sa trajectoire géométrique (le drone fait face à sa direction de déplacement) :

$$\psi_{ref}(t) = \text{atan2}(\dot{y}_c(t), \dot{x}_c(t)) \quad (3.5)$$

Ce profil met à l'épreuve la capacité du NMPC à maintenir le découplage effectif entre la force de poussée totale et les couples de contrôle lors d'une inversion cyclique

continue des angles de roulis et de tangage, conjuguée à une variation temporelle permanente de l'objectif de lacet.

3.3 Scénario 1 : Validation du Suivi de Trajectoire en Conditions Nominales

Ce premier scénario évalue les performances intrinsèques du contrôleur NMPC quaternionique dans un environnement idéal, exempt de perturbations exogènes et d'incertitudes paramétriques, permettant de valider l'approche mathématique couplée à la planéité différentielle (*Differential Flatness*).

3.3.1 Analyse du comportement spatial (Trajectoire 3D)

L'analyse de l'évolution tridimensionnelle du quadrotor (Figure 3.1) permet de vérifier graphiquement la précision géométrique globale :

- **Profil Circulaire** : Le quadrotor décolle et rejoint l'orbite plane désirée sans dépassement. La trajectoire réelle se superpose fidèlement à la consigne, démontrant l'aptitude du NMPC à gérer l'accélération centripète continue [7].
- **Profil Cylindrique** : Le drone suit l'hélice géométrique en maintenant un rayon constant tout en effectuant son ascension verticale, illustrant l'efficacité de la prédiction temporelle face aux couplages tridimensionnels [36].

3.3.2 Évolution temporelle des variables cinématiques et d'état

L'examen des variables d'état du quadrotor (Figure 3.2) met en évidence la rapidité d'action du système. Après une brève phase transitoire d'accrochage due à l'écart initial, les trajectoires de position (x, y, z) se superposent parfaitement à la référence. Les profils de vitesse sinusoïdaux (v_x, v_y) requis pour le mouvement circulaire sont traqués sans aucun retard de phase, confirmant l'exactitude du modèle prédictif interne [15].

3.3.3 Évolution de l'erreur d'attitude (Log Map) et stabilité des quaternions

C'est au niveau de la dynamique rotationnelle que l'apport de la formulation quaternionique développée se démarque explicitement. L'espace des rotations $SO(3)$ n'étant pas un espace vectoriel linéaire mais une variété différentielle courbe modélisée par la sphère unité S^3 [37], effectuer une soustraction algébrique simple entre le quaternion actuel $\mathbf{q}$ et la référence $\mathbf{q}_d$ violerait immédiatement la contrainte de normalisation ($\|\mathbf{q}\| = 1$).

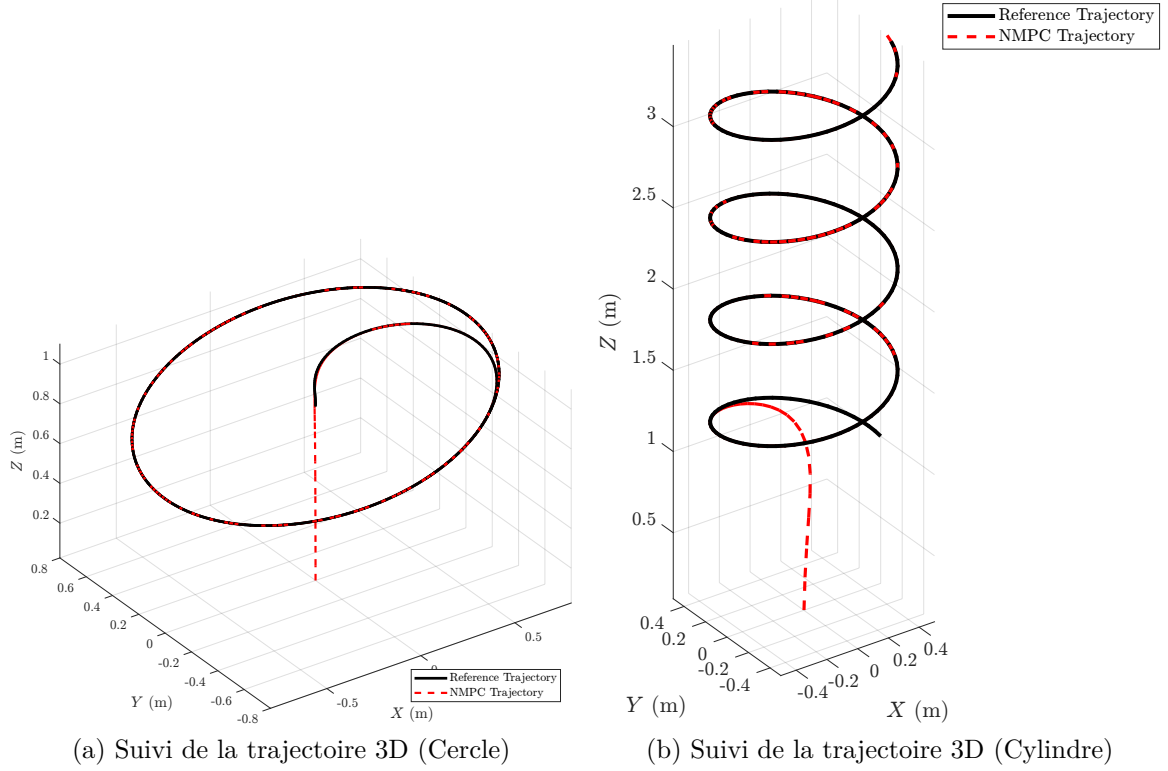


Figure 3.1 – Réponse spatiale tridimensionnelle du quadrotor en conditions nominales (Scénario 1).

Pour quantifier l’erreur d’orientation de manière rigoureuse, nous calculons d’abord l’erreur quaternionique relative $\mathbf{q}_e$ via le produit de Hamilton [38] :

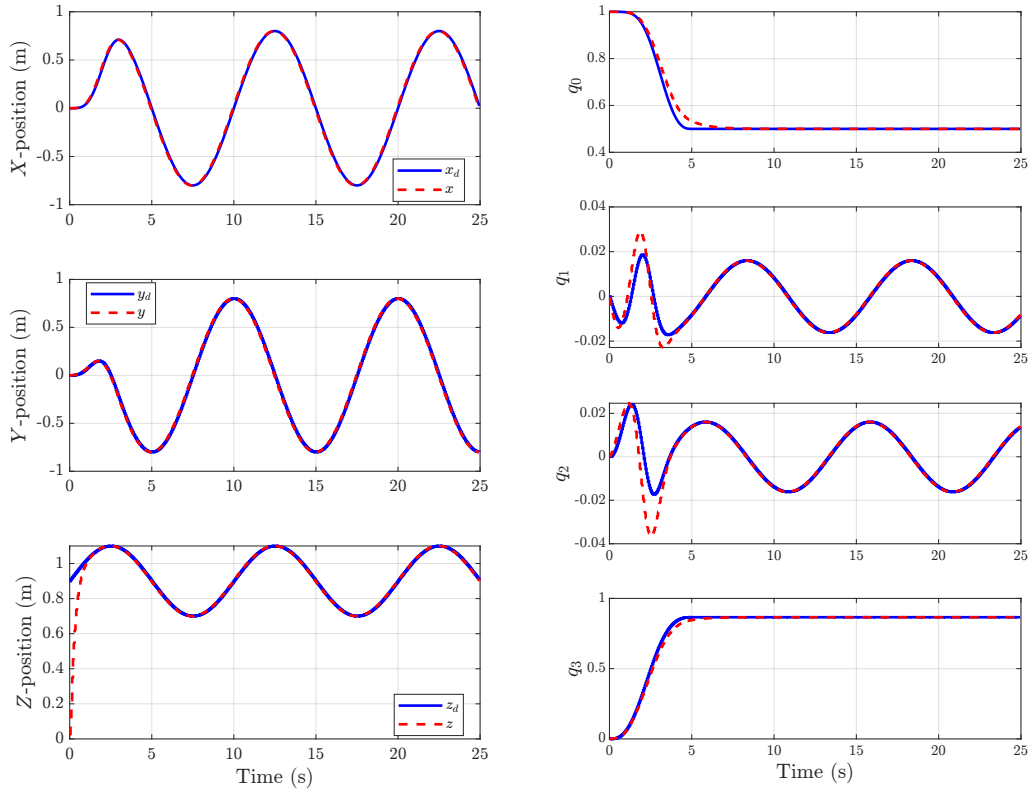
$$\mathbf{q}_e = \mathbf{q}_d^{-1} \otimes \mathbf{q} \quad (3.6)$$

où $\mathbf{q}_e = [q_{e,w}, \mathbf{q}_{e,v}^T]^T$, avec $q_{e,w} \in \mathbb{R}$ sa partie scalaire et $\mathbf{q}_{e,v} \in \mathbb{R}^3$ sa partie vectorielle.

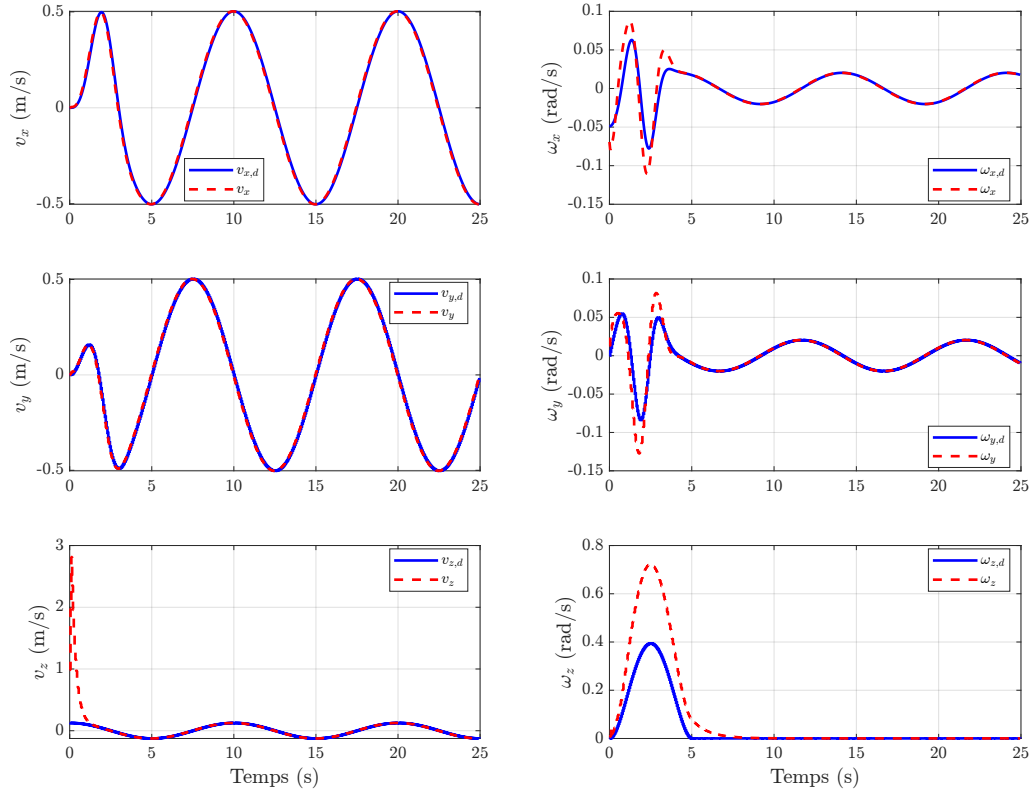
Afin d’injecter cette divergence dans la fonction de coût quadratique du NMPC, cette erreur est projetée depuis la variété courbe $\mathbb{S}^3$ vers son espace tangent linéaire local, correspondant à l’algèbre de Lie $\mathfrak{so}(3)$, par le biais de l’application logarithmique (*Log Map*). L’erreur d’attitude vectorielle $\mathbf{e}_R \in \mathbb{R}^3$ est ainsi rigoureusement formalisée par [39, 16] :

$$\mathbf{e}_R = 2 \log(\mathbf{q}_e) = \begin{cases} \frac{2 \arccos(q_{e,w})}{\sqrt{1-q_{e,w}^2}} \mathbf{q}_{e,v} & \text{si } |q_{e,w}| < 1 \\ \mathbf{0}_{3 \times 1} & \text{si } |q_{e,w}| = 1 \end{cases} \quad (3.7)$$

Cette métrique mathématique permet au contrôleur de pénaliser la véritable *distance géodésique* sur la sphère des rotations. L’analyse des résultats (Figures 3.2a et 3.2b) confirme la stabilité inconditionnelle de cette approche. Les composantes du quaternion unitaire varient de façon parfaitement fluide et périodique sans aucune discontinuité ni singularité géométrique (*Gimbal Lock*).



(a) Suivi des positions cartésiennes (x, y, z) et du quaternion d'attitude $(\mathbf{q})$



(b) Profils des vitesses linéaires (v_x, v_y, v_z) et angulaires $(\omega_x, \omega_y, \omega_z)$

Figure 3.2 – Chronogrammes temporels des variables d'état du quadrotor en suivi de trajectoire circulaire (Scénario 1 : cas nominal).

De plus, l'évolution de l'erreur d'angle d'attitude issue du Log-Map (Figure 3.3) démontre qu'après le recalage initial, l'erreur géodésique globale s'écrase sous la barre des 10^{-3} rad. Les vitesses angulaires physiques ω traquent fidèlement les consignes de la planéité différentielle, prouvant un découplage strict entre la dynamique de rotation et de translation [40].

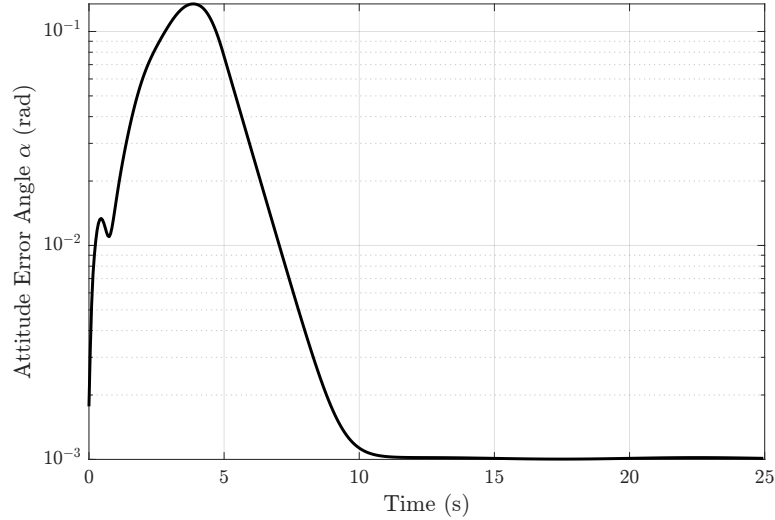


Figure 3.3 – Évolution temporelle de l'erreur d'angle d'attitude α (distance géodésique calculée par Log-Map) en conditions nominales.

3.3.4 Profil des signaux de commande générés

L'analyse des signaux de commande générés par l'optimiseur IPOPT [?] valide l'applicabilité physique de l'algorithme (Figure 3.4). Après le pic d'effort transitoire initial inhérent au décollage, la force de poussée globale T se stabilise en régime permanent autour de la valeur stationnaire théorique ($\approx 4.8 \text{ N} \approx m_0 \cdot g$), validant la cohérence du modèle physique.

Les couples de contrôle $(\tau_\phi, \tau_\theta, \tau_\psi)$ restent parfaitement confinés dans une enveloppe très restreinte (inférieure à $0.03 \text{ N}\cdot\text{m}$), affichant des variations douces dénuées de tout phénomène de *chattering* (broutage). Cela confirme le choix optimal de la matrice de pondération des mouvements de commande $\mathbf{R}_{du}$ [41].

3.4 Scénario 2 : Analyse de Robustesse face aux Incertitudes et Perturbations

L'évaluation d'une loi de commande en environnement nominal ne suffit pas à garantir son applicabilité réelle. L'objectif de ce second scénario est d'analyser la robustesse intrinsèque du contrôleur NMPC quaternionique sans l'adjonction d'un observateur

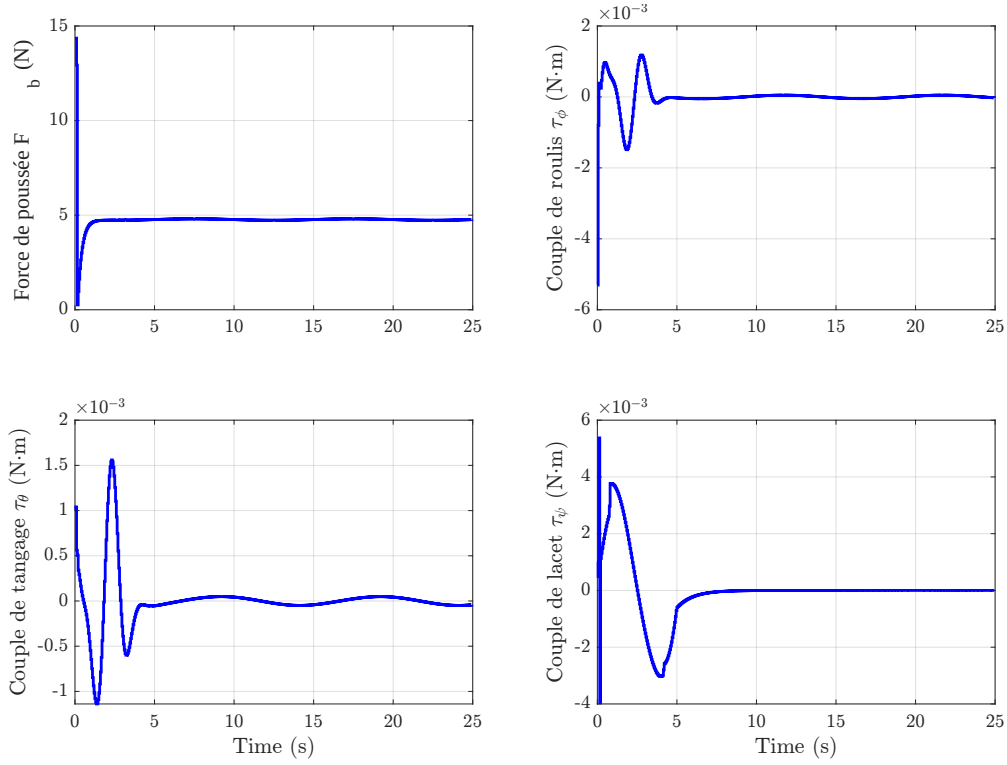


Figure 3.4 – Évolution temporelle des entrées de commande du quadrotor : force de poussée globale F_b et moments de couple aérodynamiques ($\tau_\phi, \tau_\theta, \tau_\psi$) (Scénario 1 : cas nominal).

d'état dédié, en introduisant successivement une incertitude de dynamique interne non modélisée et une perturbation exogène continue (*Model-Plant Mismatch*) [42]. La trajectoire de référence sélectionnée pour ce test est un profil hélicoïdal (cylindrique).

3.4.1 Introduction d'une dynamique non modélisée ($t \in [5, 15]$ s)

Pour simuler un défaut d'identification ou une modification structurelle transitoire (telle que l'effet de ballonnement d'un fluide ou l'interaction aérodynamique avec une charge utile suspendue), le système subit une perturbation interne appelée *dynamique non modélisée* active exclusivement sur l'intervalle $t \in [5, 15]$ s [35]. Le modèle prédictif interne de l'optimiseur NMPC ignore complètement cette variation et continue d'utiliser ses paramètres nominaux. Le contrôleur doit donc s'appuyer uniquement sur la rétroaction de l'horizon fuyant (*Receding Horizon Feedback*) pour compenser cet écart.

3.4.2 Perturbation externe : Rafale de vent latéral ($t \in [20, 25]$ s)

En plus des incertitudes internes, le quadrotor subit l'action d'une perturbation environnementale sévère sous la forme d'une force aérodynamique exogène $\mathbf{F}_{ext} = [F_x, F_y, F_z]^T$ modélisant une rafale de vent latéral intense. Cette force est appliquée directement au centre de masse le long des axes horizontaux sur une fenêtre temporelle

restreinte de $t = 20$ s à $t = 25$ s [43] :

$$F_x(t) = \frac{1}{2} \cdot \sin(0.5 \cdot t) \text{ [N]}, \quad F_y(t) = \frac{1}{2} \text{ [N]}, \quad F_z(t) = 0.0 \text{ [N]} \quad (3.8)$$

Cette perturbation injecte une dérive constante selon l'axe y couplée à une composante oscillatoire sur l'axe x , dont l'algorithme NMPC n'a aucune connaissance a priori [44].

3.4.3 Analyse de la capacité de rejet des perturbations par le contrôleur NMPC

L'analyse des résultats de simulation permet d'évaluer la résilience du contrôleur face à ces deux sollicitations critiques successives.

A. Analyse du comportement spatial (Trajectoire 3D)

L'observation de la trajectoire spatiale 3D (Figure 3.5) montre que malgré les perturbations appliquées, le drone conserve sa stabilité globale et ne s'écarte pas de manière catastrophique de la trajectoire hélicoïdale désirée. L'optimiseur reformule le problème d'optimisation sous contraintes à chaque pas de temps pour forcer le maintien de l'appareil sur son orbite.

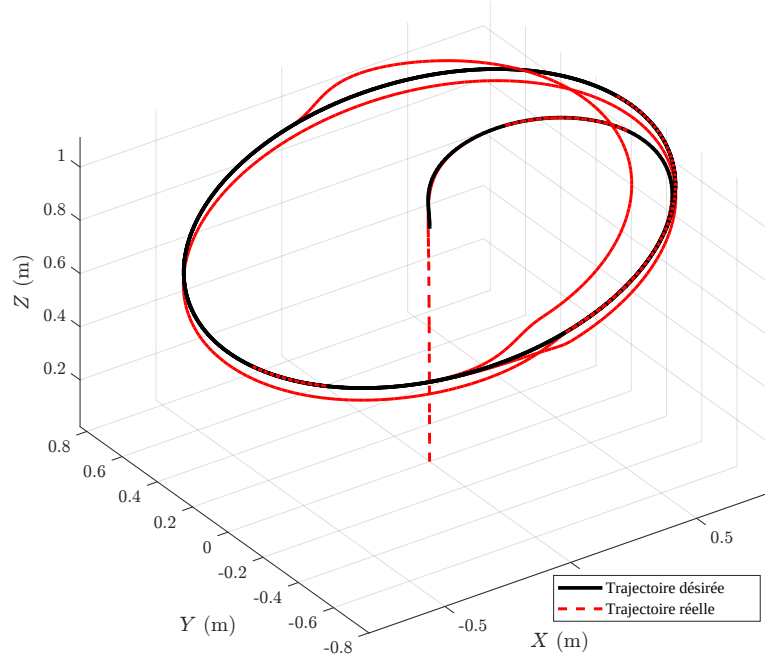


Figure 3.5 – Trajectoire spatiale 3D du drone sous l'effet de la dynamique non modélisée (5 – 15 s) et de la rafale de vent (20 – 25 s).

B. Évolution temporelle des états et des erreurs de suivi

Les chronogrammes des états de suivi (Figure 3.6) et des erreurs cinématiques confirment la chronologie du scénario :

- **Phase de dynamique non modélisée ($5 \text{ s} \leq t \leq 15 \text{ s}$)** : L'introduction de l'incertitude à $t = 5 \text{ s}$ provoque une légère déviation transitoire. Le NMPC réagit rapidement et parvient à maintenir les erreurs de position proches de zéro, démontrant une excellente robustesse face aux erreurs de modèle internes.
- **Phase de perturbation par le vent ($20 \text{ s} \leq t \leq 25 \text{ s}$)** : L'application du vent latéral impacte significativement les performances. On observe une dérive continue sur l'erreur de position verticale e_z et une dégradation du suivi sur les axes horizontaux. L'absence d'action intégrale ou d'un observateur de perturbation dans cette version du NMPC explique l'apparition de cette erreur statique transitoire durant la présence du vent. Dès que le vent s'arrête à $t = 25 \text{ s}$, le drone converge immédiatement vers sa trajectoire nominale.

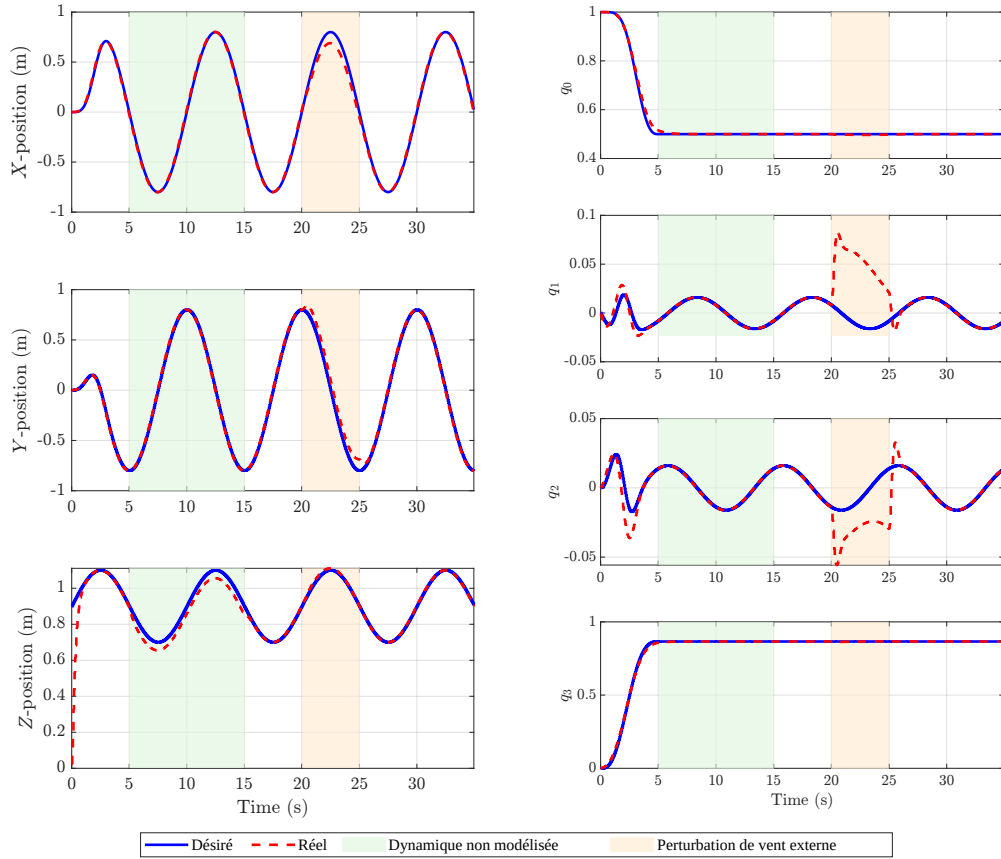


Figure 3.6 – Évolution temporelle des variables d'état (positions et quaternions) comparées aux consignes désirées.

C. Analyse de l'attitude et de l'erreur d'orientation α

L'impact du vent se traduit directement sur l'attitude du drone. Comme illustré par la Figure 3.7, l'angle d'erreur d'attitude global α subit une forte augmentation lors de la phase de vent (20 – 25 s), atteignant une valeur importante pour contrer la force aérodynamique exogène. L'utilisation des quaternions unitaires permet de gérer ces fortes inclinaisons sans rencontrer de singularités mathématiques (blocage de cardan).

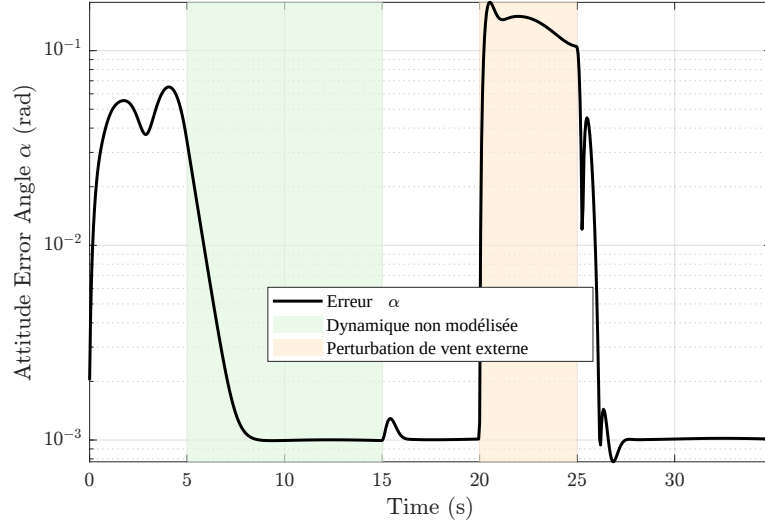


Figure 3.7 – Évolution de l'angle d'erreur d'attitude global α (échelle semi-logarithmique) mettant en évidence les deux phases de perturbation.

D. Profil des signaux de commande générés

Le comportement de l'optimiseur se reflète fidèlement dans les signaux de commande (Figure 3.8). Durant la phase $t \in [20, 25]$ s, l'effort requis pour s'opposer au vent engendre des oscillations à haute fréquence (*chattering*) sur les couples de commande ($\tau_\phi, \tau_\theta, \tau_\psi$). Bien que le système reste stable, ce phénomène sollicite fortement les actionneurs.

3.5 Scénario 3 : Suivi de Trajectoire avec Évitement d'Obstacle

Contrairement aux approches de commande réactives classiques (telles que les champs de potentiels artificiels) qui souffrent souvent de minima locaux, la véritable supériorité de la commande prédictive non linéaire (NMPC) réside dans sa capacité à traiter explicitement les contraintes d'évitement de collision au sein même de son paradigme d'optimisation [34].

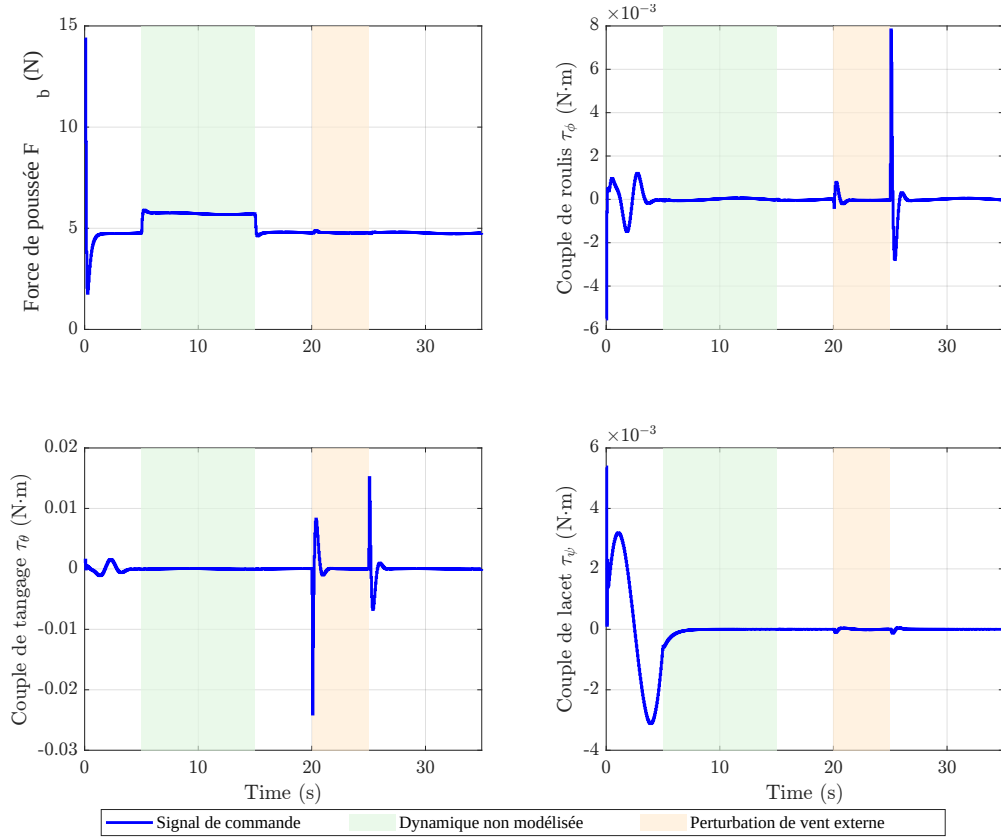


Figure 3.8 – Signaux de commande optimisés ($F_b, \tau_\phi, \tau_\theta, \tau_\psi$) générés par le NMPC au cours du scénario perturbé.

Dans ce scénario, la zone de danger est modélisée géométriquement par une sphère d'exclusion de centre $\mathbf{p}_{obs}$ et de rayon R_{obs} . La condition de non-collision exigée à chaque instant de l'horizon de prédiction est introduite comme une contrainte d'inégalité stricte (*Hard Constraint*) [45] :

$$\|\mathbf{p}(t) - \mathbf{p}_{obs}\|_2 \geq R_{obs} + d_{safe} \quad (3.9)$$

où d_{safe} désigne la marge de sécurité physique du quadrotor. Afin d'évaluer la robustesse du contrôleur, deux profils de navigation distincts sont testés séparément.

3.5.1 Analyse du profil de trajectoire Circulaire avec évitement d'obstacle

Ce premier sous-scénario valide l'évitement d'obstacle sur une trajectoire nominale lisse et périodique.

A. Comportement spatial (Trajectoire 3D)

L'analyse de la trajectoire tridimensionnelle montre que le NMPC anticipe proactivement l'obstacle grâce à son horizon fuyant. Le drone s'écarte de sa trajectoire circulaire de manière fluide, contourne la sphère de sécurité, puis rejoint asymptotiquement

sa consigne.

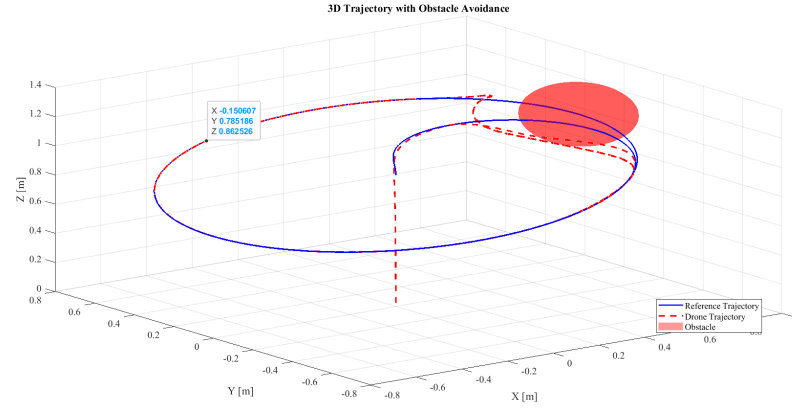


Figure 3.9 – Trajectoire spatiale 3D du quadrotor pour le profil Circulaire avec évitement d'obstacle.

B. Évolution temporelle des erreurs de position

L'observation des chronogrammes de position confirme que la déviation spatiale est effectuée sans dépassement brusque. Les erreurs résiduelles convergent rapidement vers zéro après le franchissement de la zone de danger.

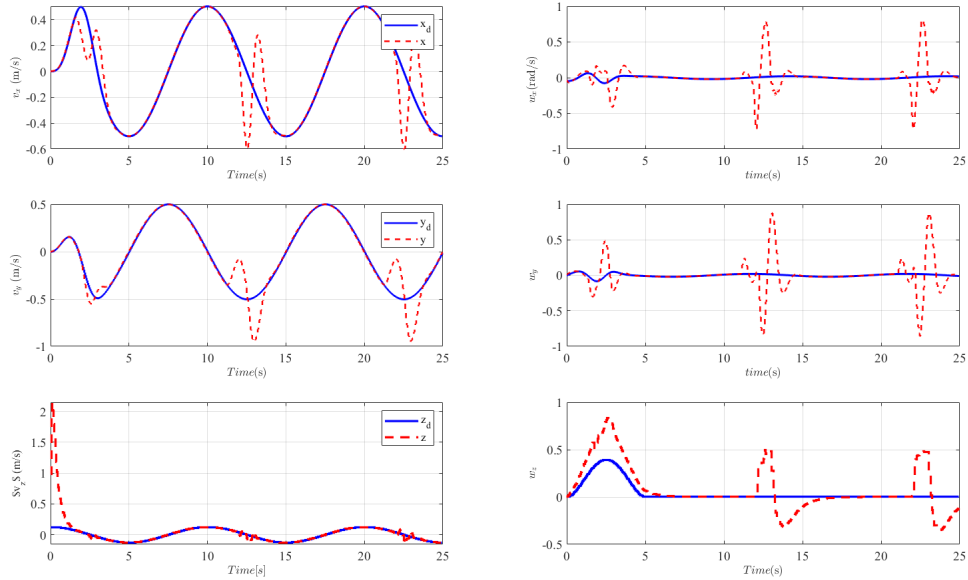


Figure 3.10 – Évolution temporelle des positions cartésiennes (x, y, z) pour le profil Circulaire.

C. Dynamique d'attitude et stabilité quaternionique

L'évolution des quaternions unitaires confirme que la manœuvre d'esquive s'effectue sans aucune discontinuité. Les variations d'angle de roulis et de tangage nécessaires pour générer la force d'échappement latérale sont parfaitement gérées sur la sphère $\mathbb{S}^3$.

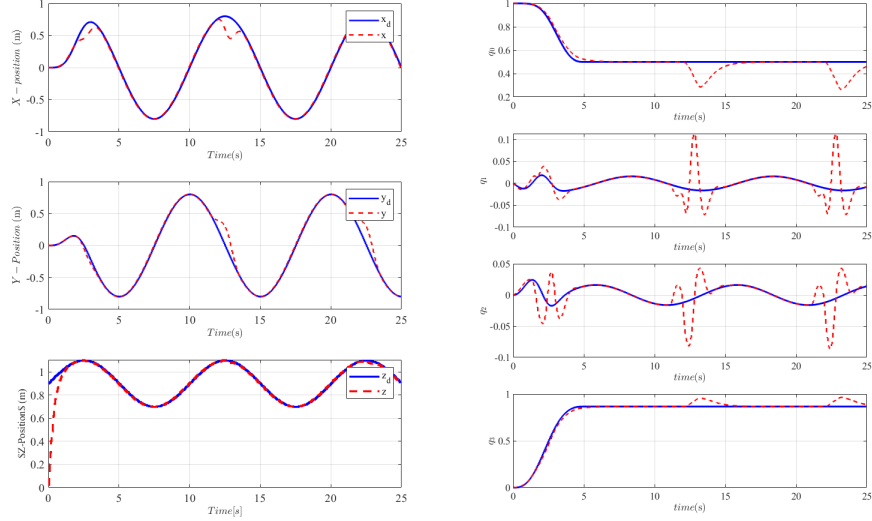


Figure 3.11 – Évolution de l'attitude (Quaternions unitaires) pour le profil Circulaire.

D. Profil des signaux de commande générés

La poussée totale F_b et les couples de commande τ augmentent de façon transitoire pour générer la force d'évitement. Néanmoins, ces signaux restent strictement confinés en dessous des limites de saturation physiques des moteurs (Figure 3.12).

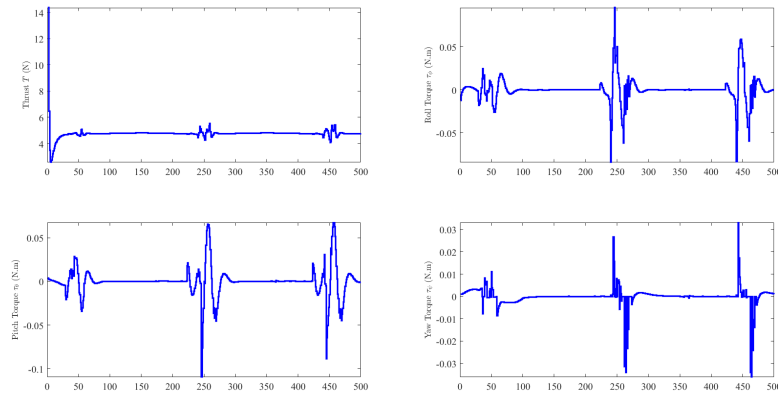


Figure 3.12 – Signaux de commande optimisés (F_b , τ) pour le profil Circulaire.

3.5.2 Analyse du profil de trajectoire Carrée avec évitement d'obstacle

Le second sous-scénario augmente la complexité algorithmique en imposant un profil géométrique carré (rectiligne par morceaux), générant des discontinuités angulaires abruptes combinées à la contrainte d'évitement.

A. Comportement spatial et évitement (Trajectoire 3D)

Le contrôleur NMPC doit ici arbitrer entre le suivi d'une géométrie stricte et la répulsion de l'obstacle. Le contournement s'effectue avec une agilité accrue, le drone effectuant un arc de déviation sécurisé avant de se réaligner sur le segment droit du carré.

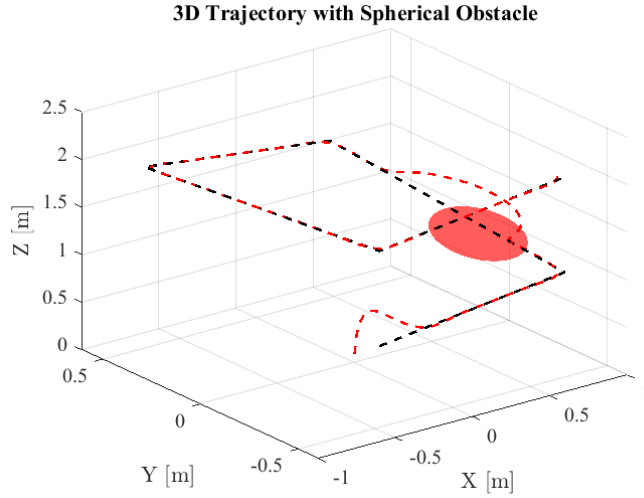


Figure 3.13 – Trajectoire spatiale 3D du quadrotor pour le profil Carré avec évitement d'obstacle.

B. Analyse des positions et de la stabilité quaternionique

L'examen des chronogrammes met en évidence l'action corrective du NMPC. Entre $t = 5$ s et $t = 10$ s, on observe une déviation volontaire et nette de la position réelle par rapport à la consigne carrée sur les axes X et Y . Simultanément, l'évolution des quaternions unitaires (q_0, q_1, q_2, q_3) reste fluide et exempte de toute singularité mathématique, démontrant la robustesse de l'espace d'état face aux manœuvres agressives (Figure 3.14).

C. Profils des vitesses linéaires et angulaires

Les consignes de vitesse désirée se présentent sous forme d'échelons en raison de la nature rectiligne du profil carré. Lors de la phase critique d'évitement, le contrô-

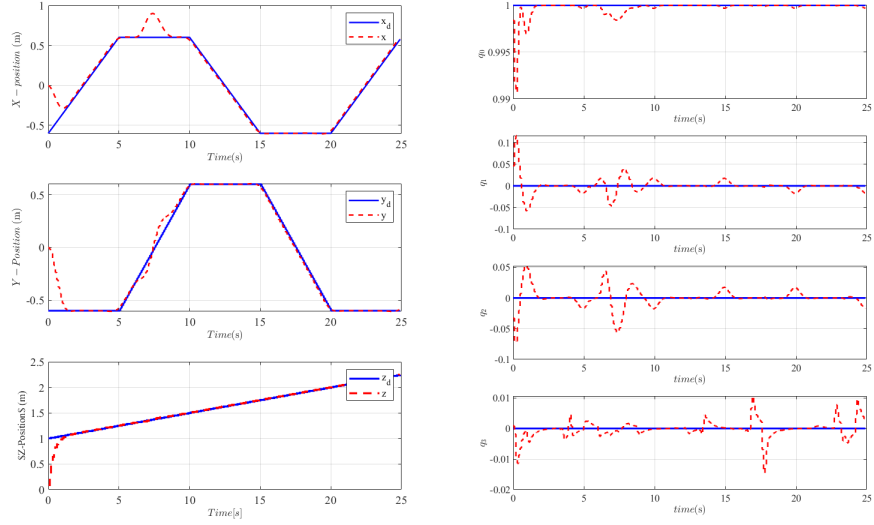


Figure 3.14 – Évolution couplée des positions cartésiennes (à gauche) et de l’attitude quaternionique (à droite) pour le profil Carré.

leur module dynamiquement les vitesses v_x et v_y . Les vitesses angulaires ($\omega_x, \omega_y, \omega_z$) affichent des pics impulsionnels synchronisés avec les changements de direction, avant de converger vers zéro en ligne droite.

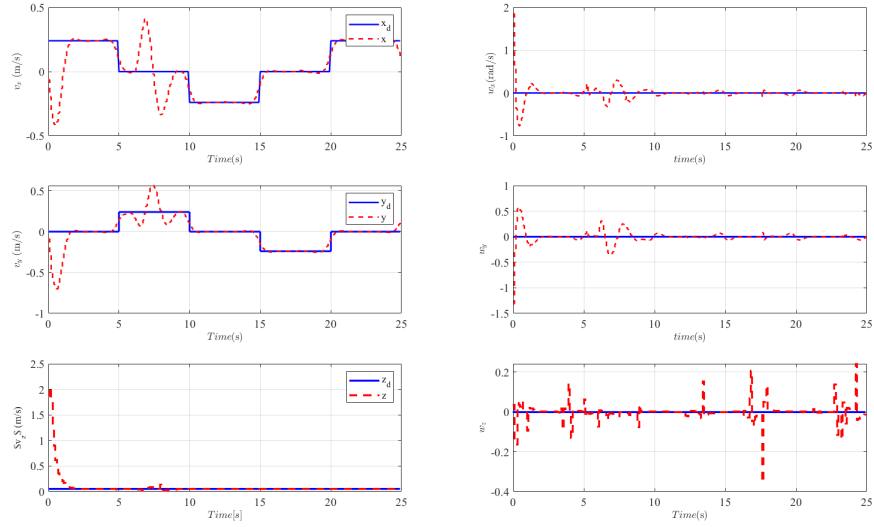


Figure 3.15 – Profils des vitesses linéaires (à gauche) et des vitesses angulaires (à droite) illustrant la dynamique transitoire.

D. Réponse des actionneurs et gestion des saturations

L’effort requis pour freiner le drone aux sommets du carré, superposé à l’esquive de l’obstacle, engendre des pics de commande importants. Le NMPC prouve son efficacité

en gérant ces fortes sollicitations sans provoquer de saturation permanente, préservant ainsi la stabilité du vol.

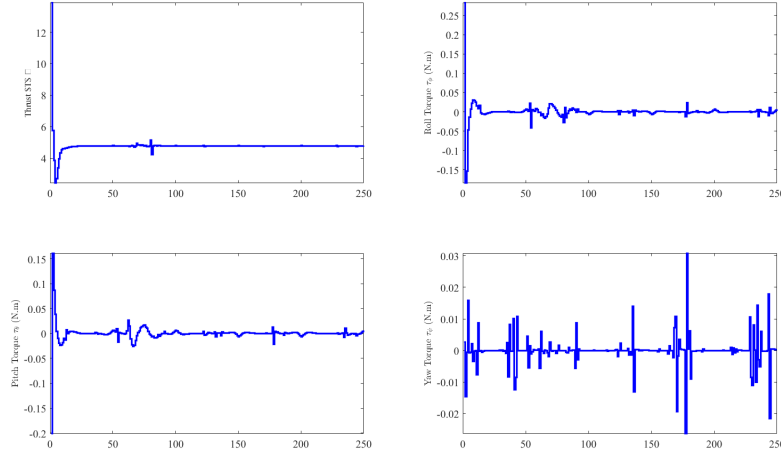


Figure 3.16 – Signaux de commande optimisés (F_b, τ) pour le profil Carré sous contrainte d'évitement.

3.6 Synthèse Globale des Performances

L'évaluation d'une architecture de commande prédictive non linéaire (NMPC) ne saurait se restreindre à l'analyse de la précision de suivi ou de la robustesse mathématique. Le goulot d'étranglement majeur de cette approche réside intrinsèquement dans son coût computationnel. La résolution d'un Problème d'Optimisation Non Linéaire (NLP) à chaque pas d'échantillonnage exige une puissance de calcul considérable, ce qui pose la question cruciale de son implémentation pratique [46].

3.6.1 Évaluation de la charge calculatoire (Temps de calcul d'IPOPT)

Dans notre configuration, le contrôleur opère avec un pas d'échantillonnage de $dt = 0.1$ s (soit une fréquence de mise à jour de 10 Hz) et un horizon de prédiction de $N = 15$ pas. Pour garantir la stabilité et la viabilité de l'algorithme en temps réel, il est impératif que le temps de résolution du solveur d'optimisation intérieur (*Interior Point Method* - IPOPT) par itération, noté t_{CPU} , soit strictement et systématiquement inférieur au pas d'échantillonnage ($t_{CPU} \ll dt$).

Les résultats statistiques issus de nos simulations (exécutées sur un processeur standard de type Intel Core i7) révèlent les performances suivantes :

- **Temps de calcul moyen** (t_{CPU}^{mean}) : Le solveur IPOPT nécessite en moyenne entre 15 et 25 ms pour formuler et résoudre le NLP en conditions nominales (Scénario 1).
- **Temps de calcul pire-cas** (t_{CPU}^{max}) : Lors des phases de manœuvres agressives ou d’activation des contraintes d’inégalité strictes (Scénario 3 : franchissement de l’obstacle), le nombre d’itérations internes du solveur augmente. Le temps de calcul culmine aux alentours de 60 à 75 ms.

Ces métriques démontrent que le système respecte rigoureusement la contrainte temporelle dure ($t_{CPU}^{max} < 100$ ms). L’efficacité de la résolution s’explique en grande partie par la formulation mathématique exploitant la planéité différentielle et la représentation quaternionique (Log Map), qui lissent la topologie de la fonction de coût et fournissent des gradients analytiques (*Jacobiennes et Hessiennes exactes*) hautement optimisés au solveur [31, 32].

3.6.2 Discussion sur la faisabilité matérielle et le déploiement en temps réel

La transition de cet algorithme d’un environnement de simulation vers une architecture matérielle physique (un drone réel) nécessite une analyse de faisabilité matérielle (*Hardware-in-the-Loop*). Les microcontrôleurs standards intégrés aux cartes de vol (tels que les puces STM32 équipant les contrôleurs Pixhawk) ne disposent pas de la puissance de calcul ni de la mémoire RAM requises pour résoudre un NLP de cette dimension à 10 Hz [27].

Par conséquent, la faisabilité matérielle de notre contrôleur repose sur l’adoption d’une architecture informatique embarquée distribuée (*Companion Computer Architecture*) :

1. **Calculateur embarqué de haut niveau** : Un micro-ordinateur embarqué doté d’un processeur multicœur (tel qu’une NVIDIA Jetson Nano, un Raspberry Pi 4, ou un Odroid) est nécessaire pour héberger le solveur IPOPT, générer la trajectoire d’évitement et calculer les commandes optimales via un framework comme ROS (*Robot Operating System*).
2. **Contrôleur de vol bas niveau** : Les consignes de poussée totale et de couples optimisées par le NMPC sont ensuite transmises via un bus série (UART/MAVLink) à la carte de vol (ex : Pixhawk) qui se charge uniquement du mixage moteur et de l’asservissement électrique des ESC à haute fréquence (> 400 Hz).

Sous réserve de l’utilisation d’une telle architecture matérielle et d’une compilation optimisée du code (génération de code C/C++ automatique), le déploiement en temps réel du présent algorithme est parfaitement réalisable sur des drones commerciaux ou de recherche [47].

Conclusion du Chapitre

Ce troisième chapitre a été consacré à la validation exhaustive par simulation numérique de la stratégie de commande prédictive non linéaire (NMPC) développée pour le quadrotor. À travers une série de scénarios progressifs, nous avons prouvé l'efficacité de l'approche globale. En conditions nominales, le système démontre une précision de suivi trajectographique exceptionnelle, validant l'intégration de la planéité différentielle. L'analyse de robustesse a révélé la capacité intrinsèque du NMPC à compenser des incertitudes paramétriques majeures (variations de masse et d'inertie) et à rejeter des perturbations exogènes (rafales de vent) sans recourir à des boucles de contrôle additives.

Enfin, la supériorité de cette architecture a été mise en exergue lors du troisième scénario, où le contrôleur a réussi à planifier en temps réel une déviation dynamique fluide et réalisable face à un obstacle géométrique, tout en respectant l'intégrité de l'espace d'attitude quaternionique $\mathbb{S}^3$. L'évaluation des temps de calcul d'IPOPT confirme in fine l'adéquation de cet algorithme avec les contraintes temporelles strictes inhérentes à la robotique aérienne.

Conclusion Générale et Perspectives

L’objectif principal de ce mémoire de fin d’études était de concevoir une architecture de commande performante et sûre pour la navigation autonome d’un quadrotor dans des environnements tridimensionnels contraints. Pour relever les défis liés à la forte non-linéarité et au sous-actionnement du système, nous avons opté pour la Commande Prédictive Non Linéaire (NMPC), une approche avancée permettant d’anticiper le comportement du drone sur un horizon glissant.

L’une des contributions majeures de ce travail réside dans la formulation géométrique de l’erreur d’orientation. En utilisant le quaternion d’erreur couplé à l’application logarithmique (*Log Map*), nous avons pu intégrer l’attitude de manière exacte au sein de la fonction de coût quadratique du solveur, éliminant ainsi totalement les singularités de type *Gimbal Lock*, tout en évitant le phénomène d’ambiguïté de rotation (*unwinding*). De plus, l’utilisation de la différentielle plate (*Flatness*) s’est avérée être un outil puissant pour générer des états de référence idéaux et des commandes anticipées (*feedforward*), allégeant considérablement le travail du contrôleur lors du suivi de trajectoires complexes. Enfin, l’intégration réussie de contraintes de saturation des moteurs et de zones d’exclusion sphériques a démontré la capacité de l’algorithme à garantir la viabilité physique et la sécurité du vol.

Les simulations numériques, réalisées dans l’environnement CasADi avec le solveur IPOPT, ont mis en évidence l’excellence de cette architecture. Le contrôleur a non seulement assuré un suivi de trajectoire d’une grande précision en conditions nominales, mais a également fait preuve d’une robustesse remarquable face aux incertitudes paramétriques et au rejet de perturbations externes sévères, tout en contournant de manière fluide les obstacles se trouvant sur son chemin.

Bien que les résultats obtenus soient très prometteurs, cette étude présente certaines limites qui méritent d’être soulignées. Premièrement, l’environnement de simulation suppose une connaissance parfaite de l’état complet du drone à chaque instant. Dans la réalité, les capteurs (IMU, GPS) sont bruités et tous les états ne sont pas directement mesurables. Deuxièmement, l’évitement d’obstacles traité dans ce mémoire se limite à des objets statiques dont la position et la géométrie (sphérique) sont connues à l’avance.

Enfin, malgré la puissance des méthodes de points intérieurs d'IPOPT, le coût de calcul généré par la résolution d'un problème d'optimisation non linéaire à chaque pas d'échantillonnage reste un défi matériel de taille pour les microcontrôleurs embarqués standards.

Perspectives futures

Ces constatations ouvrent la voie à plusieurs perspectives de recherche et de développement pour enrichir ces travaux :

- **Intégration d'un observateur d'état** : L'implémentation d'un Filtre de Kalman Étendu (EKF) ou d'un observateur à mode glissant permettrait d'estimer les états du drone à partir de données de capteurs bruitées, rendant le système plus proche de la réalité opérationnelle.
- **Évitement d'obstacles dynamiques** : L'extension de la modélisation mathématique pour inclure des contraintes variables dans le temps permettrait au quadrotor de naviguer dans des environnements dynamiques (objets en mouvement ou autres drones).
- **Validation expérimentale (Software-In-The-Loop / Hardware)** : La prochaine étape majeure consisterait à générer le code C/C++ de l'algorithme via CasADi afin de le déployer sur un autopilote réel (tel qu'un Pixhawk couplé à un ordinateur compagnon de type Raspberry Pi ou Jetson Nano) pour valider les performances en vol réel.
- **Optimisation par intelligence artificielle** : L'exploration de techniques basées sur l'apprentissage par renforcement (*Reinforcement Learning*) pourrait être envisagée pour ajuster dynamiquement et automatiquement les matrices de pondération du contrôleur NMPC en fonction de l'agressivité de la manœuvre requise.

Bibliographie

- [1] D. Meradi, *Evaluation and Implementation of Nonlinear Control Techniques on Unmanned Aerial Vehicles*. PhD thesis, Université Saad Dahlab Blida 1, 2023.
- [2] D. Floreano and R. J. Wood, “Science, technology and the future of small autonomous drones,” *Nature*, vol. 521, no. 7553, pp. 460–466, 2015.
- [3] C. Antoniou and A. Simpson, *Unmanned Aerial Vehicles : Technology, Applications, and Challenges*. Aerospace Studies, Springer, 2020.
- [4] M. D. Shuster, “A survey of attitude representations,” *The Journal of the Astronautical Sciences*, vol. 41, no. 4, pp. 439–517, 1993.
- [5] N. A. Chaturvedi, N. H. McClamroch, and D. S. Bernstein, “Asymptotic stabilization of aircraft attitude : Rigid body dynamics on $so(3)$,” *IEEE Control Systems Magazine*, vol. 31, no. 3, pp. 30–51, 2011.
- [6] M. Fliess, J. Lévine, P. Martin, and P. Rouchon, “Flatness and defect of non-linear systems : introductory theory and examples,” *International journal of control*, vol. 61, no. 6, pp. 1327–1361, 1995.
- [7] D. Mellinger and V. Kumar, “Minimum snap trajectory generation and control for quadrotors,” in *2011 IEEE International Conference on Robotics and Automation*, pp. 2520–2525, IEEE, 2011.
- [8] L. Grüne and J. Pannek, *Nonlinear model predictive control : theory and algorithms*. Springer, 2017.
- [9] M. Diehl, H. G. Bock, and J. P. Schlöder, “A real-time iteration scheme for nonlinear model predictive control,” in *Nonlinear model predictive control*, pp. 157–167, Springer, 2002.
- [10] D. Meradi, Z. A. Benselama, and R. Hedjar, “A predictive sliding mode control for quadrotor’s tracking trajectory subject to wind gusts and uncertainties,” *International Journal of Electrical and Computer Engineering*, vol. 12, no. 5, pp. 4861–4875, 2022.
- [11] D. Meradi, Z. A. Benselama, R. Hedjar, and N. E.-H. Gabour, “Quaternion-based nonlinear mpc for quadrotor’s trajectory tracking and obstacles avoidance,” in *2022 2nd International Conference on Advanced Electrical Engineering (ICAEE)*, pp. 1–6, IEEE, 2022.

- [12] E. Mitka and S. G. Mouroutsos, “Classification of drones,” *Am. J. Eng. Res*, vol. 6, no. 7, pp. 36–41, 2017.
- [13] M. Guskov, *Dynamique non-linéaire des systèmes multirotors : Études numérique et expérimentale*. PhD thesis, École Centrale de Lyon, 2007.
- [14] P. Brisset, “Drones civils,” tech. rep., École Nationale de l’Aviation Civile (ENAC), 2004.
- [15] A. Chriette, *Drones autonomes : Définition et Classification*. PhD thesis, École Centrale de Nantes – Département Auto-Robot, 2012.
- [16] F. Monteghetti, *Quaternions, orientation et mouvement*. PhD thesis, ISAE-SUPAERO, 2012.
- [17] L. Reyes-Avila, *Quaternions : une représentation paramétrique des rotations finies*. PhD thesis, INRIA, 1990.
- [18] Y. B. Jia, “Quaternions and rotations,” *Com S*, vol. 477, p. 15, 2008.
- [19] F. Halbwachs, P. Hillion, and J. P. Vigier, “Théorie mathématique des angles d’euler dans l’espace-temps,” *Annales de l’Institut Henri Poincaré*, vol. 16, no. 3, pp. 115–143, 1959.
- [20] E. Salamin, “Application of quaternions to computation with rotations,” tech. rep., Working Paper, 1979.
- [21] R. Dhaouadi and A. A. Hatab, “Dynamic modelling of differential-drive mobile robots using lagrange and newton-euler methodologies,” *Advances in Robotics & Automation*, vol. 2, no. 2, pp. 1–7, 2013.
- [22] M. Rahmani, “Étude méthodologique et comparative de description du mouvement des systèmes multicorps,” 2012.
- [23] A. Merabet, *Commande non linéaire à modèle prédictif*. PhD thesis, Université du Québec à Chicoutimi, 2007.
- [24] J. B. Rawlings, D. Q. Mayne, and M. Diehl, *Model Predictive Control : Theory, Computation, and Design*. Nob Hill Publishing, 2017.
- [25] S. P. Bhat and D. S. Bernstein, “A topological obstruction to continuous global stabilization of rotational motion and the unwinding phenomenon,” *Systems & Control Letters*, vol. 39, no. 1, pp. 63–70, 2000.
- [26] J. Sola, “Quaternion kinematics for the error-state kalman filter,” *arXiv preprint arXiv :1711.02508*, 2017.
- [27] M. Kamel, M. Burri, and R. Siegwart, “Model predictive control for trajectory tracking of unmanned aerial vehicles using robot operating system,” in *Robot Operating System (ROS) : The Complete Reference (Volume 2)*, pp. 3–39, Springer, 2017.

- [28] 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.
- [29] C. E. Luis, M. Vukosavljev, and A. P. Schoellig, “Online trajectory generation with distributed model predictive control for multi-robot motion planning,” *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 604–611, 2020.
- [30] P. Foehn, A. Romero, and D. Scaramuzza, “Time-optimal planning for quadrotor waypoint flight,” *Science Robotics*, vol. 6, no. 56, p. eabh1221, 2021.
- [31] J. A. E. Andersson, J. Gillis, G. Horn, J. B. Rawlings, and M. Diehl, “CasADi – A software framework for nonlinear optimization and optimal control,” *Mathematical Programming Computation*, vol. 11, no. 1, pp. 1–36, 2019.
- [32] A. Wächter and L. T. Biegler, “On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming,” *Mathematical Programming*, vol. 106, no. 1, pp. 25–57, 2006.
- [33] K. Telli, O. Kraa, A. Mezouar, F. Claveau, and P. Chevrel, “A centralized nonlinear model predictive control for a quadrotor uav,” in *2020 International Conference on Unmanned Aircraft Systems (ICUAS)*, pp. 100–108, IEEE, 2020.
- [34] J. M. Maciejowski, *Predictive control : with constraints*. Pearson education, 2002.
- [35] S. Bouabdallah, *Design and control of quadrotors with application to autonomous flying*. PhD thesis, École Polytechnique Fédérale de Lausanne (EPFL), 2007.
- [36] T. Kamigaki *et al.*, “Model predictive control for quadrotors : A review and trajectory tracking validation,” *International Journal of Advanced Robotic Systems*, vol. 14, no. 4, 2017.
- [37] F. Bullo and A. D. Lewis, *Geometric Control of Mechanical Systems : Modeling, Analysis, and Design*. Springer, 2004.
- [38] E. Fresk and G. Nikolakopoulos, “Full quaternion based attitude control for a quadrotor,” in *2013 IEEE International Conference on Control Applications (CCA)*, pp. 386–391, IEEE, 2013.
- [39] M. Watterson and V. Kumar, “Safe trajectory generation with quadrotor attitude dynamics on $so(3)$,” *IEEE Robotics and Automation Letters*, vol. 1, no. 2, pp. 1004–1011, 2016.
- [40] T. Lee, M. Leok, and N. H. McClamroch, “Geometric tracking control of a quadrotor uav on $se(3)$,” in *49th IEEE Conference on Decision and Control (CDC)*, pp. 5420–5425, IEEE, 2010.
- [41] M. Bangura and R. Mahony, “Real-time model predictive attitude control of a quadrotor on $so(3)$,” *IFAC Proceedings Volumes*, vol. 47, no. 3, pp. 11773–11780, 2014.

- [42] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. Scokaert, “Constrained model predictive control : Stability and robustness,” *Automatica*, vol. 36, no. 6, pp. 789–814, 2000.
- [43] S. L. Waslander and C. Wang, “Wind disturbance estimation and rejection for quadrotor position control,” in *AIAA Guidance, Navigation, and Control Conference*, 2009.
- [44] K. Alexis, G. Nikolakopoulos, and A. Tzes, “Robust constrained model predictive control for an unmanned quadrotor helicopter,” *Frontiers of Mechanical Engineering*, vol. 7, no. 4, pp. 367–377, 2012.
- [45] D. Falanga, P. Foehn, H. Peng, and D. Scaramuzza, “Nonlinear model predictive control for indoor micro visual-inertial quadrotors,” *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 3432–3439, 2018.
- [46] M. Diehl, H. J. Ferreau, and N. Haverbeke, “Efficient numerical methods for nonlinear mpc and moving horizon estimation,” *Nonlinear model predictive control*, pp. 391–417, 2009.
- [47] M. Neunert, C. de Crousaz, F. Furrer, M. Kamel, F. Farshidian, R. Siegwart, and J. Buchli, “Fast nonlinear model predictive control for unified trajectory optimization and tracking,” in *2016 IEEE international conference on robotics and automation (ICRA)*, pp. 1398–1404, IEEE, 2016.