

**République Algérienne Démocratique et Populaire**  
**Ministère de l'Enseignement Supérieur et de la Recherche Scientifique**  
**Université Mouloud MAMMERI, Tizi-Ouzou**



**Faculté de Génie Electrique et d'Informatique**  
**Département d'Automatique**

**Mémoire de Fin d'Etudes**

En vue de l'obtention du diplôme

*Master académique en automatique*  
*Option commande des systèmes*

***Thème***

***Commande Optimale Robuste d'un  
Drone***

Proposé et présenté par :

**Farid BOUNINI**

Dirigé par : **Mr Mohand Achour TOUAT**

Soutenu le : 14/09/2012.

***Promotion 2012***



## ***Remercîments***

*Je commence par dire louange, à celui qui a fixé un terme à toute chose. Je remercie le tout puissant pour m'avoir gratifié de santé et de moyens, sans eux, il serait impossible de mener à terme mon travail.*

*J'adresse mes vifs remerciements à :*

*Le chef de département d'Automatique Mr Bensidhoum, pour son soutien durant les diverses impasses.*

*Mon promoteur Mr TOUATI, Enseignant à l'université de Mouloud Mammeri Tizi-Ouzou.*

*Tous les enseignants et tous ceux à qui je dois ma formation.*

*Tous ceux qui, d'une quelconque façon, ont contribué à  
L'élaboration de ce mémoire.*

*Cordialement.*

## ***Dédicaces***

*Je dédie ce travail à ceux qui m'ont tout donné sans rien me demander et à qui je dois énormément et qui je ne remerciais jamais assez, Pour les sacrifices et les encouragements qu'ils n'ont cessé de me conférer.*

*Ma mère*

*Ma grande mère*

*Mes oncles Brahim et Hamid*

*Et ma sœur Saliha*

*• Aux deux familles AINOUZ ET BOUNINI*

*• A tous mes amis chacun à son nom*

*Farid BOUNINI*



## Table des matières

---

<b>Introduction générale</b> .....	1
<b>CHAPITRE I</b> .....	4
I.1. Surfaces de commande .....	5
I.2. Drones comme objets de commande .....	6
- Description de la dynamique du vol .....	7
I. 3. Détermination des équations de mouvement .....	9
I.3.1. Mouvement de translation .....	9
I.3.2. Mouvement de rotation .....	11
I.3.3. Contribution de la force de la gravitation aux équations de mouvement .....	13
I.3.3.a. Angles d'Euler et axes de transformation .....	15
I.3.3.b. Matrice directrice (matrice de transformation) .....	16
I.3.4. Système non linéaire .....	16
I.3.4.a. Linéarisation des termes inertiels et gravitationnels .....	17
I.3.4.c. Linéarisation des forces et moments aérodynamique à l'aide des séries de Taylor ...	19
I.4. Perturbation atmosphériques .....	21
I.4.1. Fonction mathématique d'une rafale discrète (Discret gust function) .....	21
I.4.2 Représentation continue de la rafale du vent .....	22
I.4.2.a. Modèle longitudinal .....	23
I.4.2.b. Modèle latéral .....	24
I.4.3. Représentation des interactions atmosphériques .....	25
I.4.3.a. Modèle longitudinal .....	25
I.4.3.b. Modèle latéral .....	26
I.4.4. Modèle des actionneurs .....	28
<b>Conclusion</b> .....	32
<b>CHAPITRE II</b> .....	33
<b>CONCEPTION DE LA LOI DE COMMANDE MULTIVARIABLE BASANT SUR L'OPTIMISATION ROBUSTE</b> .....	33
II.1. Synthèse de correcteur optimal basé sur le théorème de séparation dans le cas multi variables. ....	34
II.1.1. Robustification paramétrique de correcteur LQG (Linear Quadratic Gaussian) .....	39
II.3. Algorithmes génétiques comme procédure d'optimisation robuste .....	44
II. 3.1. Opérateurs de base des algorithmes génétiques. ....	45
II.3.2. L'élitisme .....	50

## Table des matières

---

II.3.3. Critère de convergence .....	50
<b>Conclusion</b> .....	51
<b>CHAPITRE III</b> .....	51
<b>RESULTATS DE SIMULATION</b> .....	51
3.1. Valeurs des normes .....	52
3.1.1. Longitudinales .....	52
3.1.2 . Latérales .....	54
3.2. Résultats de simulation et interprétation .....	57
<b>Conclusion</b> .....	66
<b>Annexe</b> .....	68
A.1. Commande Linéaire Quadratique (LQR) .....	68
A.1.1. Formulation de problème LQR .....	68
A.1.1.1. Principe du minimum de Pontriaguine .....	69
A.2. Commande linière quadratique gaussienne .....	73
A.2.1. Formulation de problème .....	73
A.2.2 . Théorème de séparation .....	74
A.3 . Structure de la commande LQG .....	74
<b>REFERENCES</b> .....	76

**INTRODUCTION**  
**GENERALE**

### Introduction générale:

Les drones ou UAV (Unmanned Aerial Vehicles) sont des engins volants sans pilote capables de mener à bien une mission de façon autonome ou en mode téléguidé. Leur utilisation est d'abord militaire pour des missions de reconnaissances ou de surveillance. En effet, ils sont bien adaptés pour la réalisation de missions qui mettraient potentiellement un équipage en danger ou qui nécessitent une permanence sur zone. Des applications civiles font leur apparition comme la surveillance du trafic autoroutier, la prévention des feux de forêts, la récolte de données météorologiques ou bien encore l'inspection d'ouvrages d'art.

Le développement de plates-formes robotiques volantes connaît un essor croissant depuis quelques années en raison de la miniaturisation toujours plus poussée des capteurs et des actionneurs, mais surtout grâce à la possibilité d'embarquer des cartes de commande toujours plus performantes capables d'exécuter la masse de calcul nécessaire au contrôle des robots volants.

Quelle que soit la méthode utilisée pour obtenir un modèle mathématique d'un système physique, il existe toujours un compromis entre la simplicité du modèle et son aptitude à décrire l'ensemble des phénomènes qui le caractérise. Ces écarts entre le modèle et le système réel sont souvent modélisés par des grandeurs d'erreurs (erreur d'entrée, erreur de sortie, erreur sur l'état,..., erreur de modèle). On peut ajouter à cela la nécessité d'avoir un modèle linéaire du système pour pouvoir appliquer la plupart des méthodes de commande, la linéarisation se faisant généralement autour d'un point de fonctionnement du processus physique. Ce modèle n'est donc valide que dans le voisinage de ce point. Le second type de phénomènes perturbateurs vient du fait que le système est soumis à des perturbations extérieures qui généralement ne sont pas prise en compte lors de la modélisation. En aéronautique, elles peuvent provenir du vent ou des turbulences atmosphériques.

Nous pouvons donc conclure que dans le meilleur des cas, il est seulement possible d'obtenir un modèle décrivant au mieux le comportement du système, ce qui nécessite une stabilisation robuste. En effet, un correcteur synthétisé sur le modèle linéarisé devra non seulement stabiliser le modèle mais aussi le système physique, incluant les phénomènes négligés, les erreurs de modèles et les perturbations extérieures auxquelles il pourrait être soumis.

Dans certains cas, on désire en plus conserver certaines spécifications faites sur les performances (temps de réponse, rapidité, découplage. . .). On parle alors de robustesse des performances. La méthode qui est généralement adoptée est de représenter le système

physique par un ensemble de modèles, appelé classe, tel que le système réel soit un élément de cet ensemble. Un correcteur qui stabilise cette classe stabilisera donc le système physique. Dès la fin des années 70, plusieurs méthodes de description de ces classes ont été mises en œuvre. Parmi ces méthodes, on peut citer la représentation non structurée, dans laquelle on considère que la seule information disponible pour caractériser la classe est la norme de la perturbation. Cette représentation non structurée peut porter soit sur la fonction de transfert du modèle linéaire, soit sur les paramètres de ses équations d'état, elle est alors appelée représentation paramétrique.

Les propriétés de robustesse, très attrayantes, du retour d'état encouragèrent le développement de plusieurs méthodes de synthèse de correcteurs stabilisant, la plus connue est la méthode LQG/LTR (Linear Quadratic Gaussian/ Loop Transfert Recovery), dont l'objectif est d'éviter les pertes de robustesse engendrées par l'adjonction d'un observateur dans la structure de commande. Cette méthode a été étendue aux variations de paramètres.

La méthode la plus utilisée dans le domaine de commande du vol est le gain programmé (gain scheduling), où la loi de commande est conçue pour chaque point de fonctionnement. Une telle technique est coûteuse, car plusieurs évaluations doivent être faites afin d'assurer les mêmes performances aux points hors conception [1].

Ces dernières années, un formalisme plus important de commande robuste est développé dans le domaine de l'aviation incluant des avions de guerre, drones, missiles et des véhicules spatiaux. Un important projet dans le domaine de la commande robuste du vol est pris en charge par GARTEUR ACTION GROUP (AG08) [1], les méthodes exposées dans leur référence sont basées sur l'optimisation  $H_2$  et  $H_\infty$ . Dans ce cas le correcteur doit d'une part stabiliser le système et d'autre part minimiser la norme  $H_2$  et  $H_\infty$  d'une fonction de transfert particulière.

L'utilisation des drones dans différents domaines est en pleine extension ces dernières années, il sera judicieux que les pays nord africains prennent des initiatives pour réguler les flux migratoire. Par exemple l'Algérie vit une crise cruciale dans la surveillance des frontières, vue l'étendu de sa surface et les problèmes vécus par les pays limitrophes. Pour ce faire il faut mettre en œuvre des moyens colossaux, au niveau des postes frontaliers et les services concernés, mais surmonter ce défi n'est pas une chose aisée, une surveillance aérienne s'impose, mais elle engendre encore un problème financier car un vol de trois heures d'un hélicoptère coûte environ 4 000 Dollars, ce qui n'est pas envisageable durant toute l'année et encore qu'il faut couvrir toutes les frontières.

## Introduction générale:

---

Ces contraintes peuvent être résolues en choisissant des DRONES comme moyen de surveillance aérienne, et semble moins coûteux à long termes car il existe des drones dont l'autonomie de vol excède 12 mois tel que helios couvert par des panneaux solaires chargeant ses batteries en lithium durant la journée.



Figure I : Drone solaire Helios

# CHAPITRE I

**CHAPITRE I****LES DRONES COMME OBJET DE COMMANDE**

Quel que soit la forme de modèle qu'un drone peut prendre, ses performances peuvent être déterminées par son pouvoir à évoluer et à respecter un chemin donné en fonction de temps et cela depuis le point de départ jusqu'à l'atteinte de point d'arrivé. Durant l'étude de mouvement d'un objet, il est possible d'exprimer son comportement en fonction de son vecteur de vitesse, les vecteurs de position et d'accélérations peuvent être obtenus avec une simple intégration ou dérivation de la vitesse respectivement, le vecteur de vitesse ; noté  $\dot{x}$  ; est exprimé en fonction de la position  $x$  de l'objet, de la commande  $u$ , qui sont à leurs tour dépendants du temps  $t$ , et éventuellement une certaine perturbation  $\xi$ . Souvent le vecteur de vitesse est représenté sous la forme générale par l'équation différentielle :

$$\dot{x} = f(x, u, \xi, t) \quad (1.1)$$

Où  $f$ : Est une fonction exprimant la relation entre les différentes grandeurs physiques.

Une autre forme est souvent rencontrée pour la plupart des machines industrielles, et elle possède une structure affine qui est très intéressante vue sa simplicité pour son étude par rapport à (1.1) :

$$\dot{x} = f(x, t) + g(x, t)u + b(x, t)\xi \quad (1.2)$$

L'expression du mouvement est caractérisée par l'ordre du système et le nombre d'entrées/sorties, la difficulté se situe dans le nombre de degrés de liberté ; par exemple, c'est plus aisé de travailler avec des systèmes évoluant dans une seule dimension que dans deux ou plus ; et en prenant un drone qui diffère des autres systèmes ordinaires car il possède six degrés de liberté : les trois premiers associés au mouvement angulaire de centre de gravité (c.g), les autres associés au mouvement de translation, c'est pour cela le problème de commande devient difficile.

La structure et les conditions de vol des drones tendent à les rendre instables, pour les maintenir dans l'air il est recommandé de concevoir des lois de commande robustes [2-5], qui assurent le contrôle tout au long de la trajectoire de vol, tout en procurant :

- Stabilité.
- Éliminations des perturbations internes (bruits de fond), externes (rafales de vent, chutes de pression...) ...etc.).
- Rendre le drone apte à poursuivre un itinéraire prédéterminé.

L'intégration d'un contrôleur AFCS (Automatic Flight Control System) dans un drone est donc inévitable. Cependant, pour ce genre d'appareil il est recommandé de concevoir des lois de commande spécifiques, qui peuvent satisfaire les performances souhaitées sans pour autant perdre la robustesse, par exemple l'utilisation des retours d'état et/ou de sortie ont une tendance d'osciller [3, 5-6], et c'est ainsi que les correcteurs procurent peu de stabilité [8], même si l'augmentation de gain de retour offre des réponses rapides et donne une précision acceptable, toutefois ; vue sa pondération il n'est pas possible de la générer, et l'augmentation excessive pourrait devenir la cause de l'instabilité du système.

### **I.1. Surfaces de commande :**

Pour diriger un objet en mouvement il suffit de lui appliquer une force, un moment et/ou une accélération au niveau de son centre de gravité, le vecteur de l'accélération peut être obtenu en appliquant sur le système la deuxième loi de mouvement. Tout objet volant possède des surfaces de commande ou d'autres moyens de générer des forces et des moments nécessaires pour la commande sur ses trois axes.

Nous pouvons distinguer ; pour la plupart des drones ; quatre surfaces de commande qui prédominent le mouvement longitudinal, transversal et latéral [7-8]:

- **Ailerons** : engendrent des moments autour de l'axe de roulis (Roll en Anglais) : Le mouvement de roulis d'un drone est contrôlé par des inclinaisons des ailerons. Ceux-ci se déplacent en sens opposés l'un de l'autre de telle sorte que lorsque l'un est relevé, l'autre est baissé et vice versa. La portance diminue du côté de l'aileron relevé et augmente du côté de l'aileron baissé : Ainsi, lorsque le drone s'incline vers la droite, l'aileron gauche se relève et celui de la droite s'abaisse simultanément entraînant un virage à droite de l'appareil (et inversement pour un virage à gauche).
- **Empennages de profondeur (Elevator en Anglais)** : engendrent des moments autour de l'axe de tangage (Pitch en Anglais) pour contrôler l'altitude et les autres grandeurs qui dépendent de l'angle d'attaque : Lorsque le drone est tiré vers l'arrière, la gouverne de profondeur se lève, provoquant la rotation de l'avion autour de l'axe de tangage (axe transversal passant par les extrémités des ailes et le centre de gravité) : l'arrière de l'appareil s'abaisse et le nez se lève (mouvement en cabré). Inversement, lorsque le

manche est poussé vers l'avant, la gouverne de profondeur s'abaisse faisant pivoter le nez de l'appareil vers le bas (mouvement de piqué). La gouverne de profondeur permet donc de contrôler l'assiette de l'avion (position de l'appareil par rapport à l'horizon).

- **Gouverne de direction (Rudder)** : crée des moments autour de l'axe lacet (Yaw en Anglais) : Le mouvement de lacet d'un drone est contrôlé par la gouverne de direction. En voulant se diriger à gauche, la gouverne de direction pivote vers la gauche, ce qui entraîne une dépression de côté droit de la gouverne et la rotation de l'avion autour de l'axe de lacet vers la gauche (et inversement pour tourner à droite). Par ailleurs, lors d'une mise en virage, le drone peut être amené à actionner la gouverne de direction du côté où il tourne de manière à corriger un phénomène appelé lacet inverse, susceptible de provoquer une dissymétrie du vol.
- **Etrangleur de moteur (Throttle)** : utilisé pour contrôler le taux d'oxygène lors de la combustion, afin de générer la poussée et la portance nécessaires lors de vol [décollage, atterrissage].

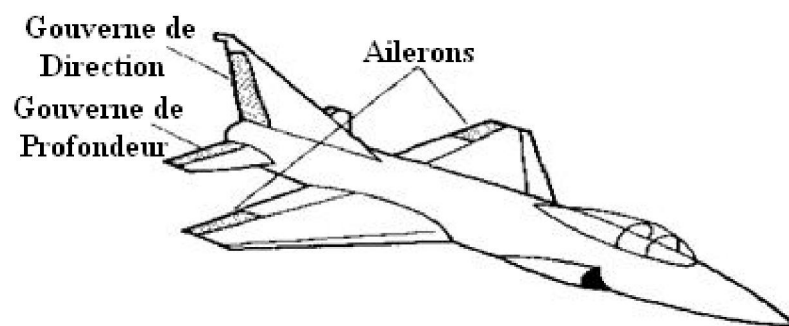


Figure 1.1 : Surfaces de commande

## I.2. Drones comme objets de commande :

Précédemment, nous avons cité les principales surfaces de commande pour gouverner le drone, et chaque appareil est doté des capteurs de mouvement qui détectent et qui délivrent les changements au niveau des grandeurs de mouvement, qui correspondent soit aux réponses de l'appareil dus aux commandes et/ou aux perturbations auxquelles le drone est soumis. Par ailleurs, vu la dimension et le poids de ce genre d'appareil il sera judicieux de réduire au minimum le nombre de capteurs utilisés à bord [9-11]. De ce fait il est primordial d'introduire un observateur dans la loi de commande. La Figure 1.2 met en évidence le sens et les signes de chaque surface de contrôle.

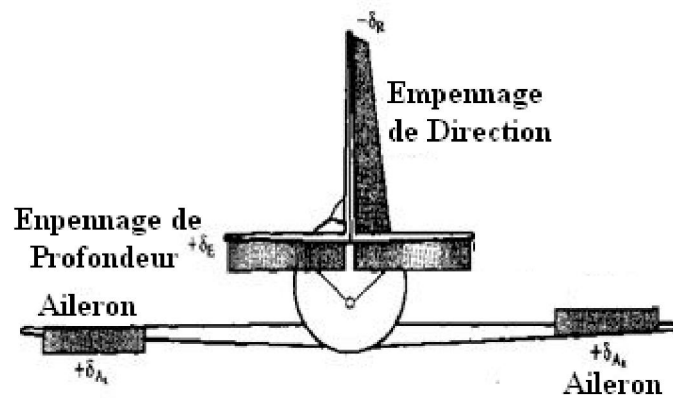


Figure 1.2 : Sens conventionnel des surfaces de commande

Le schéma bloc suivant montre la globalité des interférences existantes entre les différentes parties du système à bord, correcteur, capteurs et l'environnement [4]:

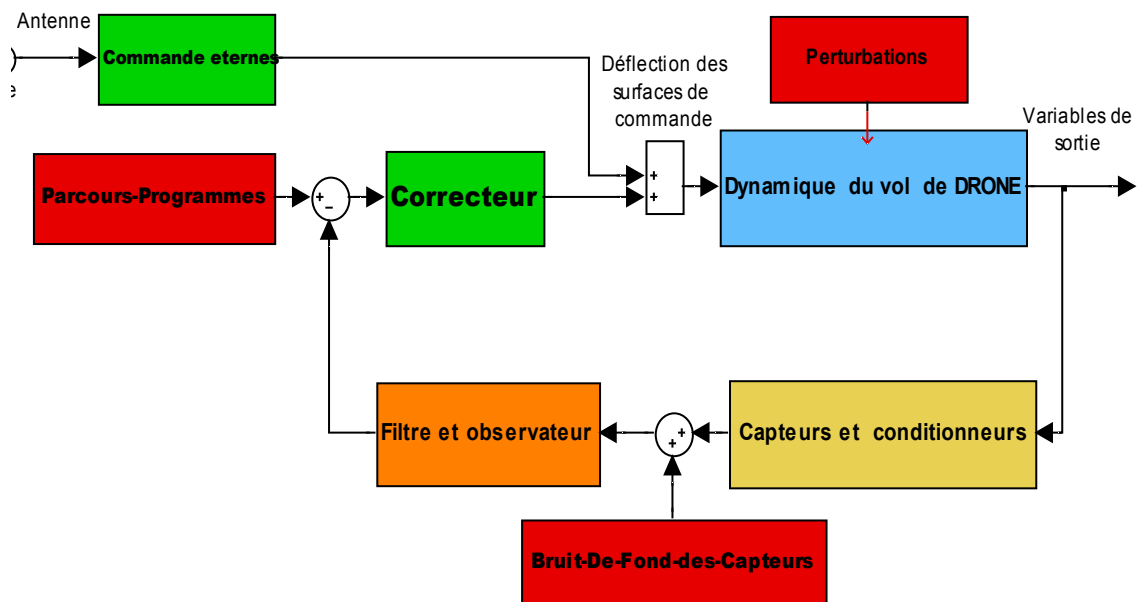


Figure 1.3 : Schéma bloc des diverses interactions de drone

#### - Description de la dynamique du vol :

Nous considérons en premier temps que le drone est un corps rigide ; la distance entre deux points de ce corps ne change pas durant le vol. Le mouvement de cet objet est donc exprimé par six degrés de liberté. Souvent, durant un vol dans l'atmosphère la mesure de mouvement de drone est liée à la terre, et par convention un repère orthogonal  $(O_E X_E Y_E Z_E)$  sera pris sur terre, l'origine de ce repère  $O_E$  qui peut être confondu avec le centre de la terre ou pris sur une certaine surface sur terre [7-8], et les axes sont orientés comme suit :

- $O_E X_E$  vers le nord.
- $O_E Y_E$  vers l'est.
- $O_E Z_E$  vers le bas.

Son utilité principale est l'expression des relations et des effets gravitationnels, tels que l'altitude, le parcours ou la distance horizontale et l'orientation de drone. Un autre est lié à la structure de drone dont l'origine  $O_B$  (B est l'initial de 'Body' dont la signification en Anglais est corps ou structure) est centre de gravité du drone et les divers axes orthogonaux leurs sens sont définis comme suit :

- $O_B X_B$  en avant de drone.
- $O_B Y_B$  vers son tribord (coté droit de drone).
- $O_B Z_B$  vers le bas suivant le vecteur gravitationnel.

La Figure 1.4 montre le système terre-drone et les deux repères

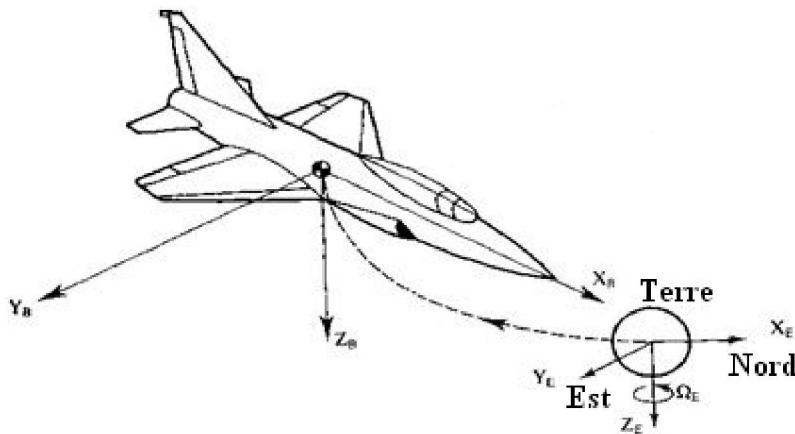


Figure 1.4 : Représentations des deux repères

Les moments, forces et vitesses sont aussi définis. Avec usage de repère fixe sur le centre de gravité, les termes dus à l'inertie apparaissent dans les équations de mouvement et peuvent être considérés constants, en outre, les forces et les moments aérodynamiques dépendent seulement de des angles  $\alpha$  et  $\beta$  (angle d'attaque et angle de dérive horizontales respectivement) qui orientent le vecteur de la vitesse résultant  $V_T$ , par rapport à l'axe  $X_B$ .

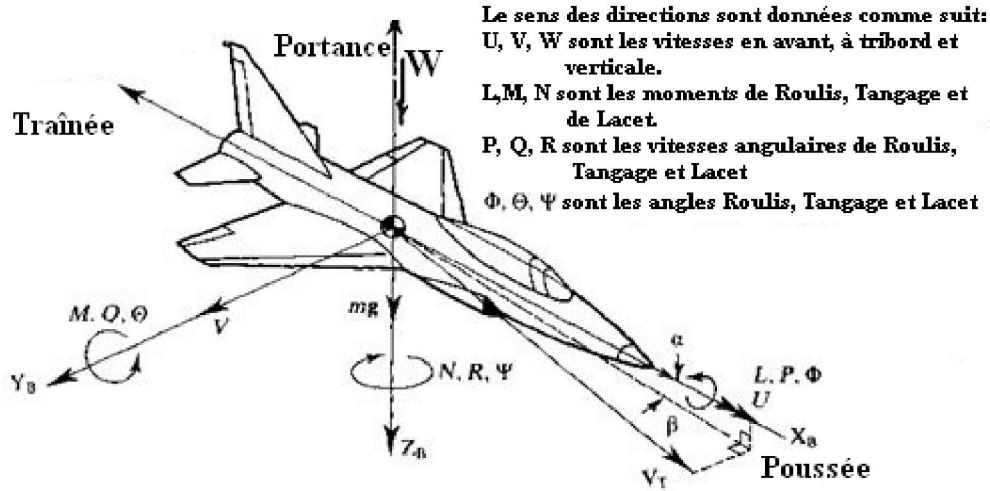


Figure 1.5 : Axes liés au drone

### I. 3. Détermination des équations de mouvement :

En appliquant la deuxième loi de Newton sur ce système rigide, les équations de mouvement peuvent être établies en termes de translation et d'accélération angulaire, dues aux forces et moments appliqués sur le drone [7-10].

#### I.3.1. Mouvement de translation :

Le mouvement de translation est obtenu en appliquant la deuxième loi de Newton, cette loi relie la masse de système, le vecteur de vitesse de son centre de masse et les forces exercées sur l'objet  $F_j$ .

$$F = \frac{d}{dt}(mV_T) \quad (1.3)$$

$$M = \frac{d}{dt}(H) \quad (1.4)$$

Avec  $F$  représente la somme des forces extérieures,  $M$  représente la somme des couples appliqués au système, et  $H$  est le moment angulaire.

La somme des forces externes est composée de trois composantes : gravitationnelles, aérodynamiques et propulsives. Dans la plupart des avions une partie de la poussée est générée en brûlant un carburant qui fait parti de leur masse. Si nous supposons qu'il n'y aura pas de changement au niveau de la source de propulsion, forces aérodynamiques et gravitationnelles autour de point de fonctionnement [8], les équations 1.3 et 1.4 deviennent :

$$F = F_0 + dF = m \frac{d}{dt}(V_T) \quad (1.5)$$

$$M = M_0 + dM = \frac{d}{dt}(H) \quad (1.6)$$

**Nota Bene :** il faut toujours mettre en tête que les grandeurs exprimées ici sont sous forme vectorielle, et le fait d'omettre ces vecteurs est du à la commodité et la simplification des écritures.

L'indexe 0 est attribué au composants d'équilibre, et  $d$  aux petits changements autour de point d'équilibre, et puisque le système est référencé par un repère inertiel sur terre :

$$dF = m \frac{d}{dt} (V_T)_E \quad (1.7)$$

$$dM = \frac{d}{dt} (H)_E \quad (1.8)$$

Par définition, un vol en équilibre ne doit pas manifester des accélérations tout au long de sa trajectoire décrivant une droite ; durant ce vol le vecteur de vitesse relatif au repère fixe est constant, et la vitesse angulaire est nulle. donc  $F_0$  et  $M_0$  sont nuls.

Les deux équations 1.7 et 1.8 requièrent l'invocation des relations de dérivation, pour le mouvement d'un repère dans un autre :

$$F = m \left[ \frac{d}{dt} (V_T) \right]_E = m \left[ \frac{d}{dt} (V_T) \right]_B + w_{B/E} \times V_T \quad (1.9)$$

Où  $w_{B/E}$  est la vitesse angulaire de drone par rapport au repère fixe (terre) et '×' est le produit vectoriel. En exprimant le système dans ce repère fixe à l'aide de coordonnées cartésiennes avec respect des axes fixes  $X_E$ ,  $Y_E$  et  $Z_E$  la vitesse devient:

$$V_T = Ui + Vj + Wk \quad (1.10)$$

$$w = Pi + Qj + Rk \quad (1.11)$$

$$\frac{d}{dt} V_T|_B = \dot{U}i + \dot{V}j + \dot{W}k \quad (1.12)$$

Et le produit vectoriel est défini comme suit :

$$\begin{aligned} w \times V_T &= \begin{vmatrix} i & j & k \\ P & Q & R \\ U & V & W \end{vmatrix} \\ &= i(QW - VR) + j(UR - PW) + k(PV - UQ) \end{aligned} \quad (1.13)$$

Les composantes des petites variations autour du point de fonctionnement peuvent être réécrites d'une façon similaire telle que :

$$dF = dF_x i + dF_y j + dF_z k \quad (1.14)$$

En conséquence,

$$dF = m[(\dot{U} + QW - VR)i + (\dot{V} + UR - PW)j + (\dot{W} + PV - UQ)k] \quad (1.15)$$

A partir de ces équations on peut exprimer les forces sur chaque axe :

$$dF_x = m(\dot{U} + QW - VR) \quad (1.16)$$

$$dF_y = m(\dot{V} + UR - PW) \quad (1.17)$$

$$dF_z = m(\dot{W} + PV - UQ) \quad (1.18)$$

Pour une commodité et simplicité d'écriture des variations  $dF_i$  seront remplacés comme suit :

$$dF_x \equiv dX \quad dF_y \equiv dY \quad dF_z \equiv dZ$$

Il est très important de se rappeler que X,Y et Z symbolisent des *forces*, les équations 1.16-1.18 deviennent :

$$dX = m(\dot{U} + QW - VR) \quad (1.19)$$

$$dY = m(\dot{V} + UR - PW) \quad (1.20)$$

$$dZ = m(\dot{W} + PV - UQ) \quad (1.21)$$

### I.3.2. Mouvement de rotation :

Pour un solide, le moment angulaire peut être défini comme suit :

$$H = I * w \quad (1.22)$$

Et  $I$  est sa matrice d'inertie, qui a comme expression :

$$I = \begin{bmatrix} I_{xx} & I_{xy} & I_{xz} \\ I_{yx} & I_{yy} & I_{yz} \\ I_{zx} & I_{zy} & I_{zz} \end{bmatrix} \quad (1.23)$$

Pour un drone,  $I$  peut s'écrire :

$$I = \begin{bmatrix} I_{xx} & -I_{xy} & -I_{xz} \\ -I_{xy} & I_{yy} & -I_{yz} \\ -I_{xz} & -I_{yz} & I_{zz} \end{bmatrix} \quad (1.24)$$

Avec  $I_{ii}$  signifier un moment d'inertie, et  $I_{ij}$  est le produit inertiel tel que  $i \neq j$ .

En partant de l'équation 1.6 et en introduisant la vitesse de rotation  $w$  nous obtenons :

$$M = \frac{d}{dt}H + w \times H \quad (1.25)$$

Le changement de coordonnées des axes de fuselage par ceux de la terre (pour plus de détails ; voir Gaines and Hoffman, 1972), l'équation 1.25 peut être ré-exprimée comme suit :

$$M = I\left[\frac{d}{dt}w + w \times w\right] + w \times H \quad (1.26)$$

Sachons que le produit vectoriel d'un vecteur par lui-même est un autre vecteur nul, par conséquent :

$$w \times w = 0 \quad (1.27)$$

$$\frac{d}{dt}w = \dot{P}i + \dot{Q}j + \dot{R}k \quad (1.28)$$

Et

$$w \times H = \begin{vmatrix} i & j & k \\ P & Q & R \\ h_x & h_y & h_z \end{vmatrix} \quad (1.29)$$

Les  $h_i$  sont déduits de l'équation 1.22 tels que :

$$hx = I_{xx}P - l_{xy}Q - l_{xz}R \quad (1.30)$$

$$hy = -l_{yx}P + l_{yy}Q - l_{yz}R \quad (1.31)$$

$$hz = -l_{zx}P - l_{zy}Q + l_{zz}R \quad (1.32)$$

Généralement les drones sont symétriques par rapport au plan  $XZ$ , ce qui engendre les conséquences suivantes :

$$I_{xy} = I_{yz} = 0 \quad (1.33)$$

$$hx = I_{xx}P - I_{xz}R \quad (1.34)$$

$$hy = I_{yy}Q \quad (1.35)$$

$$hz = -I_{zx}P + I_{zz}R \quad (1.36)$$

Et

$$dM_x = I_{xx}\dot{P} + I_{zx}(\dot{R} + PQ) + QR(I_{zz} - I_{yy}) \quad (1.37)$$

$$dM_y = I_{yy}\dot{Q} + I_{xz}(P^2 - R^2) + PR(I_{xx} - I_{zz}) \quad (1.38)$$

$$dM_z = I_{zz}\dot{R} - I_{xz}\dot{P} - PQ(I_{xx} - I_{yy}) + I_{zz}QR \quad (1.39)$$

Et suivant toujours la notation américaine :

$$dM_x \equiv dL \quad dM_y \equiv dM \quad dM_z \equiv dN$$

Avec L, M et N sont les moments des axes de roulis, de tangage et de lacet respectivement, les équations 1.37- 1.39 peuvent être réécrites :

$$dL = I_{xx}\dot{P} + I_{zx}(\dot{R} + PQ) + QR(I_{zz} - I_{yy}) \quad (1.40)$$

$$dM = I_{yy}\dot{Q} + I_{xz}(P^2 - R^2) + PR(I_{xx} - I_{zz}) \quad (1.41)$$

$$dN = I_{zz}\dot{R} + I_{xz}\dot{P} - PQ(I_{xx} - I_{yy}) + I_{zz}QR \quad (1.42)$$

### I.3.3. Contribution de la force de la gravitation aux équations de mouvement :

Les forces générées par l'effet gravitationnel sont souvent représentées sur les drones, en négligeant tout changement de champ gravitation avec la variation de l'altitude, et ces variations sont importantes dans le cas où le véhicule est destiné à voler en extra-atmosphère, ce qui nous permet de dire que la gravitation se manifeste juste à travers le centre de gravité de drone. Et puisque les centres de gravité et de masse se coïncident au niveau des drones, il n'aura pas de moment externe produit par la gravité sur le c.g. Et pour elle contribue seulement aux forces externes  $\mathbf{F}$ .

Les forces agissent sur le drone et c'est ce qu'est représenté dans le repère terrestre [8], avec respect des axes de ce repère le point,  $\mathbf{mg}$ , est dirigé tout au long de l'axe  $Z_E$  et de même sens que  $\mathbf{k}$ , la Figure 1.12 montre l'alignement de vecteur de la gravité avec les axes fixes lié au fuselage( centre de gravité), dans cette figure,  $\theta$  représente l'angle entre le vecteur gravitationnel et le plan  $Y_B Z_B$ ; l'angle est positif si le nez est dirigé vers le haut. Et  $\Phi$  représente l'angle entre l'axe  $Z_B$  et la projection de vecteur de la gravitation sur le plan  $Y_B Z_B$ ; l'angle est considéré positif si l'ail droit est inclinée vers le bas.

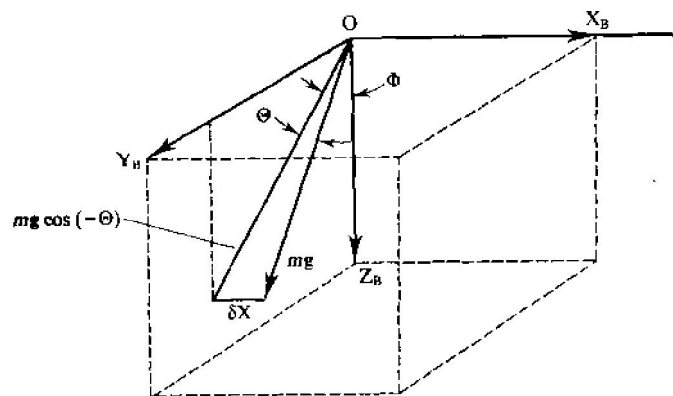


Figure 1.6 : Orientation de vecteur de poids avec les axes du système

La projection de vecteur de poids de drone sur les différents axes liés au fuselage est :

$$\delta X = mg \sin(-\theta) = -mg \sin(\theta) \quad (1.43)$$

$$\delta Y = mg \cos(-\theta) \sin(\Phi) = mg \cos \theta \sin \Phi \quad (1.44)$$

$$\delta Z = mg \cos(-\theta) \cos \Phi = mg \cos \theta \cos \Phi \quad (1.45)$$

Généralement, les angles  $\theta$  et  $\Psi$  ne sont pas directement obtenus par une simple intégration des vitesses angulaires  $P$  et  $Q$  respectivement ; pour cette effet, deux autres variables ont été introduites, et il nécessaire de les lier elles et leurs dérivées avec les vitesses angulaires  $P$ ,  $Q$  et  $R$ . Cette liaison dépend de la façon dont le vecteur gravitationnel est vu par le drone (s'il est fixe ou en rotation).

La Figure 1.13 montre les relations existantes entre les angles et les vitesses angulaires.

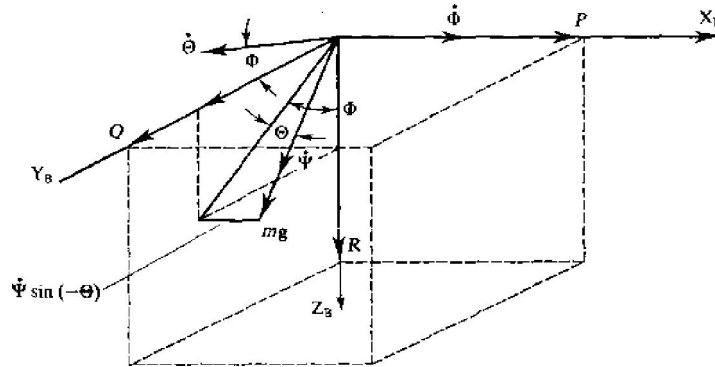


Figure 1.7 : Orientation angulaire et le vecteur de la gravité relatif aux axes de fuselage.

Il semble maintenant qu'il est plus aisé de trouver les expressions des angles :

$$\begin{cases} P = \dot{\Phi} - \dot{\Psi} \sin \theta \\ Q = \dot{\theta} \cos \Phi + \dot{\Psi} \cos \theta \sin \Phi \\ R = -\dot{\theta} \sin \Phi + \dot{\Psi} \cos \theta \cos \Phi \end{cases} \quad (1.46)$$

Par simples calculs, nous arrivons à écrire :

$$\begin{cases} \dot{\Phi} = P + \dot{\Psi} \sin \theta \\ \dot{\theta} = Q \cos \Phi - R \sin \Phi \\ \dot{\Psi} = \frac{R \cos \Phi}{\cos \theta} + \frac{Q \sin \Phi}{\cos \theta} \end{cases} \quad (1.47)$$

Et par un simple remplacement nous obtenons :

$$\dot{\Phi} = P + R \tan \theta \cos \Phi + Q \tan \theta \sin \Phi \quad (1.48)$$

$\Psi, \theta$  et  $\Phi$  sont appelés les angles d'Euler.

**I.3.3.a. Angles d'Euler et axes de transformation:**

A tout instant, les trois angles d'Euler relient l'orientation de solide (fuselage de drone) au repère choisi comme référence. Les Figure 1.14 et 1.15 montrent que les coordonnées de repère de navigation (Terre) sont premièrement transformées via une rotation de l'axe  $Z_B$  ( $Z_n$  et  $n$  pour *Navigation*) par l'angle  $\Psi$  représentant le cap de drone (heading angle), cette rotation est suivie par une autre sur l'axe  $y_1$  par l'angle  $\theta$  représentant le taux de tangage (pitch angle), et en dernier l'angle d'inclinaison des ailes (bank angle)  $\Phi$  définissant la rotation autour de l'axe  $x_2$ . pour une meilleure représentation la Figure 1.10 et 1.9 montre les trois angles et la rotation des axes en trois dimensions [9].

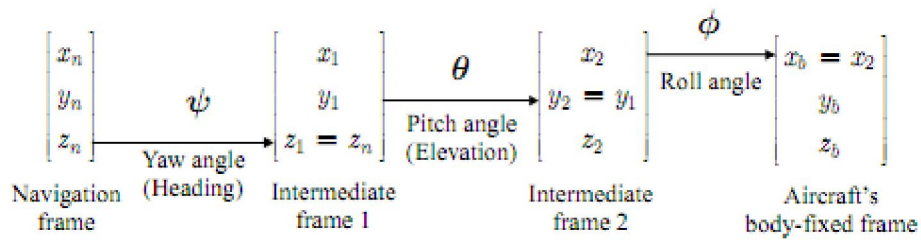


Figure 1.8 : Angles d'Euler

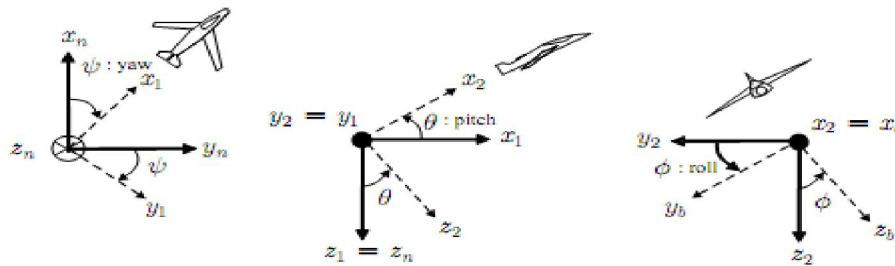


Figure 1.9 : Angles d'Euler et rotation des axes.

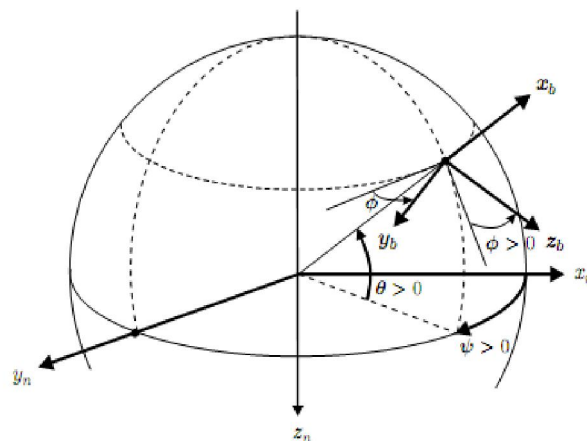


Figure 1.10: Représentation en 3D des angles d'Euler.

**I.3.3.b. Matrice directrice (matrice de transformation) :**

La matrice de transformation est impérativement nécessaire pour la transformation les composantes (coordonnées) des vecteurs et points depuis un repère pris sur un corps solide vers autre pris comme référence et vis versa.

La matrice directrice  $T_E^B$  permet de transformer un vecteur  $A$  exprimé dans le repère référentiel  $A^E$  vers un autre vecteur exprimé dans le repère pris sur le drone  $A^B$  comme suit :

$$A^B = T_E^B A^E \quad (1.49)$$

Avec  $T_E^B = T_\phi T_\theta T_\psi$  et

$$T_\psi = \begin{bmatrix} \cos\psi & \sin\psi & 0 \\ -\sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix}, T_\theta = \begin{bmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{bmatrix} \text{ et } T_\phi = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & \sin\phi \\ 0 & -\sin\phi & \cos\phi \end{bmatrix} \quad (1.50)$$

L'expression de la matrice de transformation est :

$$T_E^B = \begin{bmatrix} \cos\theta \cos\psi & \cos\theta \sin\psi & -\sin\theta \\ \sin\phi \sin\theta \cos\psi - \cos\phi \sin\psi & \sin\phi \sin\theta \sin\psi + \cos\phi \cos\psi & \sin\phi \cos\theta \\ \cos\phi \sin\theta \cos\psi + \sin\phi \sin\psi & \cos\phi \sin\theta \sin\psi - \sin\phi \cos\psi & \cos\phi \cos\theta \end{bmatrix} \quad (1.51)$$

Et la matrice directrice qui transforme un vecteur exprimé dans le repère de navigation vers les nouvelles coordonnées exprimées dans le repère lié au drone est  $T_B^E = (T_E^B)^{-1} = (T_E^B)^T$ .

**I.3.4. Système non linéaire:**

Les équations 1.19-1.21 et 1.40-1.42 représentent les forces d'inertie agissant sur le drone. Les équations 1.43-1.45 représentent la contribution des forces dues à la gravité. Toutes ces forces sont proportionnelles à la masse de drone, en conséquence, ces termes peuvent être réunis pour former une composante d'accélération qui pourrait être mesurée par capteur (un accéléromètre) situé dans le drone tel que ses axes de mesure doivent être confondus avec ceux de fuselage, donc les forces externes peuvent être réécrite comme suit :

$$\begin{cases} X = dX + \delta X \\ Y = dY + \delta Y \\ Z = dZ + \delta Z \end{cases} \quad (1.52)$$

Avec  $\delta X$ ,  $\delta Y$  et  $\delta Z$  sont les termes gravitationnels et  $dX$ ,  $dY$  et  $dZ$  représentent les forces aérodynamiques et la poussée. Et par convention  $dL$ ,  $dM$  et  $dN$  sont simplement symbolisés par

$L$ ,  $M$  et  $N$ . Cependant, les équations régissant le mouvement du drone rigide, pour ses six degrés de liberté sont :

$$\begin{cases} X = m(\dot{U} + QW - VR - g \sin\theta) \\ Y = m(\dot{V} + UR - PW + g \cos\theta \sin\Phi) \\ Z = m(\dot{W} + PV - UQ + g \cos\theta \cos\Phi) \\ L = I_{xx}\dot{P} + I_{zx}(\dot{R} + PQ) + QR(I_{zz} - I_{yy}) \\ M = I_{yy}\dot{Q} + I_{xz}(P^2 - R^2) + PR(I_{xx} - I_{zz}) \\ N = I_{zz}\dot{R} + I_{zz}QR - I_{xz}\dot{P} - PQ(I_{xx} - I_{yy}) \end{cases} \quad (1.53)$$

### I.3.4.a. Linéarisation des termes inertiels et gravitationnels :

Les équations formant 1.53 sont non linéaires ce qui est du à la présence de produit entre deux variables, puissances supérieures à un et d'autres fonctions non linéaires ; la solution d'un tel système ne peut être déterminée analytiquement. Certaines simplifications peuvent se faire autour d'un point de fonctionnement (triming mode), cependant, les variables seront construites à partir de deux autres, une est liée à la valeur de la grandeur à remplacer au point de fonctionnement suffixée par le chiffre 0 et une en minuscule représentant les petites variations autour de ce point, nous prenons à titre d'exemple  $U$ :

$$U = U_0 + u$$

Pour mettre en évidence cette méthode linéarisons la première équation de 1.53 :

$$X = m((U_0 + u) + (Q_0 + q)(W_0 + w) - (V_0 + v)(R_0 + r) - g \sin(\theta_0 + \theta))$$

En négligeant le produit de deux grandeurs en minuscule et en considérant les approximations et les transformations trigonométriques suivantes :

$\sin\xi = 0$  et  $\cos\xi = 1$  pour des valeurs faibles de  $\xi$ . Nous obtiendrons :

$$X = X_0 + x = m[Q_0W_0 - R_0V_0 - g \sin\theta_0] + m[\dot{u} + Q_0w + W_0q - R_0v - V_0r - g \cos\theta_0\theta] \quad (1.54)$$

$$\text{Donc : } \begin{cases} X_0 = m[Q_0W_0 - R_0V_0 - g \sin\theta_0] \\ Y_0 = m[U_0R_0 - P_0W_0 + g \cos\theta_0 \sin\Phi_0] \\ Z_0 = m[P_0V_0 - Q_0U_0 + g \cos\theta_0 \cos\Phi_0] \\ L_0 = Q_0R_0(I_{zz} - I_{yy}) - P_0Q_0 \\ M_0 = (P_0^2 - R_0^2)I_{zx} + (I_{xx} - I_{zz})P_0R_0 \\ N_0 = I_{zx}Q_0R_0 + (I_{xx} - I_{yy})P_0Q_0 \end{cases} \quad (1.55)$$

L'expression des petites variations est :

$$\begin{cases} dx = m[\dot{u} + Q_0 w + W_0 q - R_0 v - V_0 r - (g \cos \theta_0) \theta] \\ dy = m[\dot{v} + U_0 r + R_0 u - W_0 p - P_0 w + (g \cos \theta_0 \cos \Phi_0) \Phi - (g \sin \Phi_0 \sin \theta_0) \theta] \\ dz = m[\dot{w} + V_0 p + P_0 v - U_0 q - Q_0 u - (g \sin \theta_0 \cos \Phi_0) \theta - (g \sin \Phi_0 \cos \theta_0) \Phi] \\ dl = I_{xx} \dot{p} - I_{xz} \dot{r} + (I_{zz} - I_{yy})(Q_0 r + R_0 q) - I_{xz}(P_0 q + Q_0 p) \\ dm = I_{yy} \dot{q} + (I_{xx} - I_{zz})(P_0 r + R_0 p) - (2R_0 r - 2P_0 p) I_{xz} \\ dn = I_{zz} \dot{r} - I_{xz} \dot{p} + (I_{yy} - I_{xx})(P_0 q + Q_0 p) + I_{xz}(Q_0 r + R_0 q) \end{cases} \quad (1.56)$$

Les angles  $\Phi_0, \theta_0$  et  $\Psi_0$  représentent le maintien de cap (pas de changement de direction durant le fonctionnement autour de point d'équilibre), et  $\Psi, \theta$  et  $\Phi$  sont les petites variations des angles d'Euler autour de ce point.

Parfois, il est nécessaire de linéariser les vitesses angulaire, car les équation 1.54-1.56 dépendent de ces variables, et pour les avoir à notre disposition, il suffit de linéariser l'équation 1.46 :

$$\begin{cases} p = \dot{\Phi} - \dot{\Psi} \sin \theta_0 - \theta(\Psi_0 \cos \theta_0) \\ q = \dot{\Phi} \cos \Phi_0 - \theta(\Psi_0 \sin \Phi_0 \sin \theta_0) + \dot{\Psi} \sin \Psi_0 \cos \theta_0 + \Phi(\Psi_0 \cos \theta_0 \cos \Phi_0 - \theta_0 \sin \Phi_0) \\ r = \dot{\Psi} \cos \Phi_0 \cos \theta_0 - \Phi(\Psi_0 \sin \Phi_0 \cos \theta_0 + \Psi_0 \cos \Phi_0) - \dot{\theta} \sin \Phi_0 + \theta(\Psi_0 \sin \theta_0 \cos \Phi_0) \end{cases} \quad (1.57)$$

Il est possible de classifier le vol en trois modes (types) [8] qui sont :

- 1- Vol droit : implique que  $\Psi_0 = 0$ .
- 2- Vol symétrique : implique  $\Psi_0 = V_0 = 0$ .
- 3- Vol avec les ailes au même niveau impliquant  $\Phi_0 = 0$ .

Donc pour un vol droit, symétrique et les ailes au même niveau, les équations représentant le mouvement de translation deviennent :

$$\begin{cases} x = m[\dot{u} + Q_0 w + W_0 q - R_0 v - (g \cos \theta_0) \theta] \\ y = m[\dot{v} + U_0 r + R_0 u - W_0 p - P_0 w + (g \cos \theta_0) \Phi] \\ z = m[\dot{w} + V_0 p + P_0 v - U_0 q - Q_0 u - (g \sin \theta_0) \theta] \end{cases} \quad (1.58)$$

Les équations 1.57 deviennent :

$$\begin{cases} p = \dot{\Phi} - \dot{\Psi} \sin \theta_0 \\ q = \dot{\theta} \\ r = \dot{\Psi} \cos \theta_0 \end{cases} \quad (1.59)$$

Et en prenant toujours le même mode de vol, nous pouvons déduire que  $P_0 = Q_0 = R_0 = 0$ , ce qui nous permet de écrire :

$$\begin{cases} x = m[\dot{u} + W_0q - (g \cos\theta_0)\theta] \\ y = m[\dot{v} + U_0r - W_0p + (g \cos\theta_0)\Phi] \\ z = m[\dot{w} - U_0q - (g \sin\theta_0)\theta] \\ l = I_{xx}\dot{p} - I_{xz}\dot{r} \\ m = I_{yy}\dot{q} \\ n = I_{zz}\dot{r} - I_{xz}\dot{p} \end{cases} \quad (1.58)$$

Une vue bien minutieuse sur cet ensemble d'équation montre que le système peut être séparé en deux, ensemble distincts ; deux descriptions une longitudinale et une autre latérale, et ça peut se faire comme suit :

1- La partie longitudinale :

$$\begin{aligned} x &= m[\dot{u} + W_0q - (g \cos\theta_0)\theta] \\ z &= m[\dot{w} - U_0q - (g \sin\theta_0)\theta] \\ m &= I_{yy}\dot{q} \end{aligned} \quad (1.59)$$

2- La partie latérale :

$$\begin{aligned} y &= m[\dot{v} + U_0r - W_0p + (g \cos\theta_0)\Phi] \\ l &= I_{xx}\dot{p} - I_{xz}\dot{r} \\ n &= I_{zz}\dot{r} - I_{xz}\dot{p} \end{aligned} \quad (1.60)$$

Il faut toujours se rappeler que ce découplage est fait grâce au choix de mode du vol, car durant les vols normaux les grandeurs peuvent être fortement couplées, et la seule façon d'y remédier est d'insérer un correcteur qui, parfois, joue deux rôles ; un pour le découplage et l'autre pour la stabilité.

### I.3.4.c. Linéarisation des forces et moments aérodynamique à l'aide des séries de Taylor :

Le développement de la partie gauche des de mouvement nécessite une méthode spécifique pour sa linéarisation, et nous choisissons le développement en série de Taylor. Pour illustrer cette méthode soit  $x$  une fonction dépendante des variables  $u, v, w, q, p, r, \dot{u}, \dot{v}, \dot{w}, \delta_E$  et  $\dot{\delta}_E$ , ce développement est la somme des produits de la première dérivée de  $x$  par rapport à une des variables et cette variable.

$$x = \frac{dx}{du}u + \frac{dx}{dv}v + \frac{dx}{dw}w + \frac{dx}{dp}p + \frac{dx}{dq}q + \frac{dx}{dr}r + \frac{dx}{d\dot{u}}\dot{u} + \frac{dx}{d\dot{v}}\dot{v} + \frac{dx}{d\dot{w}}\dot{w} + \frac{dx}{d\delta_E}\delta_E + \frac{dx}{d\dot{\delta}_E}\dot{\delta}_E \dots (1.61)$$

Si la variation  $x$  dépend d'autres grandeurs, et leur apport est significatif il est judicieux de les ajouter. Pour l'instant, considérons l'équation 1.59 exprimant le mouvement longitudinal, et l'empennage de profondeur.

$$\begin{aligned} \frac{dx}{du}u + \frac{dx}{dw}w + \frac{dx}{dq}q + \frac{dx}{d\delta_E}\delta_E &= m[\dot{u} + W_0q - (g \cos\theta_0)\theta] \\ \frac{dz}{du}u + \frac{dz}{dw}w + \frac{dz}{dq}q + \frac{dz}{d\delta_E}\delta_E &= m[\dot{w} - U_0q - (g \sin\theta_0)\theta] \\ \frac{dm}{du}u + \frac{dm}{dw}w + \frac{dm}{dq}q + \frac{dm}{d\delta_E}\delta_E &= I_{yy}\dot{q} \end{aligned} \quad (1.62)$$

Et par définition l'altitude est donnée comme suit :

$$h_{cg} = -w + U_0\theta \quad (1.63)$$

Par commodité des écritures, nous proposons de suivre la symbolisation suivante :

$$x_i = \frac{1}{m} \frac{dx}{di}, z_i = \frac{1}{m} \frac{dz}{di} \text{ et } m_i = \frac{1}{I_{yy}} \frac{dm}{di}.$$

A partir des équations 1.59, 1.62 et 1.63 nous pourrions trouver les expressions des grandeurs à choisir pour l'étude des évolutions longitudinales de drone:

$$\begin{aligned} \dot{u} &= x_u u + x_w w + (x_q - W_0)q + (g \cos\theta_0)\theta + x_{\delta_E}\delta_E \\ \dot{w} &= z_u u + z_w w + (z_q + U_0)q + (g \sin\theta_0)\theta + z_{\delta_E}\delta_E \\ \dot{q} &= m_u u + m_w w + m_q q + m_{\delta_E}\delta_E \\ \dot{\theta} &= q \\ \dot{h}_{cg} &= -w + U_0\theta \end{aligned} \quad (1.64)$$

Et des équations 1.60, la translation latérale s'obtiennent en les linéarisant:

$$\begin{aligned} \frac{dy}{dv}v + \frac{dy}{dr}r + \frac{dy}{dp}p + \frac{dy}{d\delta_A}\delta_A + \frac{dy}{d\delta_R}\delta_R &= m[\dot{v} + U_0r - W_0p + (g \cos\theta_0)\Phi] \\ \frac{dl}{dv}v + \frac{dl}{dr}r + \frac{dl}{dp}p + \frac{dl}{d\delta_A}\delta_A + \frac{dl}{d\delta_R}\delta_R &= I_{xx}\dot{p} - I_{xz}\dot{r} \\ \frac{dn}{dv}v + \frac{dn}{dr}r + \frac{dn}{dp}p + \frac{dn}{d\delta_A}\delta_A + \frac{dn}{d\delta_R}\delta_R &= I_{zz}\dot{r} - I_{xz}\dot{p} \end{aligned} \quad (1.65)$$

En adaptant toujours la notation suivante :

$$y_j = \frac{1}{m} \frac{dy}{dj}, l_j = \frac{1}{I_{xx}} \frac{dl}{dj} \text{ et } n_j = \frac{1}{I_{zz}} \frac{dn}{dj} \quad (1.66)$$

Et en incluant l'équation 1.59, nous arrivons à :

$$\begin{aligned}
\dot{v} &= y_v v + (y_r - U_0)r + (y_p + W_0)p + y_{\delta_A} \delta_A + y_{\delta_R} \delta_R - (g \cos\theta_0)\Phi \\
\dot{p} &= \frac{I_{xz}}{I_{xx}} \dot{r} + l_v v + l_r r + l_p p + l_{\delta_A} \delta_A + l_{\delta_R} \delta_R \\
\dot{r} &= \frac{I_{xz}}{I_{zz}} \dot{p} + n_v v + n_r r + n_p p + n_{\delta_A} \delta_A + n_{\delta_R} \delta_R \\
p &= \dot{\phi} - \Psi \sin\theta_0 \\
r &= \dot{\psi} \cos\theta_0
\end{aligned} \tag{1.67}$$

Ces deux modèles linéarisés autour d'un point de fonctionnement (1.63 et 1.65) ci-dessus représentent les modèles de translation longitudinale et latérale respectivement, qui sont exprimés sous la forme  $\dot{x} = Ax + Bu + En$ .

Avec A : matrice d'état.

B : matrice de commande.

E : matrice d'interaction des perturbations dues à des perturbations atmosphériques.

#### I.4. Perturbation atmosphériques :

Durant le déplacement de drone dans l'atmosphère, les masses d'air rencontrées ne sont pas régulières, ce qui induit le drone à effectuer des vols irréguliers et parfois instables, la nature de ces perturbations sont influencées par plusieurs facteurs (densité de l'air, direction et vitesse de vent, altitude, ...). Et vue que les turbulences sont très variées et compliquées, les spécialistes sont mis d'accord que la seule méthode efficace pour l'analyse des problèmes de la dynamique dans telles turbulences est la méthode statistique. Cependant, une large gamme de rafale, qui est déterminée avec précision survient, mais à des instants aléatoires. Estimation des effets de la rencontre de la structure de drone avec les rafales de vent, emploi des rafales discrètes, qui seront classées parmi les phénomènes déterministes.

##### I.4.1. Fonction mathématique d'une rafale discrète (Discret gust function):

Un modèle mathématique à été proposé pour une rafale aiguisée, pour des drones menés des ailes fixes qui défini par :

$$x_g(t) = \frac{k}{T} \left( 1 - \cos\left(\frac{2\pi}{T}t\right) \right) \tag{1.68}$$

Avec : g : pour gust (rafale).

T : pour durée de la rafale donnée par :

$$T = \frac{L}{u_0}$$

L est la longueur d'onde de la rafale en mètre,  $U_0$  est la vitesse de translation (en avant) déterminée pour un certain point de fonctionnement, et k est un facteur de calibrage dont le rôle est de régler l'intensité de la rafale.

#### I.4.2 Représentation continue de la rafale du vent :

Il existe deux représentations analytiques pour la *puissance de la densité spectrale* (PSD Power Spectral Density), des turbulences atmosphériques qui sont de plus en plus utilisées dans la conception des systèmes de commande des drones [7-10]. La première, est le spectre de **Von Kerman**, il est le plus conforme aux spectres des turbulences atmosphériques enregistrés, mais le moins favorable pour les études analytiques, à cause de la complexité de densité spectrale définie comme suit :

$$\Phi_{VK}(\Omega) = \frac{\sigma^2 L}{\pi} \frac{[1 + \frac{8}{3} (1.339L\Omega)^2]}{(1 + (1.339L\Omega)^2)^{\frac{11}{6}}} \quad (1.69)$$

$\Omega$  est la fréquence spatiale en  $rad\ m^{-1}$ .

A cause de la puissance non-entière la fonction de PSD de Von Karman est difficile d'être déterminée.

La seconde est la fonction de PSD de **DRYDEN**, elle est très sollicitée pour sa simplicité et sa facilité à être programmée :

$$\Phi_{DRY}(\Omega) = \frac{\sigma^2 L}{\pi} \frac{[1 + 3L^2\Omega^2]}{(1 + L^2\Omega^2)^2} \quad (1.70)$$

La Figure 1.11 montre les graphes des fonctions des puissances des densités spectrales pour un intervalle de fréquences raisonnable, le différence entre les deux filtres (Von Kerman et Dryden) est très faible, alors, si des hautes performances de drones sont requises, par exemple pour  $0.1 < \omega < 5$ , les deux graphes ne se différencient que de quelques décibels.

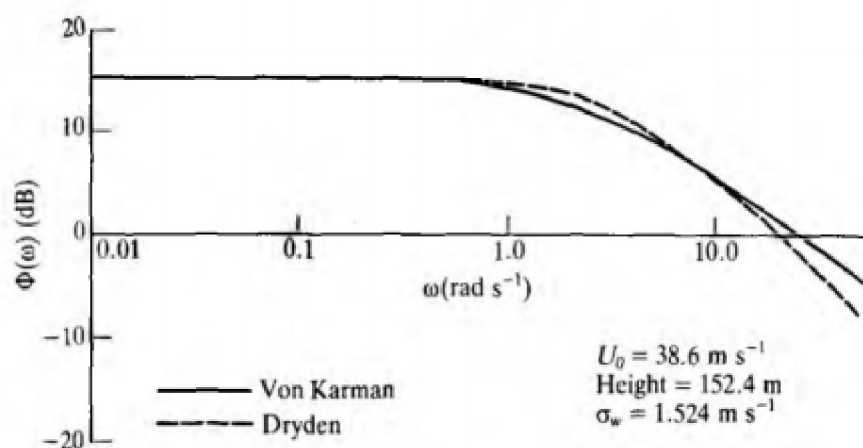


Figure 1.11 : Fonctions PSD de Von Kerman et Dryden

Les expressions des fonctions de PSD de Dryden pour chaque composante sur les trois axes sont :

$$\begin{cases} \Phi_u(\Omega) = 2\sigma_u^2 \frac{L_u}{\pi} \frac{1}{(1+L_u\Omega)^2} \\ \Phi_v(\Omega) = \sigma_v^2 \frac{L_v}{\pi} \frac{[1+3L_v^2\Omega^2]}{(1+L_v^2\Omega^2)^2} \\ \Phi_w(\Omega) = \sigma_w^2 \frac{L_w}{\pi} \frac{[1+3L_w^2\Omega^2]}{(1+L_w^2\Omega^2)^2} \end{cases} \quad (1.71)$$

Avec:  $\Omega = \frac{w}{U_0}$  et  $U_0 = V$

Les variables ;  $b$  représente l'envergure de drone,  $L_u, L_v$  et  $L_w$  représentent les longueurs d'ondes des turbulences sur chaque axe et  $\sigma_u, \sigma_v$  et  $\sigma_w$  leurs intensités.

Il existe des standards pour différentes situations, par exemple, pour un vol dans une tempête, pour toute altitude [11]:

$$\begin{aligned} L_u, &= L_v = L_w = 580 \text{ m} \\ \text{ET} \quad \sigma_u, \sigma_v \text{ et } \sigma_w &= 7 \text{ m s}^{-1} \end{aligned}$$

Les interactions atmosphériques vont se manifester sous forme de perturbations et essentiellement sur les grandeurs vitesse de drone, et puisqu'elles opposent au mouvement, elles seront donc dotées d'un signe négatif. Il est important d'inclure ces perturbations dans les modèles du vol durant la simulation, alors les modèles 1.64 et 1.66 deviennent :

#### I.4.2.a. Modèle longitudinal [9-10]:

$$\begin{cases} \dot{u} = x_u(u - u_g) + x_w(w - w_g) + (x_q - W_0)(q - q_g) + (g \cos\theta_0)\theta + x_{\delta_E}\delta_E \\ \dot{w} = z_u(u - u_g) + z_w(w - w_g) + (z_q + U_0)(q - q_g) + (g \sin\theta_0)\theta + z_{\delta_E}\delta_E \\ \dot{q} = m_u(u - u_g) + m_w(w - w_g) + m_q(q - q_g) + m_{\delta_E}\delta_E \\ \dot{\theta} = q \\ h_{cg} = -w + U_0\theta \end{cases} \quad (1.72)$$

Nous proposons de mettre en premier temps le modèle longitudinal sous la forme usuelle

$$\dot{x} = Ax + Bu + En$$

$$\dot{x}_{Long} = \begin{bmatrix} x_u & x_w & x_q - W_0 & g \cos\theta_0 & 0 & 0 \\ z_u & z_w & z_q + U_0 & g \sin\theta_0 & 0 & 0 \\ m_u & m_w & m_q & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & U_0 & 0 & 0 \\ \Omega_u & \Omega_w & 0 & 0 & \Omega_H & \Omega_\Omega \end{bmatrix} x_{Long} + \begin{bmatrix} x_{\delta_E} & 0 \\ z_{\delta_E} & 0 \\ m_{\delta_E} & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & \Omega_{\delta_{th}} \end{bmatrix} \begin{bmatrix} \delta_E \\ \delta_{th} \end{bmatrix} + \begin{bmatrix} x_u & x_w & x_q - W_0 \\ z_u & z_w & z_q + U_0 \\ m_u & m_w & m_q \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} u_g \\ q_g \\ w_g \end{bmatrix} \quad (1.73)$$

ET

$$x_{Long} = [u \quad w \quad q \quad \theta \quad h \quad \Omega]^T$$

**I.4.2.b. Modèle latéral [9-10] :**

$$\begin{cases} \dot{v} = y_v(v - v_g) + (y_r - U_0)(r - r_g) + (y_p + W_0)(p - p_g) + y_{\delta_A} \delta_A + y_{\delta_R} \delta_R - (g \cos\theta_0)\Phi \\ \dot{p} = \frac{I_{xz}}{I_{xx}} \dot{r} + l_v(v - v_g) + l_r(r - r_g) + l_p(p - p_g) + l_{\delta_A} \delta_A + l_{\delta_R} \delta_R \\ \dot{r} = \frac{I_{xz}}{I_{zz}} \dot{p} + n_v(v - v_g) + n_r(r - r_g) + n_p(p - p_g) + n_{\delta_A} \delta_A + n_{\delta_R} \delta_R \\ p = \dot{\Phi} - \dot{\Psi} \sin\theta_0 \\ r = \dot{\Psi} \cos\theta_0 \end{cases} \quad (1.74)$$

Et de la même manière, en remplaçant et en prenant certaines simplifications nous obtenons le modèle latéral, perturbé.

$$\begin{cases} \dot{v} = y_v(v - v_g) + (y_r - U_0)(r - r_g) + (y_p + W_0)(p - p_g) + y_{\delta_A} \delta_A + y_{\delta_R} \delta_R - (g \cos\theta_0)\Phi \\ \dot{p} = \frac{I_{xz}}{I_{xx}} \left(1 - \frac{I_{xz}}{I_{zz}} \cdot \frac{I_{xz}}{I_{xx}}\right)^{-1} \left( \left(\frac{I_{xz}}{I_{zz}} l_v + n_v\right) (v - v_g) + \left(\frac{I_{xz}}{I_{zz}} l_r + n_r\right) (r - r_g) + \left(\frac{I_{xz}}{I_{zz}} l_p + n_p\right) (p - p_g) \right. \\ \left. + \left(\frac{I_{xz}}{I_{zz}} l_{\delta_A} + n_{\delta_A}\right) \delta_A + \left(\frac{I_{xz}}{I_{zz}} l_{\delta_R} + n_{\delta_R}\right) \delta_R \right) + l_v(v - v_g) + l_r(r - r_g) + l_p(p - p_g) + l_{\delta_A} \delta_A + l_{\delta_R} \delta_R \\ \dot{r} = \left(1 - \frac{I_{xz}}{I_{zz}} \cdot \frac{I_{xz}}{I_{xx}}\right)^{-1} \left( \left(\frac{I_{xz}}{I_{zz}} l_v + n_v\right) (v - v_g) + \left(\frac{I_{xz}}{I_{zz}} l_r + n_r\right) (r - r_g) + \left(\frac{I_{xz}}{I_{zz}} l_p + n_p\right) (p - p_g) \right. \\ \left. + \left(\frac{I_{xz}}{I_{zz}} l_{\delta_A} + n_{\delta_A}\right) \delta_A + \left(\frac{I_{xz}}{I_{zz}} l_{\delta_R} + n_{\delta_R}\right) \delta_R \right) \\ \dot{\Phi} = p + r * tg\theta_0 \\ \dot{\Psi} = \frac{r}{\cos\theta_0} \end{cases} \quad (1.75)$$

Après une simple considération des certains paramètres, nous arrivons au modèle suivant :

$$\begin{bmatrix} \dot{v} \\ \dot{p} \\ \dot{r} \\ \dot{\phi} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} Y_v & Y_p & (Y_r - u_0) & g & 0 \\ L_v & L_p & L_r & 0 & 0 \\ N_v & N_p & N_r & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} v \\ p \\ r \\ \phi \\ \psi \end{bmatrix} + \begin{bmatrix} Y_{\delta a} & Y_{\delta r} \\ L_{\delta a} & L_{\delta r} \\ N_{\delta a} & N_{\delta r} \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \delta_a \\ \delta_r \end{bmatrix} + \begin{bmatrix} Y_v & Y_p & Y_r \\ L_v & L_p & L_r \\ N_v & N_p & N_r \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} v_g \\ p_g \\ r_g \end{bmatrix} \quad (1.76)$$

Qui est mis sous la forme d'état:

$$\dot{x} = A_{Lat}x_{Lat} + B_{Lat}u_{Lat} + H_{Lat}w_{Lat}$$

Tels que :

$$x_{Lat} = [p \ v \ r \ \phi \ \psi]^T$$

### I.4.3. Représentation des interactions atmosphériques :

Pour concevoir un correcteur fiable, il faut inclure le modèle mathématique de l'interaction de l'atmosphère et celui de la dynamique des actionneurs dans les modèles de vol de drone, et la Figure 1.12 montre les interactions atmosphériques. Et pour se faire, commençons par conversion des fonctions de transfert de Dryden en un modèle, et pour une commodité des écritures, réécrivons chacune sous forme de polynômes moniques.

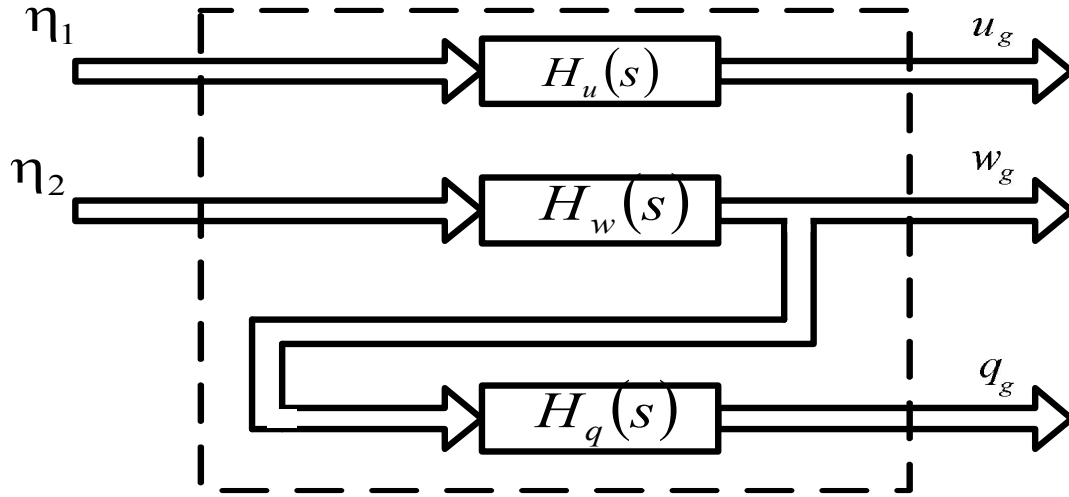


Figure 1.12 : Représentation des interactions atmosphériques selon le modèle de Dryden.

#### I.4.3.a. Modèle longitudinal [11]:

$$H_u(s) = \sigma_u \sqrt{\frac{2L_u}{\pi V}} \cdot \frac{1}{1 + \frac{L_u}{V}s} = \frac{K_u}{T_u} \cdot \frac{1}{s + \frac{1}{T_u}}$$

$$H_w(s) = \sigma_w \sqrt{\frac{L_w}{\pi V}} \cdot \frac{1 + \frac{\sqrt{3}L_w}{V}s}{\left(1 + \frac{L_w}{V}s\right)^2} = \frac{K_w}{T_w} \cdot \frac{(1 + b_w s)}{s^2 + \frac{2}{T_w}s + \frac{1}{T_w^2}}$$

$$H_q(s) = \frac{\frac{s}{V}}{1 + \frac{4b}{\pi V}s} = \frac{b_q}{T_q} \cdot \frac{s}{s + \frac{1}{T_q}} = \frac{b_q}{T_q} \cdot \frac{s}{s + \frac{1}{T_q}} = \frac{b_q}{T_q} - \frac{\frac{b_q}{T_q^2}}{s + \frac{1}{T_q}}$$
(1.77)

Nous n'avons pas précisé la fonction de transfert  $H_w(s)$  à  $H_q(s)$  pour la simple raison que l'entrée du système dont la TF est  $H_q(s)$  sera la sortie de celui (système)  $H_w(s)$ , qui est une équivalence, ce qui est représenté par la Figure 1.1 1.

Avec :

$$K_u = \sigma_u \sqrt{\frac{2L_u}{\pi V}}, \quad T_u = \frac{L_u}{V}$$

$$K_w = \sigma_w \sqrt{\frac{L_w}{\pi V}}, \quad T_w = \frac{L_w}{V}, \quad b_w = \frac{\sqrt{3}L_w}{V}$$

$$T_q = \frac{4b}{\pi V} \quad \text{et} \quad b_q = \frac{1}{V}$$

Donc le modèle d'état suivant est déduit de ces dernières fonctions de transfert :

$$\dot{x}_n(t) = A_n x_n + B_n N$$

Tel que :

$$\begin{bmatrix} \dot{x}_u \\ \dot{x}_{w1} \\ \dot{x}_{w2} \\ \dot{x}_q \end{bmatrix} = \begin{bmatrix} -\frac{1}{T_u} & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & -\frac{1}{T_w^2} & -\frac{2}{T_w} & 0 \\ 0 & \frac{K_w}{T_w^2} & \frac{K_w b_w}{T_w^2} & -\frac{1}{T_q} \end{bmatrix} \begin{bmatrix} x_u \\ x_{w1} \\ x_{w2} \\ x_q \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} N_1 \\ N_2 \end{bmatrix}$$
(1.78)

$$\begin{bmatrix} u_{gust} \\ w_{gust} \\ q_{gust} \end{bmatrix} = \begin{bmatrix} \frac{K_u}{T_u} & 0 & 0 & 0 \\ 0 & \frac{K_w}{T_w^2} & \frac{K_w b_w}{T_w^2} & 0 \\ 0 & \frac{b_q K_w}{T_q T_w^2} & \frac{b_q K_w b_w}{T_q T_w^2} & -\frac{b_q}{T_q^2} \end{bmatrix} \begin{bmatrix} x_u \\ x_{w1} \\ x_{w2} \\ x_q \end{bmatrix}$$

#### I.4.3.b. Modèle latéral [11]:

Le modèle latéral de Dryden sera déduit de la même façon des fonctions de transfert suivantes et de la Figure 1.13:

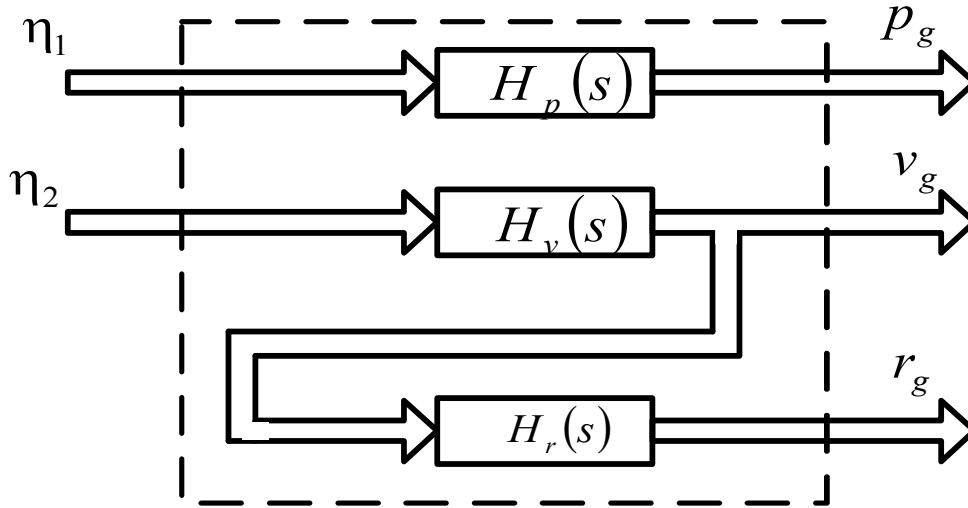


Figure 1.13 : Représentation des perturbations atmosphériques pour le mouvement latéral Selon le modèle de Dryden.

$$\begin{aligned}
 H_p(s) &= \sigma_w \sqrt{\frac{0.8}{V}} \cdot \frac{\left(\frac{\pi}{4b}\right)^{\frac{1}{6}}}{L_w^{1/3} \left(1 + \frac{4b}{V}s\right)} = \frac{K_v}{T_v} \cdot \frac{1}{s + \frac{1}{T_v}} \\
 H_v(s) &= \sigma_v \sqrt{\frac{L_v}{\pi V}} \cdot \frac{1 + \frac{\sqrt{3}L_v}{V}s}{\left(1 + \frac{L_v}{V}s\right)^2} = \frac{K_v}{T_v^2} \cdot \frac{(1 + b_v s)}{s^2 + \frac{2}{T_v}s + \frac{1}{T_v^2}} \\
 H_r(s) &= \frac{\frac{s}{V}}{1 + \frac{3b}{\pi V}s} = \frac{b_r}{T_r} - \frac{\frac{b_r}{T_r^2}}{s + \frac{1}{T_r}}
 \end{aligned} \tag{1.79}$$

Et nous proposons les mêmes notations que précédemment :

$$\begin{aligned}
 K_p &= \sigma_w \left(\frac{\pi}{4bL_w^2}\right)^{\frac{1}{6}} \sqrt{\frac{0.8}{V}}, & T_p &= \frac{4b}{\pi V} \\
 K_v &= \sigma_v \sqrt{\frac{L_v}{\pi V}}, & T_v &= \frac{L_v}{V}, & b_v &= \frac{\sqrt{3}L_v}{V} \\
 T_r &= \frac{3b}{\pi V} \quad \text{et} \quad b_r = \frac{1}{V}
 \end{aligned}$$

$$\begin{bmatrix} \dot{x}_p \\ \dot{x}_{v1} \\ \dot{x}_{v2} \\ \dot{x}_r \end{bmatrix} = \begin{bmatrix} -\frac{1}{T_p} & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & -\frac{1}{T_v^2} & -\frac{2}{T_v} & 0 \\ 0 & \frac{K_v}{T_v^2} & \frac{K_w}{T_v^2} & -\frac{1}{T_r} \end{bmatrix} \begin{bmatrix} x_p \\ x_{v1} \\ x_{v2} \\ x_r \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} N_1 \\ N_2 \end{bmatrix} \tag{1.80}$$

$$\begin{bmatrix} p_{gust} \\ v_{gust} \\ r_{gust} \end{bmatrix} = \begin{bmatrix} \frac{K_p}{T_p} & 0 & 0 & 0 \\ 0 & \frac{K_v}{T_v^2} & \frac{K_v b_w}{T_v^2} & 0 \\ 0 & \frac{b_r K_v}{T_r T_v^2} & \frac{b_r b_v K_v}{T_r T_v^2} & -\frac{b_r}{T_r^2} \end{bmatrix} \begin{bmatrix} x_p \\ x_{v1} \\ x_{v2} \\ x_r \end{bmatrix}$$

#### I.4.4. Modèle des actionneurs :

La dynamique des actionneurs sera modélisée par un système de premier ordre avec une constante de temps  $T$  avec prise de compte des saturations [12], la Figure 1.13 les schématise, dont la fonction de transfert est :

$$G(s) = \frac{1}{Ts+1} \quad (1.81)$$

Ou encore :

$$\dot{x}_a = A_a x_a + B_a U_a$$

$a$  pour actionneurs.

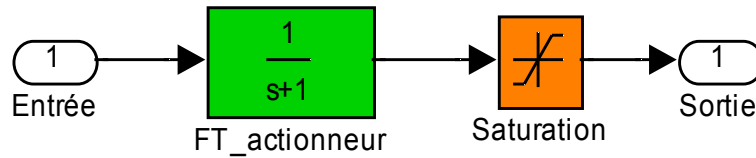


Figure 1.13 : Schématisation des actionneurs

La dimension de  $A_a$  dépend essentiellement de nombre d'actionneurs et leur dynamique, et puisque nous avons spécifié qu'elle est représentée à laide d'une fonction de transfert de premier ordre, donc sa dimension est la même que le nombre d'actionneurs.

Donc :

$$\begin{bmatrix} \dot{x}_{a1} \\ \dot{x}_{a2} \end{bmatrix} = \begin{bmatrix} -\frac{1}{T_{a1}} & 0 \\ 0 & -\frac{1}{T_{a2}} \end{bmatrix} \begin{bmatrix} x_{a1} \\ x_{a2} \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} U_{a1} \\ U_{a2} \end{bmatrix} \quad (1.82)$$

$$\begin{bmatrix} y_{a1} \\ y_{a2} \end{bmatrix} = \begin{bmatrix} \frac{1}{T_{a1}} & 0 \\ 0 & \frac{1}{T_{a2}} \end{bmatrix} \begin{bmatrix} x_{a1} \\ x_{a2} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} U_{a1} \\ U_{a2} \end{bmatrix}$$

Le nouveau système drone-atmosphère et actionneurs sera augmenté de deux variables dues aux actionneurs et de quatre autres dues au modèle atmosphérique, il devient alors :

$$\begin{bmatrix} \dot{x}_d \\ \dot{x}_n \\ \dot{x}_a \end{bmatrix} = \begin{bmatrix} A & HC_n & BC_a \\ 0 & A_n & 0 \\ 0 & 0 & A_a \end{bmatrix} \begin{bmatrix} x_d \\ x_n \\ x_a \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ B_n & 0 \\ 0 & B_a \end{bmatrix} \begin{bmatrix} N \\ U_a \end{bmatrix} \quad (1.83)$$

Et les matrices  $A_e = \begin{bmatrix} A & HC_n & BC_a \\ 0 & A_n & 0 \\ 0 & 0 & A_a \end{bmatrix}$ , ET  $B_e = \begin{bmatrix} 0 & 0 \\ B_n & 0 \\ 0 & B_a \end{bmatrix}$  sont appelées **matrices étendues**.

Le modèle utilisé durant l'étude de la procédure proposée dans le mémoire est nommé *Aerosonde Mark 3*, qui est utilisé par *Australian Meteorological Bureau* pour l'étude des phénomènes météorologiques. D'ailleurs, en 1998 l'université de Taiwan a acheté huit modèles pour les utiliser dans la recherche scientifique. Ça est dû à leur fiabilité, son modèle et ses caractéristiques sont donnés par la figure 1.14 et le tableau 1 suivants :

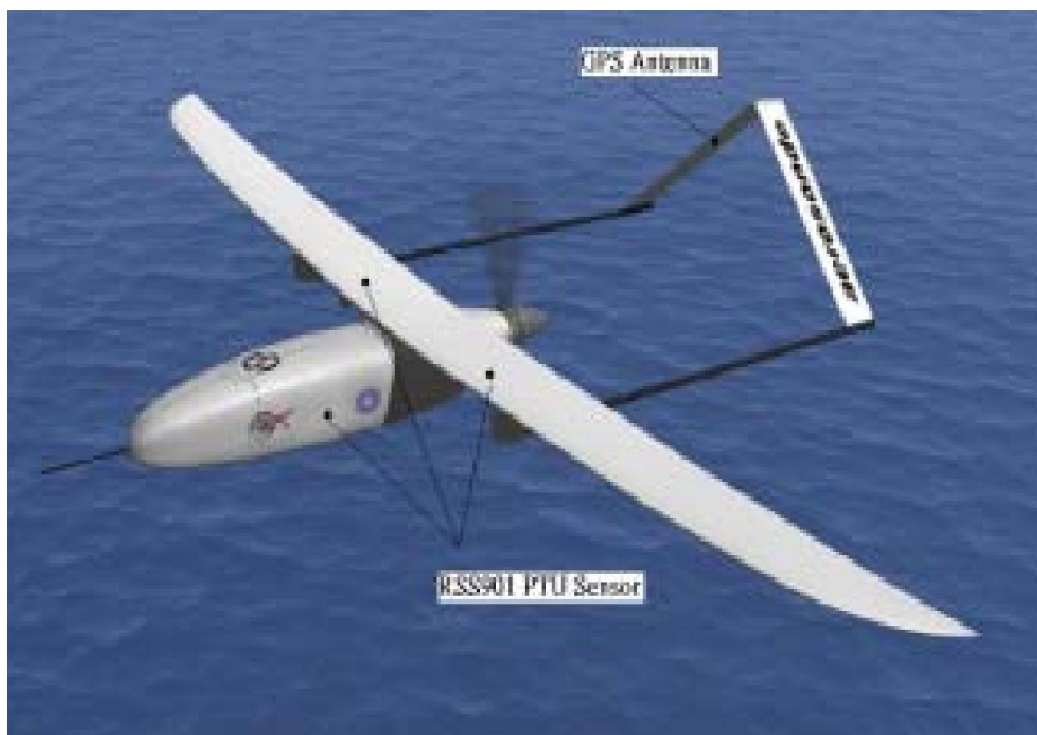


Figure 1.14: Aérosonde et ses capteurs (utilisé pour l'étude des typhons).

Caractéristiques	
Poids, Envergure	13-15 kg, 2.9 m
Moteur	24 cc, Injection en essence de type H.
Poids du carburant Max	5 Kg
Navigation	GPS
Distance max de communication Bonde UHF	180 Km
Energie générée à bord	Générateur Maxon délivrant 18 v DC et 1 Amp, 40 W.
Temps de vol maximal	250 heures
Performances	
Vitesse de: translation, élévation	Translation 18-32 ms <sup>-1</sup> , élévation > 2.5 ms <sup>-1</sup>
Endurance	20 à30 h, 2000 to 3000 Km.
Altitude	50 m -> 7000 m
Charge utile	Max 5 Kg~ 10 heures d'endurance. Max 2 Kg ~ 30 heures d'endurance.

Tableau 1 : Caractéristiques d'Aérosonde Mark 3

Le modèle nominal et ceux perturbés sont définis [12-14] comme suit :

**Modèles longitudinaux :**

- Modèle nominal  $V=30$  m/s:

$$A = \begin{bmatrix} -0.293 & 0.38 & -0.55 & -9.78 & 0 & 0.01 \\ -0.55 & -5.36 & 30 & -0.18 & 0 & 0 \\ 0.33 & -5.63 & -6.19 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0.01 & -1 & 0 & 30 & 0 & 0 \\ 41.53 & 0.78 & 0 & 0 & -0.63 & -3.85 \end{bmatrix}; B = \begin{bmatrix} -0.3 & 0 \\ -3.7 & 0 \\ -50 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 2664 \end{bmatrix}$$

$$G = \begin{bmatrix} 0.293 & -0.38 & 0.55 \\ 0.55 & 5.36 & 30 \\ 0.33 & 5.36 & 6.19 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (1.84)$$

- Modèle perturbé  $V=25$  m/s<sup>2</sup> :

$$\begin{aligned}
 Alon_{p1} &= \begin{bmatrix} -0.24 & 0.53 & -1.19 & -9.8 & 0 & 0.0116 \\ -0.59 & -4.47 & 25 & -0.47 & 0 & 0 \\ 0.43 & -4.83 & -5.15 & 0 & 0 & -0.009 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0.04 & -0.99 & 0 & 25 & 0 & 0 \\ 34.66 & 1.68 & 0 & 0 & -0.03 & -3.23 \end{bmatrix}; Blon_{p1} = \begin{bmatrix} 0.35 & 0 \\ -2.55 & 0 \\ -35.2 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 389.6 \end{bmatrix}; \\
 Glon_{p1} &= \begin{bmatrix} -0.24 & 0.53 & -1.19 \\ 0.59 & -4.47 & 25 \\ 0.43 & -4.83 & -5.15 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{1.85}
 \end{aligned}$$

- Modèle perturbé  $V=35 \text{ m/s}^2$  :

$$\begin{aligned}
 Alon_{p2} &= \begin{bmatrix} -0.35 & 0.28 & -0.058 & -9.81 & 0 & 0.016 \\ -0.55 & -6.25 & 35 & -0.016 & 0 & 0 \\ 0.28 & -6.43 & -7.21 & 0 & 0 & -0.012 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0.001 & -1 & 0 & 35 & 0 & 0 \\ 48.48 & 0.07 & 0 & 0 & -0.78 & -4.43 \end{bmatrix}; Blon_{p2} = \begin{bmatrix} -0.5 & 0 \\ -5 & 0 \\ -68 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 3040 \end{bmatrix}; \\
 Glon_{p2} &= \begin{bmatrix} -0.35 & 0.28 & -0.058 \\ -0.55 & -6.25 & 35 \\ 0.28 & -6.43 & -7.21 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{1.86}
 \end{aligned}$$

### 1- Modèles latéraux :

- Modèle nominal  $V=30 \text{ m/s}$ :

$$\begin{aligned}
 Alat &= \begin{bmatrix} -0.8297 & 0.5669 & -30 & 9.78 & 0 \\ -5.48 & -26.97 & 12.98 & 0 & 0 \\ 0.89 & -3.5 & -1.35 & 0 & 0 \\ 0 & 1 & 0.01 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}; Blat = \begin{bmatrix} -2.13 & 5.44 \\ -187.35 & 3.37 \\ -7.38 & -34.41 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}
 \end{aligned}$$

$$Glat = \begin{bmatrix} 0.8297 & -0.5669 & 0 \\ 5.48 & 26.97 & 0 \\ -0.89 & 3.5 & 1.35 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (1.87)$$

- Modèle perturbé  $V=25 \text{ m/s}^2$  :

$$Alat_{p1} = \begin{bmatrix} -0.69 & 1.21 & -25 & 9.80 & 0 \\ -4.55 & -22.42 & 10.79 & 0 & 0 \\ 0.74 & -2.9 & -1.12 & 0 & 0 \\ 0 & 1 & 0.04 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}; \quad Blat_{p1} = \begin{bmatrix} -1.47 & 3.77 \\ -129.7 & 2.35 \\ -5.11 & -23.83 \\ 0 & 0 \\ 0 & 0 \end{bmatrix};$$

$$Glat_{p1} = \begin{bmatrix} -0.69 & 1.21 & -25 \\ -4.55 & -22.42 & 10.79 \\ 0.73 & -2.9 & -1.12 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (1.88)$$

- Modèle perturbé  $V=35 \text{ m/s}^2$  :

$$Alat_{p2} = \begin{bmatrix} -0.97 & 0.05 & -35 & 9.81 & 0 \\ -6.38 & -31.43 & 15.13 & 0 & 0 \\ 1.03 & -4.07 & -1.58 & 0 & 0 \\ 0 & 1 & 0.0016 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}; \quad Blat_{p2} = \begin{bmatrix} -2.9 & 7.4 \\ -254.7 & 4.58 \\ -10 & -46.79 \\ 0 & 0 \\ 0 & 0 \end{bmatrix};$$

$$Glat_{p2} = \begin{bmatrix} -0.97 & 0.05 & -35 \\ -6.38 & -31.43 & 15.13 \\ 1.03 & -4.07 & -1.58 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (1.89)$$

## Conclusion

Dans ce chapitre, nous avons présenté le DRONE avec ses surfaces de commande et comme système de commande. Également, la modélisation et la mise sous forme d'un modèle d'état non-linéaire de la dynamique de vol ont été obtenues, dans le but de le linéariser, ce qui nous a permis séparer les dynamiques longitudinale et latérale, et c'est ainsi que nous sommes arrivés à deux modèles linéaires de sixième et cinquième ordre, respectivement.

# CHAPITRE II

**CHAPITRE II****CONCEPTION DE LA LOI DE COMMANDE MULTIVARIABLE BASANT SUR  
L'OPTIMISATION ROBUSTE**

La propriété la plus importante pour la construction de la loi de commande est la robustesse, vu que de tels véhicules sont soumis à la variation des paramètres internes comme aux perturbations externes durant leurs missions de vol, où plusieurs contraintes doivent être prises en compte au niveau matériels et logiciels embarqués pour réduire au mieux la charge utile du drone [12-16].

Ce chapitre est consacré à l'optimisation paramétrique. La procédure est divisée en deux phases; la première tâche est la détermination de la commande optimale de système multi variables basée sur le théorème de séparation et la deuxième est l'optimisation robuste basée sur les algorithmes génétiques. Le théorème de séparation ; qui est la base de la construction de la loi de commande ; à pour condition deux niveaux [21-25] ; le premier est de concevoir un observateur pour reconstruire toutes les variables d'état du système, et le deuxième est la détermination d'un correcteur en utilisant un retour d'état statique mais d'une manière optimale (LQR). La construction de l'observateur est nécessaire pour estimer le vecteur d'état ; dans le domaine de commande des drones c'est une approche très attractive, pour réduire au minimum le nombre des capteurs à bord. La construction de la loi de commande passe par les étapes suivantes :

1. La synthèse de retour optimal du système multi variables basée sur le théorème de séparation ;
2. Robustifier la loi de commande en se basant sur les algorithmes génétiques ;
3. Simulation du système en boucle fermée pour le mouvement longitudinal et latéral.

Il est prouvé que, la synthèse de la commande optimale des systèmes Multi variables (MIMO) basée sur le théorème de séparation ne traite pas les performances de stabilité et celles de la robustesse en présence des perturbations internes et/ou externes [26-27]. En conséquence, la conception de la loi de commande pour le modèle nominal, ne procure pas les mêmes performances, si les paramètres du système changent. Pour surmonter ce problème, il est primordial de robustifier le correcteur, ce qui sera accompli par la procédure d'optimisation robuste  $H_2/H_\infty$ . L'étude de cette méthode a fourni de bons résultats pour les systèmes à une seule entrée et plusieurs sorties (SIMO) [16-17] en utilisant les méthodes traditionnelles d'optimisation ; dans ce document

---

nous traiterons un système multi variables (multi entrées et multi sorties MIMO), et en plus la procédure d'optimisation utilisée est les algorithmes génétiques avec les paramètres d'optimisation le gain de l'observateur "L" et le gain de retour d'état K.

Il est important de bien noter que ce chapitre est partitionné en deux parties pour la génération de la commande ; les deux sont adoptées pour la restitution de vecteur d'état. Dans le premier cas ; le filtre de Kalman sera mis en œuvre pour prévenir la contamination du système par les bruit de mesure en boucle fermée. Les signaux bruités utilisés par ce filtre ; pour la reconstruction de vecteur d'état ; sont affectés par les bruits de fond des capteurs. Les perturbations ; qui sont dues aux turbulences de vent et de l'atmosphère ; sont approximées par le modèle de Dryden. Les entrées du modèle de perturbation sont des bruits blanc et ses sorties sont des bruits colorés. Il sera mis en évidence que l'usage du filtre de Kalman pour l'estimation de vecteur d'état augmente l'ordre de correcteur.

L'optimisation robuste  $H_2/H_\infty$  consiste à satisfaire des multi-objectifs et compromis entre des objectifs opposés performances et robustesse. Il est connu que les AG sont développés pour résoudre des problèmes difficiles tout en ayant une fonction objectif dont les propriétés ne sont pas à ravir telles que la non-continuité, non-dérivabilité...etc). Fondés sur la sélection naturelle et la génétique ; les AG font parti d'une classe de recherches stochastique dans le domaine des algorithmes évolutionnaires (AE) ou Calculs évolutionnaires (CE)) Ces algorithmes manipulent et maintiennent une famille, ou une population d'individus et appliquent " plusieurs arrangements et stratégies " pour la recherche d'une meilleure solution. Les AG permettent de parcourir plusieurs régions d'intérêt au même temps ce qui promet une plus grande chance d'atteindre l'optimum global [28-29].

### II.1. Synthèse de correcteur optimal basé sur le théorème de séparation dans le cas multi variables.

Le modèle d'état continu de Drone est donné par le quintuple matriciel [A,B,C,D,G], où :  
 $A \in \mathbb{R}^{n \times n}$ ,  $B \in \mathbb{R}^{n \times q}$ ,  $C \in \mathbb{R}^{p \times n}$ ,  $D \in \mathbb{R}^{p \times q}$ ,  $G \in \mathbb{R}^{n \times 1}$ , défini comme suit :

$$\begin{cases} \dot{X}(t) = AX(t) + BU(t) + Gw(t) \\ Y(t) = CX(t) + DU(t) + v(t) \end{cases} \quad (2.1)$$

Avec  $X(t)$  est le vecteur d'état de dimension  $(n \times 1)$ ,  $A$  est la matrice d'état de dimension  $(n \times n)$ ,  $U(t)$  est le vecteur de commande de dimension  $(1 \times q)$ ,  $B$  est la matrice de commande  $(n \times q)$ ,  $C$  matrice d'observation,  $D$  est la matrice d'intervention directe de l'entrée sur la sortie,  $G$  représente la matrice de connexion entre le modèle de perturbation et celui du Drone,  $y(t)$  est le vecteur de mesure en sortie de dimension  $(p \times 1)$ ,  $v(t)$  est le vecteur de bruit de mesure des capteurs et le vecteur  $w(t)$  représente le processus des turbulences (bruit coloré), ici il est donné par les perturbations dues à l'environnement décrit par les sorties de filtre de Dryden. Comme mentionné dans l'introduction, pour améliorer la construction de correcteur basé sur le théorème de séparation en utilisant le filtre de Kalman, il serait nécessaire d'augmenter le modèle d'état de drone par celui des turbulences (Filtre de Dryden) pour former le modèle étendu donné par l'équation (2.3) Soit le quadruple  $[A_{dr}, B_{dr}, C_{dr}, D_{dr}]$  les matrices formant le modèle d'état de filtre tels que :

$$\begin{cases} \dot{X}_{dr}(t) = A_{dr} X_{dr}(t) + B_{dr} \eta(t) \\ Y_{dr}(t) = C_{dr} X_{dr}(t) + D_{dr} \eta(t) \end{cases} \quad (2.1)$$

Où  $A_{dr} \in \mathbb{R}^{r \times r}$ ,  $B_{dr} \in \mathbb{R}^{r \times 2}$ ,  $C_{dr} \in \mathbb{R}^{3 \times r}$ ,  $D_{dr} \in \mathbb{R}^{3 \times 2}$  Ce filtre possède deux entrées  $\eta(t) = [\eta_1(t) \quad \eta_2(t)]$ : horizontale et verticale dues aux rafales de vent et trois sorties : vitesse de turbulence longitudinale ( $u_g(t)$ ), vitesse de turbulence verticale ( $w_g(t)$ ) et le ( $q_g(t)$ ) (taux de variation du tangage ; en anglais pitch rate. Pour le mouvement latéral les entrées sont les rafales de vent latérale et verticale, les sorties sont : vitesse de turbulence latérale ( $v_g(t)$ ), le taux de variations du roulis (roll rate) des turbulences ( $r_g(t)$ ) et le taux variations lacet (yaw rate) des turbulences ( $p_g(t)$ ).

Après avoir connecté le modèle des turbulences à celui du drone, le modèle étendu est donné par :

$$\begin{cases} \begin{bmatrix} \dot{X}(t) \\ \dot{X}_{dr}(t) \end{bmatrix} = \begin{bmatrix} A & GC_{dr} \\ 0_{r \times n} & A_{dr} \end{bmatrix} \begin{bmatrix} X(t) \\ X_{dr}(t) \end{bmatrix} + \begin{bmatrix} B & GB_{dr} \\ 0_{r \times q} & B_{dr} \end{bmatrix} \begin{bmatrix} U(t) \\ \eta(t) \end{bmatrix} \\ Y_{ex}(t) = \begin{bmatrix} C & 0_{p \times r} \end{bmatrix} \begin{bmatrix} X(t) \\ X_{dr}(t) \end{bmatrix} + \begin{bmatrix} D & 0_{p \times 2} \end{bmatrix} \begin{bmatrix} U(t) \\ \eta(t) \end{bmatrix} + v(t) \end{cases} \quad (2.3)$$

Ou courtement :

$$\begin{cases} \dot{X}_{ex}(t) = A_{ex} X_{ex}(t) + B_u U(t) + B_s \eta(t) \\ Y_{ex}(t) = C_{ex} X_{ex}(t) + D_u U(t) + D_s \eta(t) + v(t) \end{cases} \quad (2.4)$$

Avec :  $B_u = \begin{bmatrix} B & 0_{r \times q} \end{bmatrix}^T$ ,  $D_u = D$ ,  $B_s = \begin{bmatrix} GB_{dr} & B_{dr} \end{bmatrix}^T$  and  $D_s = 0_{p \times 2}$ .

La synthèse de correcteur numérique pour le système décrit dans (2.4) se fait par le passage dans le domaine discret, en l'échantillonnant d'une période d'échantillonnage adéquate  $T_s = 100$  ms; le modèle discret équivalent de modèle étendu par la représentation d'état suivante :

$$\begin{cases} X_d(n+1) = A_d X_d(n) + B_d U(n) + B_{sd} \eta(n) \\ Y_d(n) = C_d X_d(n) + D_d U(n) + D_{sd} \eta(n) + v(n) \end{cases} \quad (2.5)$$

A ce stade ; nous pouvons appliquer le théorème de séparation, pour la synthèse du correcteur optimal pour le système (2.5), composé de l'observateur (filtre de Kalman) pour pouvoir reconstruire tout le vecteur d'état, en minimisant la fonction stochastique donnée par (2.6) et en déterminant le retour d'état optimal basé sur le correcteur linéaire quadratique (Linear quadratic regulator LQR).

$$J_s = \lim_{T \rightarrow \infty} \sum_{n=0}^T E \left( X^T(n) R_\eta X(n) + U^T(n) R_v U(n) \right) \quad (2.6)$$

$R_\eta$  et  $R_v$  sont les matrices de variance associées aux processus de perturbations et bruits de mesure des capteurs, respectivement, E représente la valeur attendue ( Espérance mathématique).

Dans la section qui suit le filtre discret de Kalman est construit pour le modèle (2.5), pour estimer la totalité de vecteur d'état.

Définissons les matrices covariance suivantes :

$$\begin{aligned} E\{ \eta(n) \} &= 0 & E\{ v(n) \} &= 0 \\ E\{ \eta(n) \eta(n)^T \} &= R_\eta & E\{ v(n) v(n)^T \} &= R_v \\ E\{ \eta(n) v(n)^T \} &= 0 & E\{ X(0) \} &= X_0 \end{aligned}$$

Les états initiaux sont supposés être aléatoires et non-corrélés avec les processus de perturbation  $\eta(n)$  et les bruits de mesure  $v(n)$ .

Le modèle discret de filtre de Kalman résultant est défini par le bloc matriciel suivant :

$$\left[ \begin{array}{c|c} \mathbf{A}_{\text{kal}} & \mathbf{B}_{\text{kal}} \\ \hline \mathbf{C}_{\text{kal}} & \mathbf{D}_{\text{kal}} \end{array} \right] = \left[ \begin{array}{c|cc} \mathbf{A}_d - \mathbf{L}\mathbf{C}_d & \mathbf{B}_d - \mathbf{L}\mathbf{D}_d & \mathbf{L} \\ \hline \mathbf{C}_d(\mathbf{I} - \mathbf{M}\mathbf{C}_d) & (\mathbf{I} - \mathbf{C}_d\mathbf{M})\mathbf{D}_d & \mathbf{C}_d\mathbf{M} \\ \mathbf{I} - \mathbf{M}\mathbf{C} & -\mathbf{M}\mathbf{D}_d & \mathbf{M} \end{array} \right]$$

Où  $\mathbf{M}$  est une matrice de gain innovatrice définie par :

$$\mathbf{M} = \mathbf{P}\mathbf{C}_d^T (\mathbf{C}_d\mathbf{P}\mathbf{C}_d^T + \mathbf{R}_v)^{-1}$$

$\mathbf{L}$  est la matrice de gain de Kalman donnée par :

$$\mathbf{L} = \mathbf{A}_d\mathbf{P}\mathbf{C}_d^T (\mathbf{C}_d\mathbf{P}\mathbf{C}_d^T + \mathbf{R}_v)^{-1}$$

$\mathbf{P}$  est la solution de l'équation algébrique à temps discret de Riccati associée au filtre de Kalman, qui minimise l'erreur calculée dans le plant d'état comme étant l'écart entre la valeur de vecteur d'état actuel et celle prédite  $\mathbf{e}(n) = \mathbf{X}_d(n) - \mathbf{X}_d(n/n-1)$  :

$$\mathbf{P} = \lim_{n \rightarrow \infty} \mathbf{E}(\mathbf{e}(n/n-1)\mathbf{e}(n/n-1)^T)$$

L'équation algébrique de Riccati en temps discret est donnée par :

$$\mathbf{A}_d^T \mathbf{P} \mathbf{A}_d - \mathbf{P} - \mathbf{A}_d^T \mathbf{P} \mathbf{B}_{sd} (\mathbf{B}_{sd}^T \mathbf{P} \mathbf{B}_{sd} + \mathbf{R}_\eta)^{-1} \mathbf{B}_{sd}^T \mathbf{P} \mathbf{A}_d + \mathbf{R}_v = 0$$

Après avoir reconstruit la totalité de vecteur d'état, la loi de commande est alors construite à l'aide d'un retour d'état optimal en temps discret (ou Discrete Linear Quadratic Regulator DLQR), qui consiste à trouver une loi de commande qui minimise le critère de performance donné par l'expression suivante;

$$\mathbf{J} = \sum_{n=1}^{\infty} \tilde{\mathbf{X}}_d(n)^T \mathbf{Q} \tilde{\mathbf{X}}_d(n) + \mathbf{U}(n)^T \mathbf{R} \mathbf{U}(n) \quad (2.7)$$

Où  $\mathbf{Q}$  et  $\mathbf{R}$  sont des matrices de pondération des vecteurs d'état et de commande, respectivement.  $\tilde{\mathbf{X}}_d$  est le vecteur d'état reconstruit.

Il est montré que la loi de commande est un retour d'état statique donnée par :

$$\mathbf{U}(n) = -\mathbf{F} \tilde{\mathbf{X}}_d(n) \quad (2.8)$$

La matrice de gain  $F$  est calculée d'une manière optimale dont l'expression est la suivante:

$$F = \left( B_d^T S B_d + R \right)^{-1} B_d^T S A_d$$

Avec  $S$  est la solution d'équation algébrique de Riccati à temps discret associée au correcteur :

$$A_d^T S A_d - S - A_d^T S B_d \left( B_d^T S B_d + R \right)^{-1} B_d^T S A_d + Q = 0$$

Le modèle d'état de système en boucle fermée, après avoir connecté le filtre de Kalman au correcteur linéaire quadratique (LQ), est exprimé comme suit :

$$\begin{cases} \dot{X}_d(n+1) = A_d X_d(n) - B_d F \tilde{X}_d(n) \\ \tilde{X}_{ex}(n+1) = (A_d - L C_d - B_d F) \tilde{X}_d(n) + L Y_d(n) \\ U(n) = -F \tilde{X}_d(n) \end{cases} \quad (2.9)$$

La structure de correcteur connecté au plan de contrôle (système) est montré dans la figure (2.1) :

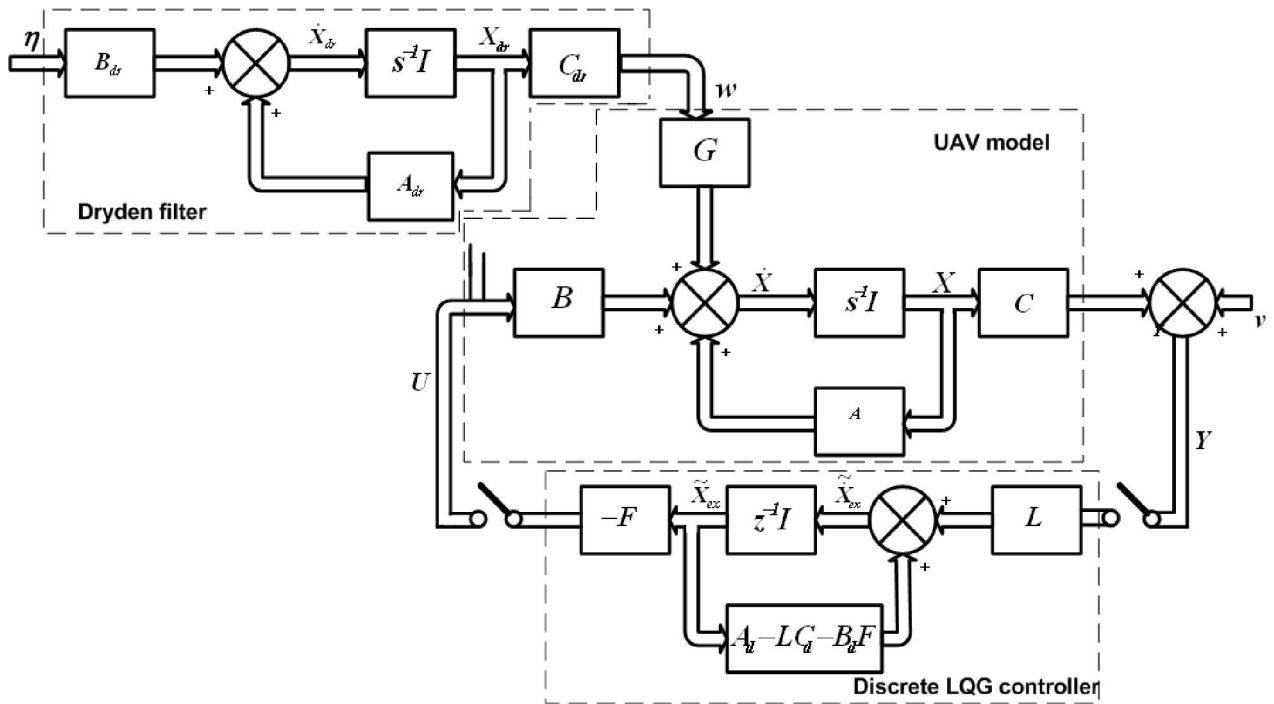


Figure 2.1 : Commande optimale en boucle fermée.

**II.1.1. Robustification paramétrique de correcteur LQG (Linear Quadratic Gaussian) :**

La robustesse des systèmes de commande de vol est la propriété plus cruciale ce qui devrait être le cas lors de l'élaboration des correcteur. Comme mentionné avant, durant le vol des drones plusieurs incertitudes peuvent surgir ce qui est du au changement paramétrique. Ces incertitudes peuvent être internes et/ou externes, structurelles ou non-structurelles, ce qui produit certaines déviations de comportement nominal de celui perturbé. La tâche majeure de la commande robuste est de permettre la commande de n'importe quel modèle perturbé par le correcteur calculé a priori pour le modèle nominal. Dans ce travail, trois modèles linéaires de drone sont calculés avec variation de certains paramètres ; comme montré dans le "chapitre 1" ; sont pris pour montrer l'efficacité de cette méthode.

Le modèle non linéaire de drone est linéarisé en trois différents points de fonctionnement pour former le modèle nominal, représenté par le quadruple matriciel suivant  $[A, B, C, D, G]$  et les modèles perturbés données par les deux quadruples  $[A_{p1}, B_{p1}, C_{p1}, D_{p1}, G_{p1}]$  et  $[A_{p2}, B_{p2}, C_{p2}, D_{p2}, G_{p2}]$ . Le problème est d'assurer que le correcteur prédéterminé pour le modèle nominal offre toujours les mêmes performances et la stabilité dans le cas de variation paramétrique. Un critère de performance est formé pour l'estimation des performances et la robustesse des trois modèles basé sur les normes  $H_2/H_\infty$  calculée pour différentes fonctions de sensibilité "S" et la fonction de sensibilité complémentaire du système bloc montré dans la Figure 2.2, avec les poids de pondération de LaGrange qui mesurent le poids de contribution de chaque terme.

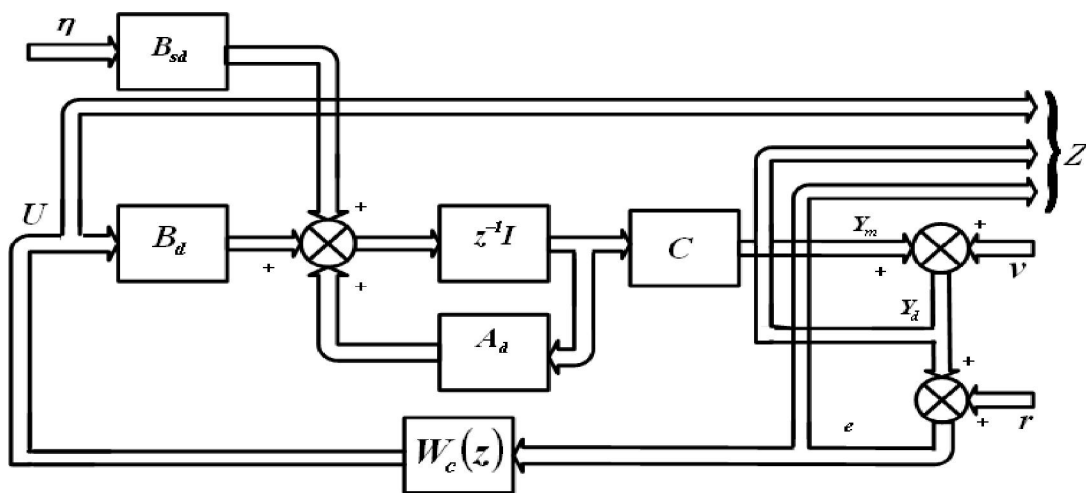


Figure 2.2 : schéma bloc de calcul des fonctions de transferts.

$r(t)$  est les signaux de référence à poursuivre,  $W_c(z)$  est la fonction de transfert discret de correcteur LQG. Dans le diagramme ci-dessous, pour estimer les performances et la robustesse, le vecteur à optimiser est  $Z = [Y_d \quad U \quad e]^T$ , où  $Y_d$  est la sortie de système,  $U$  est l'entrée de correcteur, et  $e$  est l'erreur entre le signal de référence  $r$  et la sortie mesurée  $Y_d$ . Les entrées dans la boucle fermée Figure 2.2 sont les signaux de référence  $r$ , les perturbations  $\eta$  et les bruits de mesure  $v$ .

Les matrices de transfert vues ci-dessous sont calculées pour le modèle nominal, la même chose peut être faite pour les deux autres modèles perturbés en remplaçant dans (2.3) le modèle nominal de drone  $[A, B, C, D, G]$  par  $[A_{p1}, B_{p1}, C_{p1}, D_{p1}, G_{p1}]$  et le modèle de Dryden associé au modèle nominal  $[A_{dr}, B_{dr}, C_{dr}, D_{dr}]$  par  $[A_{drp1}, B_{drp1}, C_{drp1}, D_{drp1}]$  pour le premier modèle perturbé et  $[A_{p2}, B_{p2}, C_{p2}, D_{p2}, G_{p2}]$ ,  $[A_{drp2}, B_{drp2}, C_{drp2}, D_{drp2}]$  pour le second modèle.

Comme mentionné avant les performances sont mesurées en calculant la norme  $H_2$  pour les différentes matrices données dans (2.10), et de même pour les deux autres modèles perturbés. Il est rependu que dans la théorie de la commande robuste que les performances de systèmes en boucle fermée peuvent être estimées en utilisant la norme  $H_\infty$ . Pour avoir de bonnes performances il faut que cette norme soit petite.

La norme  $H_2$  des matrices de transfert données en (2.10), qui constituent la partie déterministe de problème de commande, pour chaque modèle (nominal et perturbés) est calculée en utilisant la représentation de l'espace d'état pour chaque modèle comme suit :

$$S_n = \left[ \begin{array}{c|c} A_{S_n} & B_{S_n} \\ \hline C_{S_n} & D_{S_n} \end{array} \right]; \quad S_{p1} = \left[ \begin{array}{c|c} A_{S_{p1}} & B_{S_{p1}} \\ \hline C_{S_{p1}} & D_{S_{p1}} \end{array} \right]; \quad S_{p2} = \left[ \begin{array}{c|c} A_{S_{p2}} & B_{S_{p2}} \\ \hline C_{S_{p2}} & D_{S_{p2}} \end{array} \right] \quad (2.10)$$

$$T_n = \left[ \begin{array}{c|c} A_{T_n} & B_{T_n} \\ \hline C_{T_n} & D_{T_n} \end{array} \right]; \quad T_{p1} = \left[ \begin{array}{c|c} A_{T_{p1}} & B_{T_{p1}} \\ \hline C_{T_{p1}} & D_{T_{p1}} \end{array} \right]; \quad T_{p2} = \left[ \begin{array}{c|c} A_{T_{p2}} & B_{T_{p2}} \\ \hline C_{T_{p2}} & D_{T_{p2}} \end{array} \right]$$

Et la norme  $H_2$  est donnée :

$$H2_{S_n} = \|S_n\|_2 = \sqrt{\text{trace}(C_{S_n} Q_{S_n}) C_{S_n}^T}; \quad H2_{S_{p1}} = \|S_{p1}\|_2 = \sqrt{\text{trace}(C_{S_{p1}} Q_{S_{p1}}) C_{S_{p1}}^T};$$

$$H2_{S_{p2}} = \|S_{p2}\|_2 = \sqrt{\text{trace}(C_{S_{p2}} Q_{S_{p2}}) C_{S_{p2}}^T}$$

Et

$$H2_{T_n} = \|T\|_2 = \sqrt{\text{trace}(C_{T_n} Q_{T_n}) C_{T_n}^T}; \quad H2_{T_{p1}} = \|T_{p1}\|_2 = \sqrt{\text{trace}(C_{T_{p1}} Q_{T_{p1}}) C_{T_{p1}}^T};$$

$$H2_{T_{p2}} = \|T_{p2}\|_2 = \sqrt{\text{trace}(C_{T_{p2}} Q_{T_{p2}}) C_{T_{p2}}^T}$$

Où  $Q_{S_n}$ ,  $Q_{S_{p1}}$ ,  $Q_{S_{p2}}$ ,  $Q_{T_n}$ ,  $Q_{T_{p1}}$  et  $Q_{T_{p2}}$  sont les gramians de commandabilité des modèles décrits en (2.10), et donnée par la solution des équations de Lyapunov suivantes :

$$A_{S_n} Q_{S_n} A_{S_n} - Q_n + B_{S_n} B_{S_n}^T = 0$$

$$A_{S_{p1}} Q_{S_{p1}} A_{S_{p1}} - Q_{p1} + B_{S_{p1}} B_{S_{p1}}^T = 0$$

$$A_{S_{p2}} Q_{S_{p2}} A_{S_{p2}} - Q_{p2} + B_{S_{p2}} B_{S_{p2}}^T = 0$$

Et

$$A_{T_n} Q_{T_n} A_{T_n} - Q_n + B_{T_n} B_{T_n}^T = 0$$

$$A_{T_{p1}} Q_{T_{p1}} A_{T_{p1}} - Q_{p1} + B_{T_{p1}} B_{T_{p1}}^T = 0$$

$$A_{T_{p2}} Q_{T_{p2}} A_{T_{p2}} - Q_{p2} + B_{T_{p2}} B_{T_{p2}}^T = 0$$

Où le suffixe n désigne nominal, et pi pour le ième modèle perturbé.

La norme  $H_\infty$  estime la robustesse de système et indique le gain maximal en boucle fermée, cependant pour assurer la stabilité et la robustesse du système, il faut que ce gain reste petit. La norme  $H_\infty$  est calculée comme suit:

$$\|H\|_\infty = \sup_w \bar{\sigma}(H(jw)) \quad (2.11)$$

$\bar{\sigma}(H(jw))$  représente le maximum des valeurs singulières des matrices de sensibilité prédéfinies en (2.10), et  $\sup_w \bar{\sigma}(H(jw))$  le maximum global des valeurs singulières de modèle

MIMO ( Multi Input Multi Output) tout en long de la gamme des fréquences  $w=2\pi f$ .

Le maximum des valeurs singulières  $\bar{\sigma}(H(jw))$  doit être minimisé de sorte à assurer au système la robustesse en stabilité et en performance, et surtout éviter la saturation des actionneurs ce qui est primordiale pour les objets volants, pour ne pas atteindre les conditions de décrochage.

Après avoir estimé les performances et la robustesse du système en utilisant la norme  $H_2$  et la norme  $H_\infty$  des différentes matrices de transfert pour les trois modèles (nominal et perturbés), nous pourrions appliquer la procédure d'optimisation robuste  $H_2/H_\infty$  pour chercher le compromis entre robustesse et performances en utilisant les algorithmes génétiques. Il faut noter que les matrices de transfert en boucle fermée (2.10) dépendent de la matrice de gain de filtre de Kalman et celle de correcteur (LQ)  $K$ , donc les normes  $H_2$  et  $H_\infty$  y dépendent elles aussi, ces deux gains constituent les variables d'optimisation qui seront utilisées pour trouver le compromis entre robustesse et performances.

Le critère de performance est formé pour chaque modèle avec les poids de pondérations de LaGrange correspondant à la contribution de chaque terme estimé, le critère de performance pour le modèle nominal est donné comme suit :

$$J^n(L, F) = \lambda_1^n H_{2S_n} + \lambda_2^n H_{\infty S_n} \quad (2.12)$$

Le même critère de performance est attribué pour les deux modèles perturbés, dont les expressions sont :

$$J^{p1}(L, F) = \lambda_1^{p1} H_{2S_{p1}} + \lambda_2^{p1} H_{\infty S_{p1}} \quad (2.13)$$

$$J^{p2}(L, F) = \lambda_1^{p2} H_{2S_{p2}} + \lambda_2^{p2} H_{\infty S_{p2}} \quad (2.14)$$

La fonction de cout totale à optimiser est donnée par la sommation de (2.12), (2.13) et (2.14), exprimée comme suit :

$$J(L, F) = J^n(L, F) + J^{p1}(L, F) + J^{p2}(L, F) \quad (2.15)$$

Tels que les facteurs de LaGrange  $\lambda_i^k$ , ( $k = n, p_1, p_2$ ), pondèrent la contribution des termes estimant performance et robustesse dans la fonction cout, respectivement,  $n$ ,  $p_1$ , et  $p_2$  correspondent aux modèles nominal, premier et deuxième modèle perturbé, respectivement.

Puisque le calcul de la norme  $H_2$  est basé sur la gouvernabilité de gramian des matrices de sensibilité de système en boucle fermée définies dans (2.10), en conséquent ces matrices doivent rester stables et complètement commandables durant la procédure d'optimisation. Cependant, la fonction cout doit inclure un autre terme nommé *fonction de pénalité* (FP), définissant la localisation des pôles de système en boucle fermée dans une région prédéfinie dans le plan

complexe figurant dans la Figure 2.3a. La fonction de pénalité pourrait être représentée graphiquement :

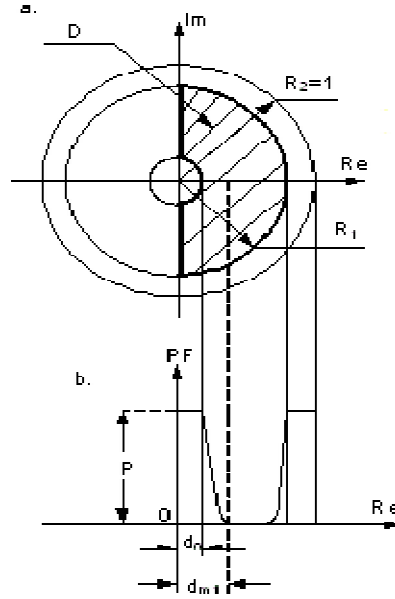


Figure 2.3a : Fonction de pénalité dans le domaine complexe.

$$PFi(d_m) = \begin{cases} 0, & \text{si } d_m \geq d_{ml} \\ \frac{P}{2} \left[ 1 + \cos\left(\frac{\pi(d_m - d_0)}{d_{ml} - d_0}\right) \right] & \text{si } d_0 < d_m < d_{ml} \\ P, & \text{si } d_m \leq d_0 \end{cases} \quad (2.16)$$

Telle que la valeur de P soit très grande (par exemple  $P \in [10^4, 10^6]$ ). Cette fonction est lisse et dérivable tout au long de cercle unitaire. La recherche de la distance minimale de  $d_m$  est une nécessité pour que chaque pôle des modèles nominal et perturbés soient loin des bords de domaine D décrit dans le plan complexe z.

Après avoir augmenté la fonction cout (2.15) de ce terme, la nouvelle fonction objectif à optimiser devient :

$$J_{\Sigma}(L, F) = J(L, F) + PF_i \quad (2.17)$$

### II.3. Algorithmes génétiques comme procédure d'optimisation robuste :

Tel qu'il est mentionné avant, la propriété la plus importante pour les systèmes de commande de vol (pilote automatique) est la robustesse et garantir des bonnes performances.

L'optimisation robuste  $H_2/H_\infty$  devrait satisfaire plusieurs objectifs et trouver des compromis entre eux, pour assurer de bonnes qualités de manœuvre de système commandé, spécialement la commande des drones (Flight Dynamic Control System FDCS) où le système est soumis à des perturbations internes et externes. La fonction coût dans (2.17) contient des termes relatifs à la rejection des perturbations, à la stabilité robuste et à l'influence de la variation paramétrique sur les performances en boucle fermée. De telles fonctions sont nommées *fonctions objectif à multi-objectifs*, et l'usage des méthodes d'optimisation traditionnelles (classiques) va induire aux minimums locaux. Cependant, la méthode des algorithmes génétiques est plus adéquate et la plus adaptée pour de telles fonctions.

Les algorithmes génétiques (AG) ont pour objectif une recherche stochastique de l'optimum global basée sur la théorie Darwinienne "survie du plus adapté". Ils cherchent la solution optimale dans l'espace d'évolution en échangeant des informations entre les individus adaptés aléatoires. Les principes des AGs sont premièrement publiés par Holland. Le modèle mathématique a été développé à la fin des années soixante, et ils sont apparus dans son livre intitulé "Adaptation in Natural and Artificial System" publié en 1975. Les AGs sont développés pour résoudre des problèmes difficiles avec des fonctions qui ne possèdent pas de jolies propriétés telles que continuité, dérivabilité, etc, basés sur la sélection naturelle et la génétique. Ces algorithmes font partir des techniques de recherche des *algorithmes évolutionnaires* (AEs). Ils maintiennent et manipulent une famille, ou une population d'individus et appliquent la stratégie "Survie de l'individu le plus adapté" "Survival of the fittest, en Anglais" durant la recherche d'une meilleure solution. Son déroulement se fait sur toute une population de point contrairement autres qui prennent un seul point, ce qui offre l'effet de parallélisme, permettant ainsi une exploitation de plusieurs régions d'intérêt au même temps. Dans cette section ; une brève formulation des AGs sera présentée. A la base, il y a trois opérateurs pour un simple AG : reproduction, croisement et mutation, qui sont représentés dans l'organigramme sur la Figure 2.5. Ces opérateurs travaillent avec des individus artificiels nommés *génération*. Par échange d'information entre chaque individu de la population, les AGs préservent les individus les plus adaptés d'une génération à une autre pour l'atteinte de l'objectif de minimisation. Cependant, pour la recherche de l'optimum global, une autre méthode est ajoutée dit méthode d'élite (Elite Methode in English) pour enrichir l'efficacité de la recherche.

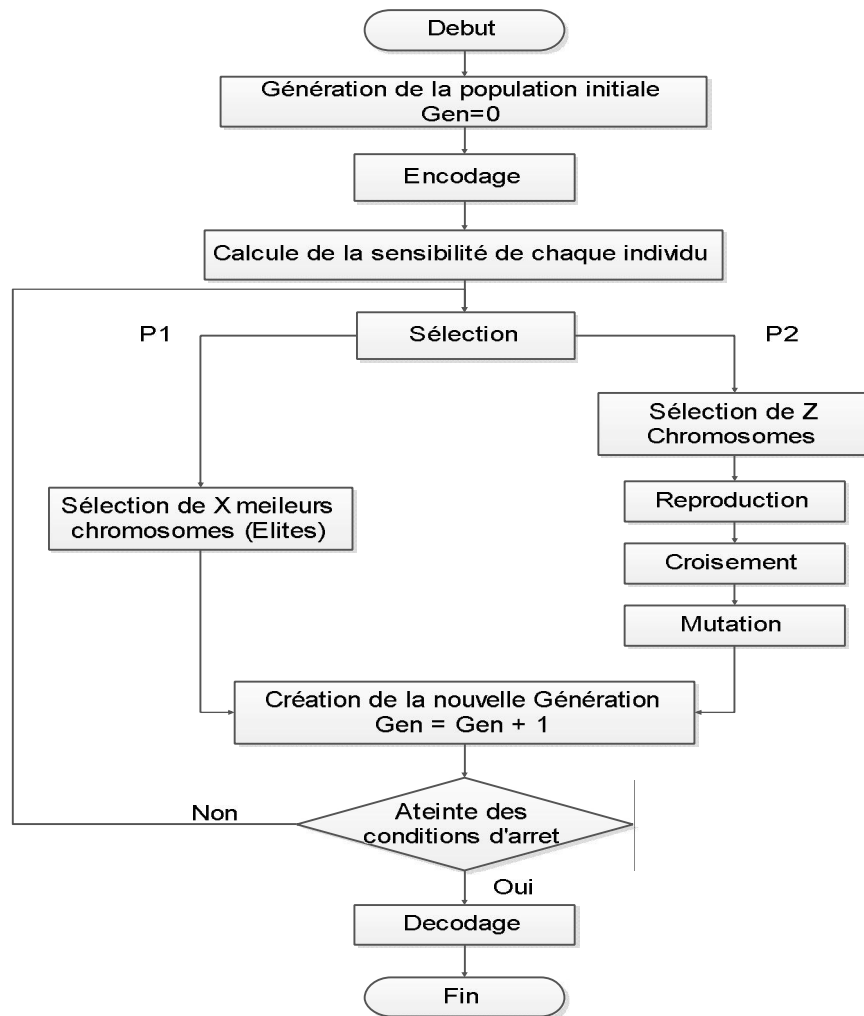


Figure 2.5 : Organigramme des algorithmes génétiques.

### II. 3.1. Opérateurs de base des algorithmes génétiques.

Les individus initiaux, à qui nous appliquons les fonctions d'adaptation sont générées aléatoirement, de plus, la population initiale composée d'un ensemble d'individus est aussi générée. A chaque itération, les AGs font des calculs sur la population actuelle et produisent une nouvelle génération.

Soit  $y$  une fonction à optimiser et à  $N$  variables données par :

$$\begin{cases} y = F(X_1, \dots, X_N) \\ \text{Sujet à} \\ a_i < X_i < b_i \end{cases}$$

Et  $i=1, \dots, N$ .

Dans l'intervalle  $[a_i, b_i]$  une population initiale ; appelée "initial candidate solution" en Anglais ; est créée pour chaque variable d'optimisation. Le nombre d'individu dans cette population  $X_i(j)$  est égale à la taille de la population  $M_i$  ( $j=0, \dots, M_i$ ), donc, si la taille d'une population est 150 et le nombre de variables évoluant durant le procédure de l'optimisation est  $N$ , une matrice de  $150 \times N$  d'individus sera générée d'une manière aléatoire à l'intérieur des intervalles  $[a_i, b_i]$ , dans les algorithmes génétique, la représentation de chaque individu  $X_i(j_0)$  de la population est appelée chromosome, cette étape est l'encodage et différents alphabets ont été proposés, la représentation binaire est la première méthode proposée par Holland.

Dans cette méthode un vecteur ou registre est généré et sa longueur dépend de la précision requise. L'intervalle  $[a_i, b_i]$  est divisé au moins  $2^{N_b-1} < (b_i - a_i) \cdot 1000 \dots 000 \dots (N_b - 1/b_i - a_i) < 2^{N_b}$ , où  $N_b$  et la longueur de registre en binaire. Diverses méthodes de représentation ont été proposées en AG, et il est prouvé que la meilleure représentation de chromosome est sa valeur réelle choisie entre les deux limites de l'intervalle choisi.

Après avoir appliqué les opérateurs au chromosome codé, et pour avoir la valeur représentée par ce dernier les deux étapes sont utilisées :

- 1- Convertir le registre binaire  $\langle b_{N_b-1} \ b_{N_b-2} \ \dots \ b_0 \rangle$  de la base deux 02 vers la base dix 10.

$$\langle b_{N_b-1} \ b_{N_b-2} \ \dots \ b_0 \rangle_2 = \left( \sum_{i=0}^{N_b-1} b_i \cdot 2^i \right)_{10} = X'_i(j)$$

- 2- La valeur réelle est obtenue comme suit :

$$X_i(j) = a_i + X'_i(j) \frac{b_i - a_i}{2^{N_b} - a_i}$$

Pour évaluer la fonction donnée à (2.17), la représentation binaire de chromosome doit être convertie vers une valeur réelle,  $y_j = F(X_1(j), \dots, X_N(j))$  avec  $j=0, \dots, M_i$ .

Dans ce qui suit nous présenterons les opérations de base en algorithmes génétiques.

- 1- Sélection :** dans ce processus le chromosome est sectionné d'une population selon sa valeur d'adaptation. Grace à cette opération les chromosomes dont les valeurs plus appropriées auront plus de chance à être reproduits dans la génération suivante. Diverses méthodes de sélection ont été proposées, et certaines d'entre elles sont données dans le paragraphe suivant.

- **Sélection de la roulette** est la première méthode proposée par Holland, et elle est définie par :

$$P_i = \frac{F_i}{\sum_{j=1}^N F_j}$$

Où  $P_i$  est la probabilité que l'individu  $i$  soit sélectionné,  $F_i$  est la l'adaptation de l'individu  $i$  et  $N$  est le nombre d'individus dans une population. Dans cette méthode l'individu possédant la valeur d'adaptation la plus élevée a la chance la plus élevée d'être sélection et plus qu'une seule fois. En conséquent, ce n'est pas sophistiquée car nous réduisons la diversité dans la population puisque l'individu dont l'adaptation est faible a moins de chance d'être sélectionné.

- **Sélection normalisée de Rang géométrique (Normalized Geometric Rank Selection)** définit  $P_i$  pour chaque individu par :

$$P_i = q^l(1 - q)^{r-1}$$

Où:

- $q$  est la probabilité de sélectionner le meilleur individu ;
- $r$  est le rang de l'individu, avec  $l$  est le meilleur ;
- $P_i$  est la probabilité de l'individu  $i$  d'être sélectionné ;
- $q^l = \frac{q}{1 - (1 - q)^N}$  où  $N$  la taille de la population.

- **Sélection par tournoi (Tournament selection)** : elle a seulement besoin de la fonction d'évaluation pour situer les solutions pour un système partiellement ordonné. Cette méthode ne nécessite pas les probabilités; elle travaille en sélectionnant  $j$  individus au hasard, les plaçant dans la population et inserant le meilleur  $j$  dans la nouvelle génération. Cette procédure est répétée jusqu'à ce-que les  $N$  individus seront sélectionnés.

Après avoir sélectionné  $N$  individus par les méthodes mentionnées ci-dessous ou par d'autres, les opérateurs génétiques sont appliqués pour générer la nouvelle solution basée sur la solution existante dans la population. Les deux opérateurs de base sont le croisement et la mutation.

- 2- **Croisement** : prend deux individus parmi ceux sélectionné et crée deux nouveaux dans la nouvelle sélection. L'utilisation de cet opérateur dépend de l'encodage utilisé dans les AGs,

si la chaîne binaire est utilisée ; les opérateurs suivants sont définis pour le croisement en un point, en deux points, en multi points, croisement par remaniement. Les méthodes de croisement utilisées pour l'évaluation réelle des chaînes de caractères (chromosomes) sont croisement intermédiaire, ligne de croisement, ligne étendue de croisement et le croisement heuristique. D'autres méthodes existent tels que le croisement arithmétique ou encore discret. Quels que méthodes de croisement sont définies par :

- **Croisement en un seul point** : dans cette méthode le croisement en un point est choisi pour les deux parents aléatoirement. Toutes les données au dessus de ce point sont échangées entre eux. Le schéma suivant illustre la procédure :

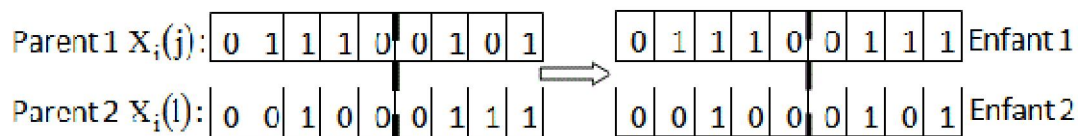


Figure 2.6 : Croisement en un point

Pour le croisement double deux points sont choisis aléatoirement et les informations entre eux seront échangées. Même procédure se fait pour le croisement en multipoints, sur les  $m$  points sélectionnés, les bits entre chaque deux points successifs sont échangés entre les deux parents pour former les enfants.

#### . Croisement à valeur réelle (Real valued crossover) :

Les méthodes de croisement suivantes travaillent seulement avec les valeurs réelles d'encodage de chromosome. L'un d'entre eux est décrit ci-après :

. **Croisement intermédiaire** : Cette méthode utilise une combinaison linéaire de tous les parents pour produire les nouveaux enfants en usant cette expression :

$$x'_i = x_{S,i} + \alpha_i (x_{T,i} - x_{S,i})$$

Avec les indices S et T désignent les individus parentaux sélectionnés au hasard par l'une des méthodes,  $\alpha \in [0, 1]$  est une variable aléatoire uniforme et  $x'_i$  est le nouveaux individu induit pour la nouvelle génération.

- 3- **Mutation** : La mutation consiste à changer les informations présentes au niveau des individus. Si ces derniers sont représentés par des vecteurs, la mutation est le changement

aléatoire d'un élément d'un vecteur, la façon dont cette mutation intervient dépend selon le type des éléments de vecteur, comme le cas de mécanisme de croisement. Si le vecteur est une chaîne de bits, la mutation alterne le bit ou pas (avec la même probabilité). Pour la représentation de la valeur réelle d'autres opérateurs de mutation plus sophistiqués sont nécessaires. Certaines de ces méthodes sont définies comme suit :

- a. **Mutation binaire** bascule chaque bit dans chaque individu de la population avec une probabilité  $p_m$ , en s'appuyant sur l'équation (2.44) :

$$x'_i = \begin{cases} 1 - x_i & \text{if } \alpha_i < p_m \\ x_i & \text{Sinon} \end{cases} \quad (2.44)$$

Tel que  $p_m$  est définie comme étant très petite ( $p_m \approx 10^{-3}$  par bit) et  $\alpha_i \in [0, 1]$  est une variable aléatoire, choisi pour chaque bit.

- b. **Représentation par des valeurs réelle** : pour un réel  $\bar{X}$  les opérateurs de mutation sont définis. Soient  $a_i$  et  $b_i$  la limite inférieure et supérieure, pour chaque variable  $i$ .

Une mutation uniforme sélectionne aléatoirement une variable  $j$  et la rend un nombre aléatoire uniforme entre les deux limites inférieure et supérieure  $U(a_i, b_i)$  :

$$x'_i = \begin{cases} y_i & \text{if } i = j \\ x_i & \text{Sinon} \end{cases}$$

Et  $y_i \in [a_i, b_i]$ .

- c. **Mutation tout ou rien (boundary mutation)** : Sélectionne une variable  $j$  aléatoirement et lui attribue une valeur limite inférieure ou supérieure.

$$x'_i = \begin{cases} a_i & \text{if } i = j, \alpha < 0.5 \\ b_i & \text{if } i = j, \alpha \geq 0.5 \\ x_i & \text{Sinon.....} \end{cases}$$

Avec  $\alpha \in [0, 1]$  est une variable aléatoire uniforme.

- d. **Mutation non-uniforme** : Sélectionne une variable  $j$  aléatoirement et lui attribue une des valeurs aléatoires non uniformes :

$$x'_i = \begin{cases} x_i + (b_i - x_i)f(G) & \text{if } \alpha < 0.5 \\ x_i + (x_i + a_i)f(G) & \text{if } \alpha \geq 0.5 \\ x_i & \text{Sinon.....} \end{cases}$$

Avec :

$$f(G) = \left( \alpha_2 \left( 1 - \frac{G}{G_{\max}} \right) \right)^b ;$$

- $(\alpha_1, \alpha_2) \in [0, 1]$  sont des nombres uniformes ;
- $G$  est la génération actuelle ;
- $G_{\max}$  est le nombre maximal de générations ;
- $b$  est un paramètre de forme (shape parameter).

### II.3.2. L'élitisme

Les opérations citées ci-dessous des algorithmes génétiques sont utilisées pour altérer et étendre le domaine de recherche de la solution de problème ; mais durant le processus de calcul il se peut que le meilleur candidat puisse être perdu. Pour cette raison, un modèle d'élite est créé pour y remédier. Dans cette méthode, le meilleur ou les meilleurs chromosomes sont copiés dans la nouvelle génération. Et pour le reste, tout se fait de la façon classique. Donc des individus pourraient se perdre s'ils ne sont pas sélectionnés pour se reproduire, ou si le croisement ou la mutation les détruisent. L'élitisme améliore les performances des algorithmes génétiques.

### II.3.3. Critère de convergence :

Les critères d'arrêt des algorithmes génétiques sont brièvement cités ci-après :

- **Atteinte de nombre de générations :** Les algorithmes génétiques s'arrêtent si le nombre de générations est atteint.
- **Achèvement de temps de calcul :** le processus de calcul des AGs s'arrête si le temps spécifié est écoulé.
- **Inadaptation :** le déroulement des algorithmes génétiques s'achève s'il n'y a pas de changement des paramètres d'adaptation tout au long de nombre de générations.
- **Figement des générations :** s'il n'y a pas d'amélioration de la fonction objectif d'une génération à d'autres l'algorithme s'arrête.

- **Figage de temps limite** : l'algorithme s'arrête s'il n'y a pas d'amélioration de la fonction objectif sur un intervalle de temps.

**Conclusion:**

Ce chapitre s'est porté en premier lieu sur la commande optimale associée à un observateur, ce qui a donné LQG, qui a permis de trouver un correcteur optimal, mais ce dernier n'est pas aussi efficace en présence des perturbations et des bruits, donc il a fallu le robustifier en minimisant un critère de performance et en maintenant ; à tout instant ; le système stable. La résolution d'un tel problème d'optimisation nécessite une méthode d'optimisation globale telle que les algorithmes génétiques.

# CHAPITRE III

**CHAPITRE III****RESULTATS DE SIMULATION**

La détermination des valeurs initiales des paramètres du correcteur optimal est faite par la méthode LQG ( Linear Quadratic Gaussian ), la réponse du système ( drone) est aussi optimale présentant certaines performances assez intéressantes, et vu son incapacité de remplir les mêmes performances sur un intervalle de fonctionnement, nous proposons de robustifier ce correcteur en imposant certains critères de performance dans une fonction objectif (f.b), de sorte qu'elle soit la somme des normes  $H_2$  de la fonction de sensibilité "S" représentant le critère de performance, et la somme des normes infinies  $H_\infty$  de la fonction de sensibilité complémentaire "T" représentant la robustesse de la commande, sans oublier le critère de stabilité ce qui est plus important.

La procédure de d'optimisation de la fonction objectif est faite en utilisant Algorithmes Génétiques, après avoir essayé plusieurs méthodes de croisement et de mutation il est en résulté que les critères qui donnent de bon résultats sont :

- Méthode de croisement : croisement intermédiaire ;
- Méthode de mutation : mutation non-uniforme ;
- Nombre d'élite : 2 pour l'optimisation du correcteurs généré pour le mouvement longitudinal et 5 pour latéral.
- La taille de la population : 250.
- Nombre de génération : 20.

Après trente itérations nous obtenons les normes et les résultats suivants:

### 3.1. Valeurs des normes :

#### 3.1.1. Longitudinales :

H <sub>2</sub> LONGITUDINALES						
Iteration	H2_S_n	H2_S_p1	H2_S_p2	H2_T_n	H2_T_p1	H2_T_p2
0	8.1436	1.5112	8.6091	8.1436	1.5112	8.6091
1	4.7130	0.9851	4.8948	4.7130	0.9851	4.8948
2	4.8618	0.9630	5.0732	4.8618	0.9630	5.0732
3	5.1603	1.0921	5.3554	5.1603	1.0921	5.3554
4	4.3899	0.8856	4.5728	4.3899	0.8856	4.5728
5	4.5681	1.0029	4.7732	4.5681	1.0029	4.7732
6	5.2846	1.0871	5.5426	5.2846	1.0871	5.5426
7	5.1692	1.0686	5.3370	5.1692	1.0686	5.3370
8	4.3889	0.9285	4.5588	4.3889	0.9285	4.5588
9	4.0644	0.8712	4.2223	4.0644	0.8712	4.2223
10	5.5665	1.0853	5.7602	5.5665	1.0853	5.7602
11	4.8364	1.0608	4.9754	4.8364	1.0608	4.9754
12	4.4278	0.9810	4.5979	4.4278	0.9810	4.5979
13	4.2762	0.9505	4.4210	4.2762	0.9505	4.4210
14	4.3457	0.9468	4.5384	4.3457	0.9468	4.5384
15	4.2967	0.9240	4.4441	4.2967	0.9240	4.4441
16	4.1256	0.8697	4.2358	4.1256	0.8697	4.2358
17	5.3205	1.0531	5.6051	5.3205	1.0531	5.6051
18	4.1412	0.8803	4.3088	4.1412	0.8803	4.3088
19	4.7113	0.9965	4.8867	4.7113	0.9965	4.8867
20	5.1580	1.0333	5.3510	5.1580	1.0333	5.3510
21	4.4511	0.9313	4.6302	4.4511	0.9313	4.6302
22	5.4594	1.1215	5.7195	5.4594	1.1215	5.7195
23	4.9832	1.0315	5.1682	4.9832	1.0315	5.1682
24	3.6552	0.8147	3.8003	3.6552	0.8147	3.8003
25	5.1181	1.1341	5.3307	5.1181	1.1341	5.3307
26	4.3100	0.9634	4.4563	4.3100	0.9634	4.4563
27	3.9745	0.9283	4.1181	3.9745	0.9283	4.1181
28	3.9852	0.9445	4.1103	3.9852	0.9445	4.1103
29	4.4808	0.9094	4.6942	4.4808	0.9094	4.6942
30	5.4876	1.1109	5.6977	5.4876	1.1109	5.6977

Tableau 3.1: Valeurs des normes H<sub>2</sub> des deux fonctions de sensibilité S et T sur 30 itérations pour les modèles longitudinaux.

H <sub>∞</sub> LONGITUDINALES						
Iteration	Hinf_S_n	Hinf_S_p1	Hinf_S_p2	Hinf_T_n	Hinf_T_p1	Hinf_T_p2
0	2.0180	1.7890	2.1958	2.0472	1.7964	2.0239
1	1.6025	1.4412	1.8084	1.5750	1.3701	1.5320
2	1.6452	1.5248	1.8479	1.6850	1.4790	1.6271
3	1.7942	1.4959	1.9013	1.8305	1.4248	1.7659
4	1.5294	1.4437	1.7258	1.5134	1.4006	1.4794

5	2.0264	1.8769	2.2105	2.0336	1.8871	2.0842
6	2.0049	1.7341	2.2294	1.9778	1.6632	2.0314
7	1.6157	1.5916	1.8088	1.6225	1.6287	1.5902
8	1.6936	1.5014	1.9435	1.6482	1.4243	1.6422
9	1.5980	1.5860	1.9154	1.6039	1.6148	1.5679
10	1.7603	1.5925	2.0772	1.6762	1.5578	1.6997
11	1.5299	1.5290	1.7307	1.5142	1.5851	1.4598
12	1.7836	1.5270	1.8832	1.8521	1.4967	1.7796
13	1.7596	1.7172	1.8958	1.8119	1.7448	1.7801
14	1.6266	1.5244	1.8807	1.5272	1.5445	1.5635
15	1.6410	1.5325	1.8282	1.6078	1.5049	1.6176
16	1.6465	1.5579	1.9132	1.4847	1.5420	1.5576
17	1.7424	1.5255	1.8983	1.8215	1.4811	1.7454
18	1.5318	1.4259	1.7007	1.5641	1.4130	1.5190
19	1.7220	1.5605	1.9378	1.7964	1.5331	1.6947
20	1.6366	1.6869	1.8675	1.7246	1.7371	1.6533
21	1.5486	1.3868	1.6675	1.5747	1.3764	1.5236
22	1.7519	1.4755	1.8622	1.7863	1.4170	1.7228
23	1.6481	1.5156	1.7944	1.7590	1.5329	1.6532
24	1.6132	1.5799	1.8481	1.5886	1.6027	1.5967
25	1.7460	1.6998	2.1256	1.6664	1.7759	1.7129
26	1.6873	1.5546	1.8423	1.7245	1.5460	1.6765
27	1.7952	1.6521	1.9982	1.7824	1.6393	1.7931
28	1.7118	1.5202	1.9024	1.7139	1.4896	1.6855
29	1.7161	1.5797	2.1554	1.6542	1.6015	1.6367
30	1.7658	1.6692	2.0239	1.6972	1.6501	1.7375

Tableau 3.2: Valeurs des normes  $H_\infty$  des deux fonctions de sensibilité S et T sur 30 itérations pour les modèles longitudinaux.

### 3.1.2 . Latérales :

H <sub>2</sub> LATÉRALE						
Iteration	H2_S_n	H2_S_p1	H2_S_p2	H2_T_n	H2_T_p1	H2_T_p2
0	0.7770	0.7289	0.8317	0.7770	0.7289	0.8317
1	0.6297	0.5779	0.6842	0.6297	0.5779	0.6842
2	0.6348	0.5835	0.6882	0.6348	0.5835	0.6882
3	0.5590	0.5174	0.6040	0.5590	0.5174	0.6040
4	0.6805	0.6381	0.7285	0.6805	0.6381	0.7285
5	0.6738	0.6249	0.7273	0.6738	0.6249	0.7273
6	0.7552	0.6971	0.8175	0.7552	0.6971	0.8175
7	0.6761	0.6278	0.7283	0.6761	0.6278	0.7283
8	0.7525	0.6925	0.8152	0.7525	0.6925	0.8152
9	0.5708	0.5315	0.6140	0.5708	0.5315	0.6140
10	0.5958	0.5428	0.6528	0.5958	0.5428	0.6528
11	0.7558	0.7101	0.8055	0.7558	0.7101	0.8055
12	0.5735	0.5328	0.6169	0.5735	0.5328	0.6169
13	0.5072	0.4656	0.5533	0.5072	0.4656	0.5533

14	0.7639	0.7115	0.8203	0.7639	0.7115	0.8203
15	0.7349	0.6804	0.7925	0.7349	0.6804	0.7925
16	0.7185	0.6621	0.7788	0.7185	0.6621	0.7788
17	0.8247	0.7642	0.8873	0.8247	0.7642	0.8873
18	0.6964	0.6391	0.7575	0.6964	0.6391	0.7575
19	0.7563	0.6961	0.8197	0.7563	0.6961	0.8197
20	0.6735	0.6163	0.7336	0.6735	0.6163	0.7336
21	0.6773	0.6195	0.7383	0.6773	0.6195	0.7383
22	0.7030	0.6495	0.7607	0.7030	0.6495	0.7607
23	0.6300	0.5904	0.6734	0.6300	0.5904	0.6734
24	0.6655	0.6159	0.7202	0.6655	0.6159	0.7202
25	0.6564	0.6047	0.7138	0.6564	0.6047	0.7138
26	0.6648	0.6110	0.7208	0.6648	0.6110	0.7208
27	0.6737	0.6325	0.7213	0.6737	0.6325	0.7213
28	0.6930	0.6454	0.7454	0.6930	0.6454	0.7454
29	0.7479	0.6856	0.8124	0.7479	0.6856	0.8124
30	0.7241	0.6700	0.7820	0.7241	0.6700	0.7820

Tableau 3.3: Valeurs des normes  $H_2$  des deux fonctions de sensibilité S et T sur 30 itérations pour les modèles latéraux.

H <sub>∞</sub> LATERALES						
Iteration	Hinf_S_n	Hinf_S_p1	Hinf_S_p2	Hinf_T_n	Hinf_T_p1	Hinf_T_p2
0	3.6336	3.7507	3.6794	3.6318	3.8238	3.6137
1	2.6090	2.5398	2.7415	2.5004	2.5875	2.5138
2	2.6582	2.5979	2.7666	2.4827	2.5978	2.5522
3	2.6466	2.7128	2.7032	2.5904	2.8114	2.5729
4	3.2861	3.4749	3.2696	3.3626	3.6178	3.2704
5	2.9348	2.9304	3.0492	2.8250	2.9663	2.8925
6	2.8725	2.7838	3.0245	2.7149	2.8081	2.8235
7	2.6611	2.6536	2.7449	2.5717	2.7400	2.5748
8	2.8217	2.8324	2.9157	2.8421	2.9690	2.8214
9	2.3553	2.4046	2.4031	2.3040	2.5439	2.2712
10	2.5921	2.4089	2.8040	2.2801	2.3301	2.4996
11	3.0402	3.1063	3.0571	2.9825	3.1102	2.9684
12	2.4983	2.5236	2.5504	2.4316	2.6207	2.4138
13	2.3410	2.2644	2.4782	2.1654	2.4279	2.2028
14	2.9343	2.9681	2.9948	2.8879	3.0908	2.8904
15	2.7915	2.7115	2.8961	2.5853	2.5901	2.6879
16	2.7307	2.6206	2.8888	2.5464	2.5595	2.6556
17	2.8448	2.8281	2.9199	2.7617	2.8285	2.8003
18	2.7953	2.6877	2.9501	2.5787	2.6136	2.7080
19	2.8613	2.7656	3.0074	2.6936	2.8352	2.7818
20	2.7401	2.5831	2.9138	2.4976	2.4383	2.6797
21	2.9353	2.7722	3.1315	2.7381	2.6477	2.9288
22	2.5874	2.5180	2.6986	2.4022	2.5382	2.4688
23	2.5346	2.6598	2.5439	2.5535	2.8046	2.5067
24	2.8804	2.8743	2.9956	2.7609	2.9699	2.8076

25	2.7298	2.6195	2.8950	2.5997	2.8907	2.6802
26	2.9088	2.8362	3.0418	2.7553	2.7862	2.8582
27	3.0636	3.1279	3.1159	3.0601	3.2384	3.0560
28	2.8202	2.9150	2.8770	2.8308	3.0957	2.7797
29	2.7281	2.6076	2.8849	2.5439	2.6044	2.6644
30	2.8873	2.8349	3.0006	2.7618	2.8065	2.8346

Tableau 3.4: Valeurs des normes  $H_\infty$  des deux fonctions de sensibilité S et T sur 30 itérations pour les modèles latéraux.

La minimisation de critère de performance réduit les normes  $H_2$  représentant les performances, et les normes  $H_\infty$ , mais avec des pas de réduction différents dépendants des poids de pondération.

Il est à noter que les résultats de simulation du mouvement longitudinal sont donnés en utilisant les paramètres obtenus en sixième itération et pour le mouvement latéral la huitième itération. Les matrices du gain du filtre de Kalman et le gain du régulateur avant et après la procédure d'optimisation sont données comme suit :

- **Mouvement longitudinal :**

**Gain du filtre de Kalman :**

AVANT L'OPTIMISATION			APRES L'OPTIMISATION		
u	q	h	u	q	h
0.0156	0.0026	-0.0026	0.0108	0.0034	-0.0021
-0.0211	-0.0113	-0.0084	-0.0148	-0.0071	-0.0105
0.0043	0.0035	0.0011	0.0045	0.0043	0.0008
-0.0007	-0.0004	0.0003	-0.0005	-0.0004	0.0002
-0.0033	0.0009	0.0265	-0.0044	0.0006	0.0264
0.1440	0.0178	-0.0212	0.1280	0.0216	-0.0258
-0.0146	0.0018	0.0135	-0.0112	0.0025	0.0105
0.0268	0.0090	0.0265	0.0177	0.0061	0.0227
0.0137	0.0080	0.0028	0.0105	0.0087	0.0018
0.0026	0.0015	0.0011	0.0021	0.0014	0.0011
0	0	0	0	0	0
0	0	0	0	0	0

**Gain du retour d'état :**

<b>AVANT L'OPTI</b>	4.6530	1.4284	-3.1976	-38.4636	-1.0919	0.0059	-0.1294 ...
	0.0739	1.2217	1.3417	32.8509	2.2245		
<b>APRES L'OPT</b>	4.5512	0.2364	-0.1536	-7.7117	0.3096	0.0247	-0.3280...
	0.0871	0.5789	0.4309	0.5169	19.6969		

3.5140	0.2889	-0.1833	-7.5929	0.2520	0.0213	-0.3075...
0.0673	0.4978	0.3426	0.6298	22.6863		

- **Mouvement Latéral :**
- **Gain du filtre de Kalman :**

AVANT L'OPTIMISATION				APRES L'OPTIMISATION			
p	r	Phi	Psi	p	r	Phi	Psi
0.0040	0.0126	0.0048	0.0182	0.0038	0.0099	0.0064	0.0099
0.0074	0.0102	0.0040	0.0077	0.0090	0.0085	0.0026	0.0095
0.0017	0.0058	0.0021	0.0089	0.0026	0.0073	0.0013	0.0064
0.0062	0.0189	0.0071	0.0291	0.0072	0.0269	0.0104	0.0326
0.0007	0.0046	0.0017	0.0162	0.0008	0.0069	0.0016	0.0239
0.0000	-0.0000	0.0000	-0.0000	0.0000	-0.0001	0.0000	-0.0000
0.0104	0.0305	0.0114	0.0438	0.0083	0.0451	0.0090	0.0591
0.0117	0.0154	0.0060	0.0102	0.0134	0.0148	0.0056	0.0140
0.0007	0.0009	0.0004	0.0007	0.0003	0.0007	0.0005	0.0004
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

- **Gain du retour d'état :**

<b>AVANT L'OPT</b>	0.1663	-1.1000	0.0173	-2.6561	-2.5473	-0.1846	-0.0437 ...
	-1.2881	-0.2871	41.7222	0			
	0	0	0	0.0000	0	-0.0000	0.0000 0 ... 0
	0.0603						
<b>APRES L'OPT</b>	0.1258	-1.1924	0.0207	-3.0307	-2.0238	-0.1644	-0.0353 -
	1.2440	-0.2651	46.2356	0			
	0	0	0	0.0000	0	-0.0000	0.0000 0...
	0	0.0754					

### 3.2. Résultats de simulation et interprétation :

Pour le mouvement longitudinal, les signaux de référence sont l'altitude h=50 mètres pendant 100 secondes, après nous réduit l'altitude à 25 mètres pour vérifier les performances du régulateur ; les résultats de simulation sans et avec les perturbations (bruit de fond et externe) sont D donnés sur les figures suivantes :

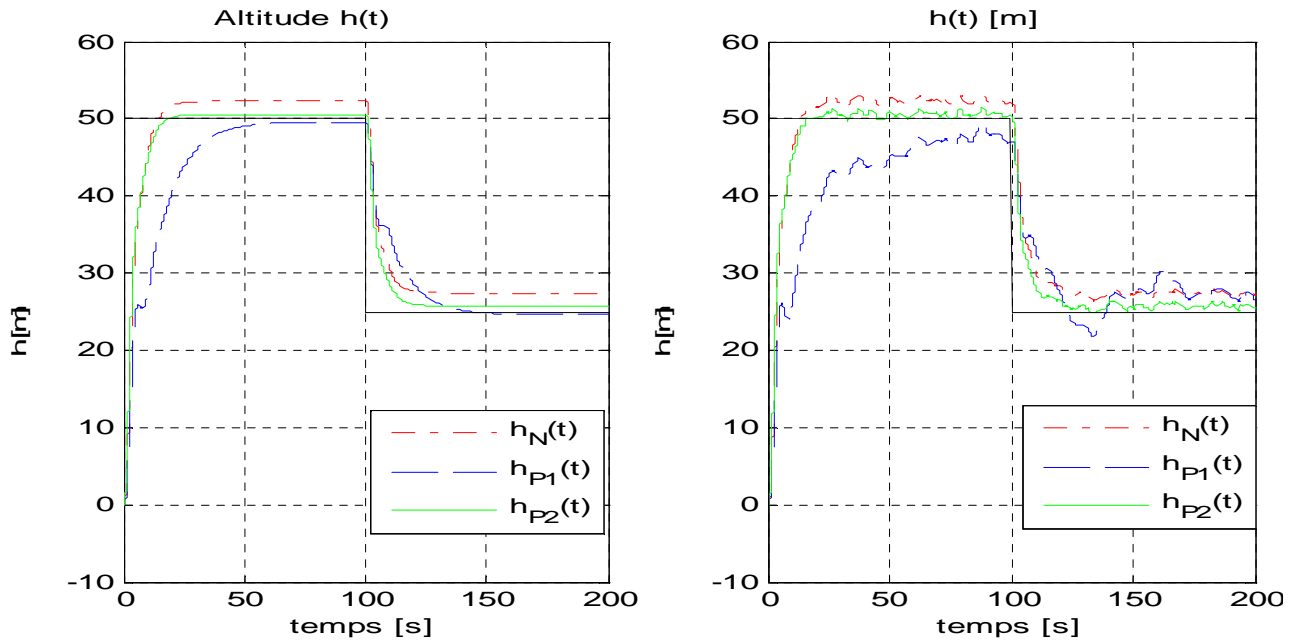


Figure 3.1 : Altitude en fonction de temps

La réponse de drone converge vers la consigne après un temps de réponse spécifique au modèle choisi, mais ce temps diffère d'un modèle à un autre vu que le correcteur est premièrement construit pour le modèle nominal, ce qui explique la différence entre celui de modèle nominal et ceux perturbés ( avec variation parametrique ).

L'introduction des perturbations atmosperiques et les bruits des capteurs, fait perdre certaines performances telle que la rapidité, precision ...etc, mais le correcteur fait toujours en sorte que le drone soit stable et maintienne les sorties ( réponses ) prêt des consignes.

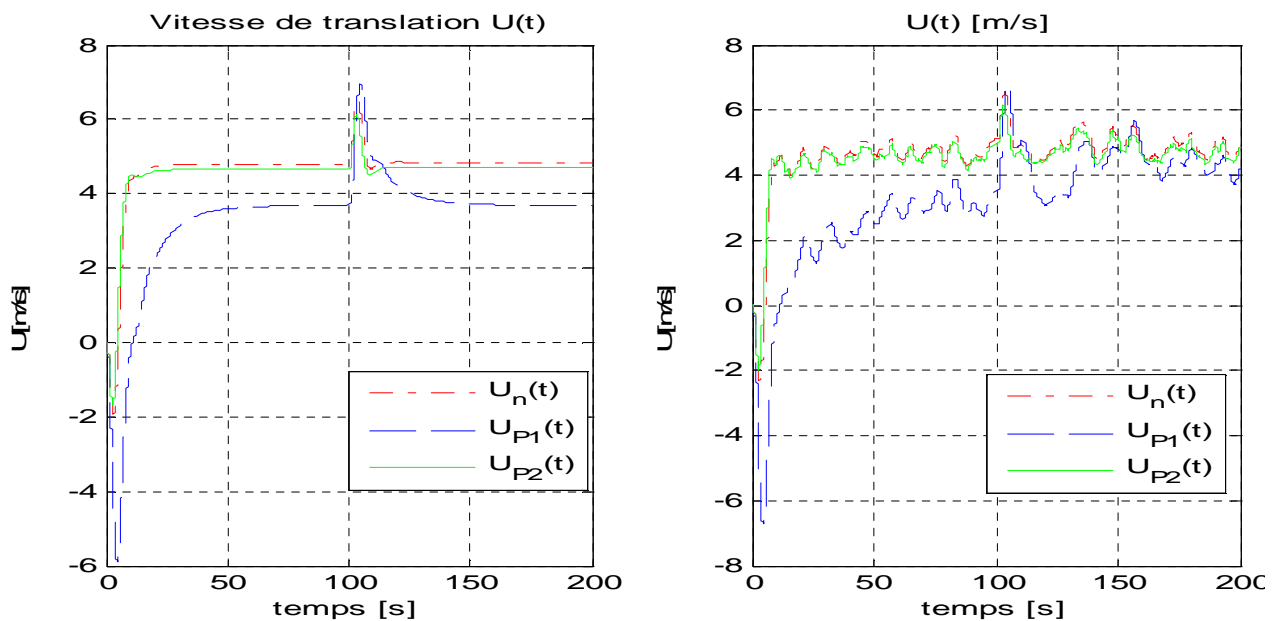


Figure 3.2 : Vitesse de translation en avant en fonction de temps

Il est à observer que pendant la suite de la consigne (altitude) ; la vitesse du drone déminue. Ce fait est dû à la contrubution de la commande en altitude, car l'augmentation de théta (angle de tangage) génère une fornce contreingnante à la translation et cela est du aux frottements génénrés par le contact de l'air sur le bas du fuselage du drone.

La réponse de premier modèle perturbé est moins precise ce qui est du à l'énorme variation de parametres et les erreurs de linearisation, contrimement à celle de deuxième modèle, mais à partir d'un certain instant les trois reponses convergent.

L'évolution de la réponse des modèles et son oscillation autour des réfférences, avec une perturbation de  $\pm 2 m$  et en presence des bruits de fond des capteurs avec une erreur de  $\pm 1.5 m$  montre la robustesse de correcteur choisi.

Les pics apparus à 100 secondes ont pour origine le changement de consigne de l'altitude, en diminuant cette derniere le drone fait une descente vers la consigne souhaitée et gagne de la vitesse où même temps, parallelemnt le correcteur corrige la vitesse.

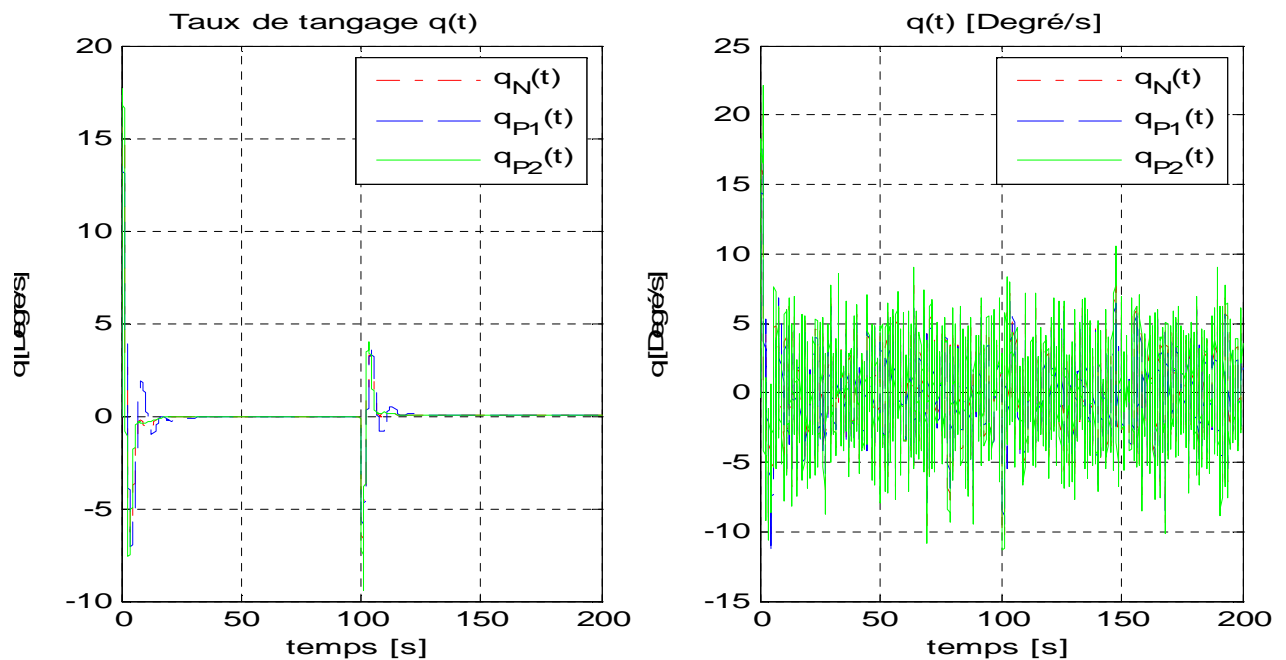


Figure 3.3 : Taux de variation deTangage en fonction de temps.

Le  $q$  (taux de variation de tangage ) augmente en fonction de la variation de l'amplitude qui dépende de l'angle de tangage  $\theta$  (tetha), et à fur et à mesure que la réponse en amplitude s'approche de la consigne le  $q$  se réduit jusqu'à son annulation. Mais dans la pratique les perturbations et les bruits mènent le drone à pencher de nez vers le haut ou le bas de sorte qu'il aparait des fluctuations (oscillation) autour de l'axe de tangage et la tâche du correcteur et de maintenir le  $q$  autour de zero.

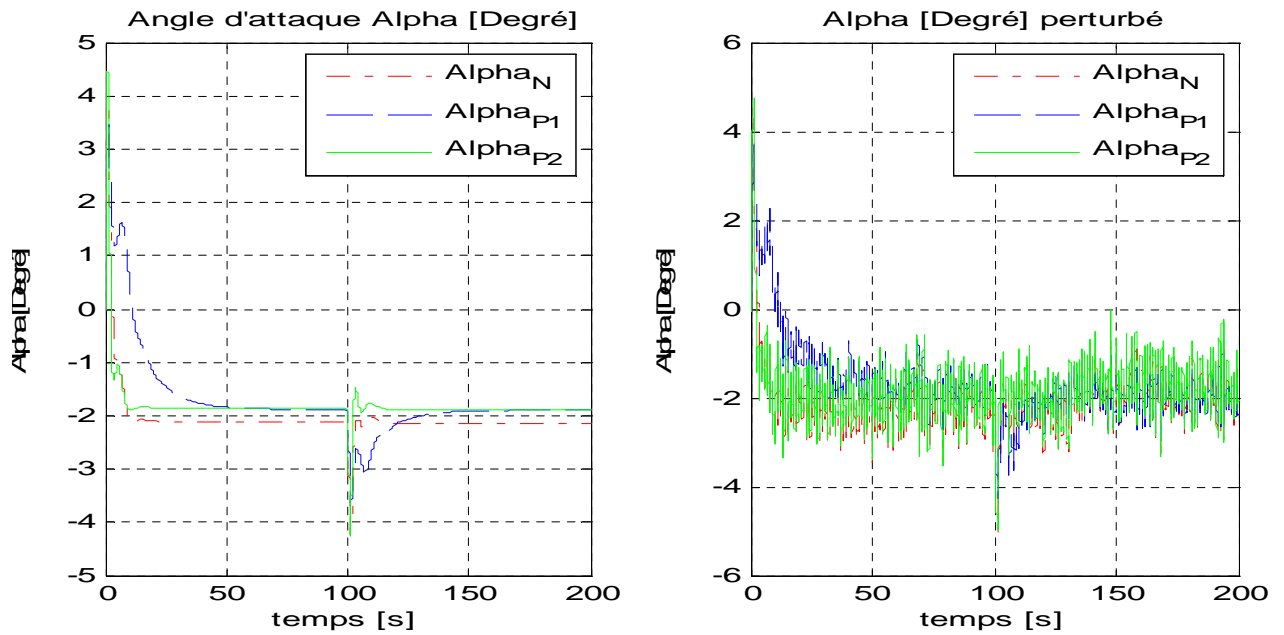


Figure 3.4 : Angle d'attaque en fonction de temps.

$\alpha$  (angle d'attaque) se maintient toujours entre -5 et 5 degrés pour éviter d'atteindre le seuil de décrochage, sa valeur finale ( -2 degrés ) est du maintien de l'amplitude durant le régime permanent ( génération d'une portance suffisente pour le maintien du vol).

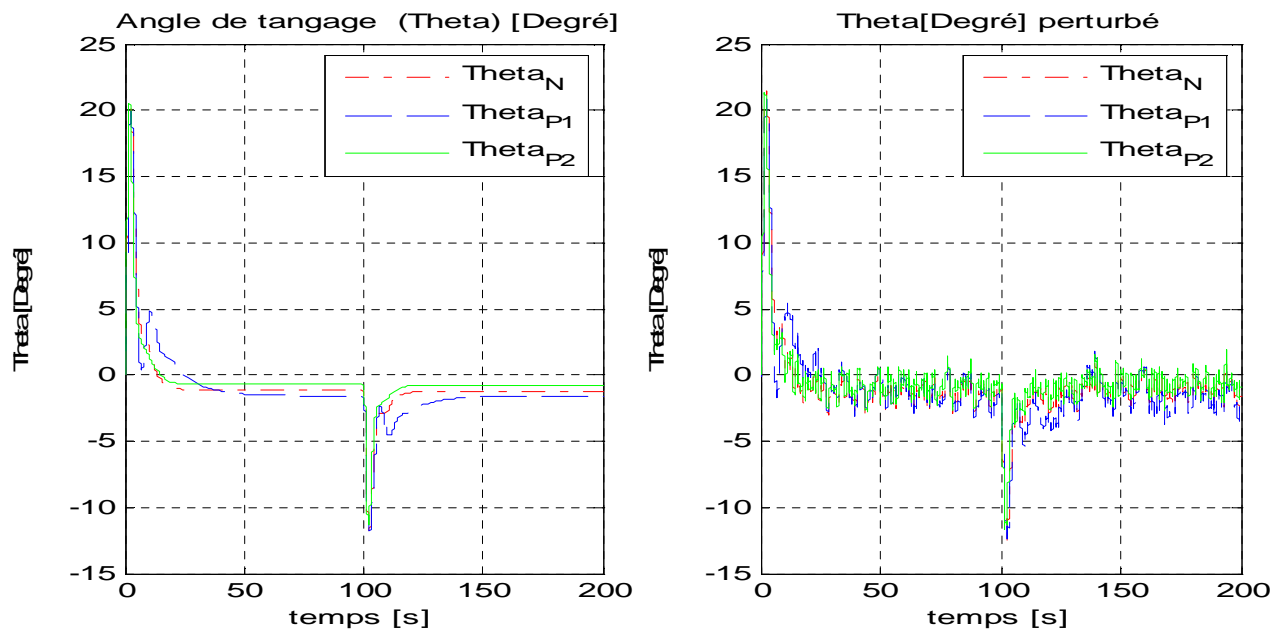


Figure 3.5 : Angle de tangage en fonction de temps.

$\Theta$  (tetha) est l'angle de tangage (angle d'assiette) nécessaire pour l'orientation de drone vers le haut et le bas surtout pour créer les forces aérodynamiques telle que la portance. Il est toujours

tenu inférieur à 22 degrés. Ce qui rentre dans les limitations des caractéristiques du drone utilisé dans le présent mémoire (Aerosonde).

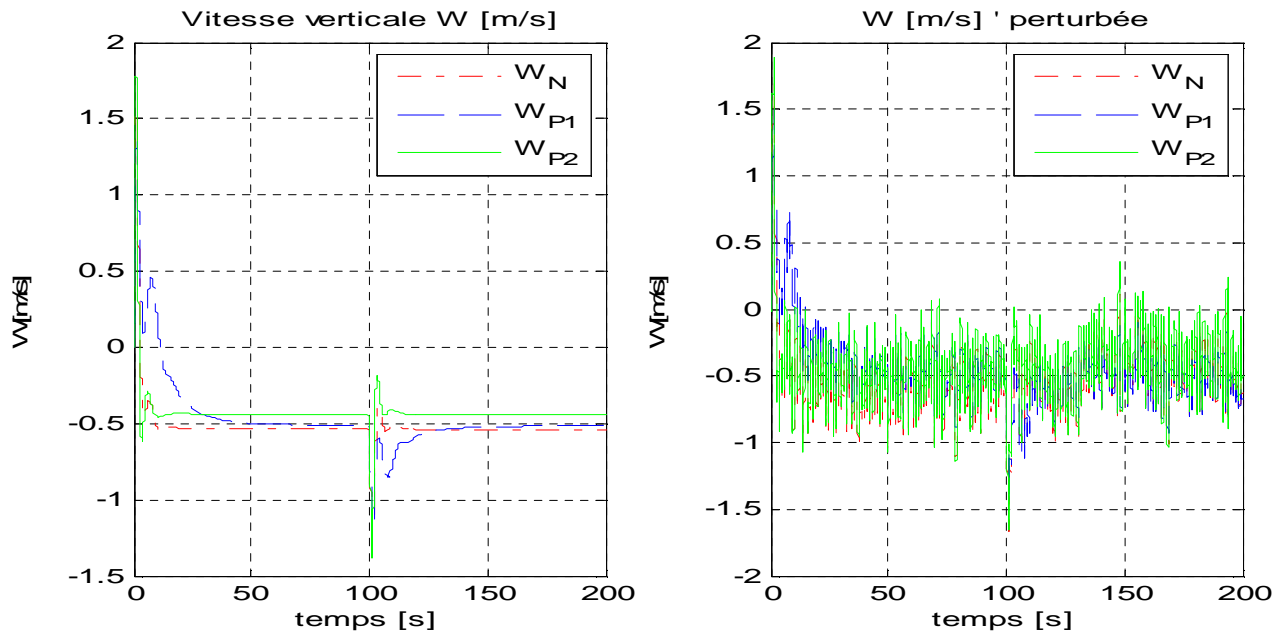


Figure 3.6 : Vitesse de translation verticale en fonction de temps.

L'augmentation brusque de la vitesse verticale est due à l'importance de l'amplitude de la consigne, donc le drone doit l'atteindre et à fur et à mesure que la consigne est atteinte la vitesse se maintient constante à -0.5 m/s ce qui est dû à l' $\alpha$  (angle d'attaque) -2 degrés en régime permanent.

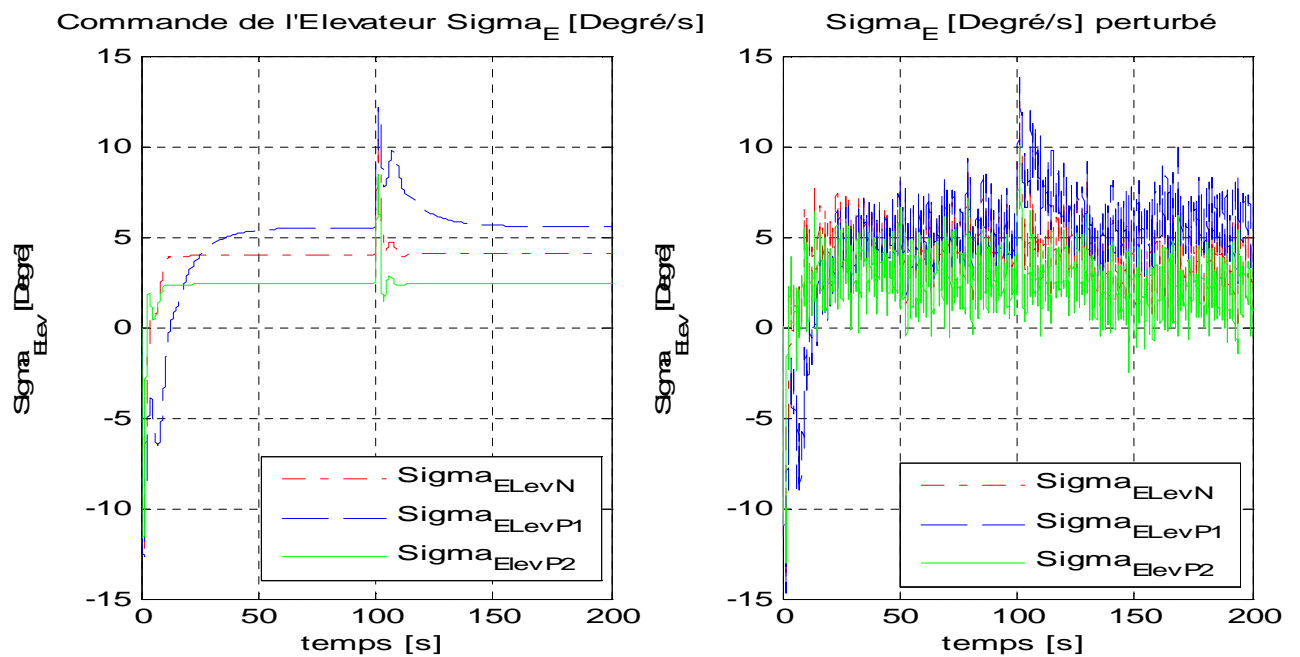


Figure 3.6 : Commande de la gouverne de profondeur en fonction de temps.

Nous remarquons que la commande des empennages de profondeur est toujours maintenue dans le domaine de non saturation des actionneurs  $\pm 22$  degrés .

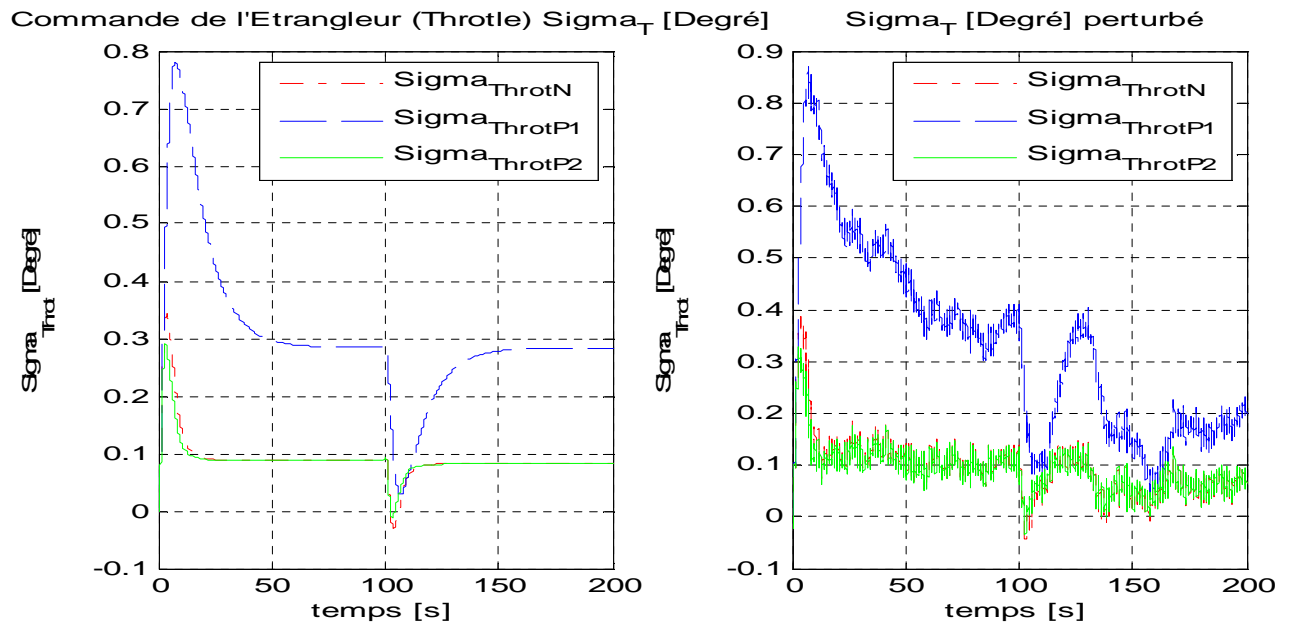


Figure 3.7 : Commande de l'étrangleur du moteur en fonction de temps.

Même chose pour la commande de l'étrangleur du moteur qui est tenue dans son domaine de fonctionnement. Donc les deux commandes (élevateurs et étrangleur) sont réalisables.

#### - **Mouvement latéral :**

Pour une meilleure analyse du mouvement latéral, nous proposons en premier lieu d'introduire les consignes comme suit :

- 1- La référence de direction est tenue à 0 degrés pendant les 50 secondes, et l'angle de roulis est maintenu à 15 degrés.
- 2- La référence de direction est tenue à -35 degrés après 50 secondes, et l'angle de roulis est maintenu à 0 degrés.

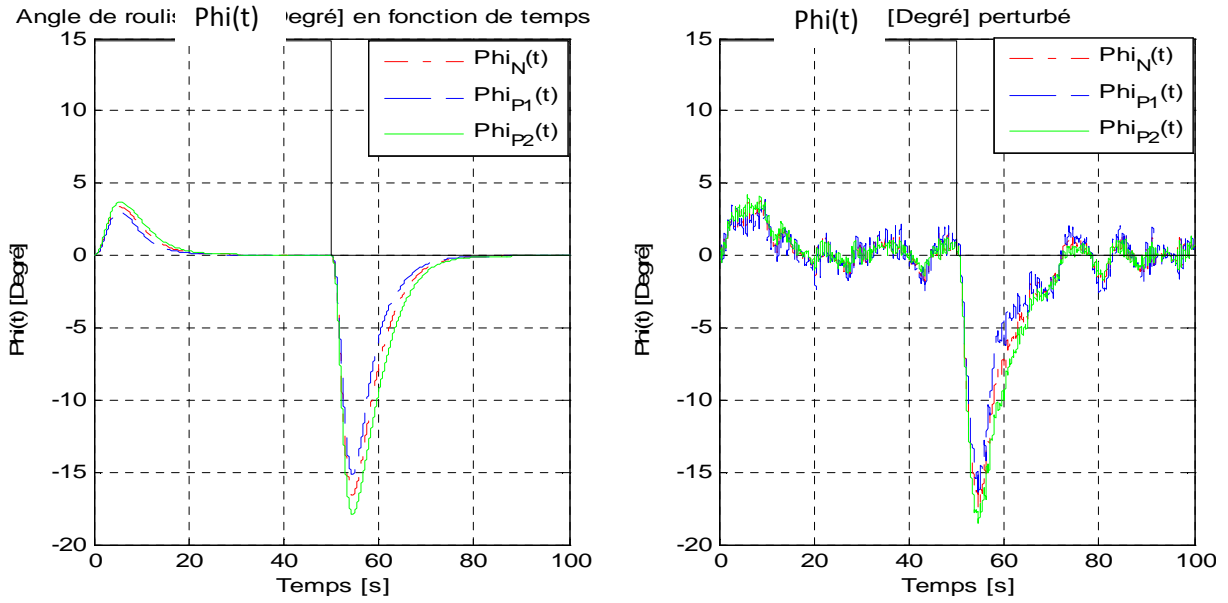


Figure 3.8 : Angle de Roulis en fonction de temps.

En donnant premièrement 15 degrés comme référence pour l'angle de roulis phi (bank angle) à travers les commande des ailerons, phi croit jusqu'à 4 degrés et commence à descendre jusqu'à son annulation et n'atteint pas la référence, ce qui attendu car le vol lors de modélisation est supposé symétrique. Malgré l'introduction des perturbations atmosphériques et les incertitudes des capteurs, phi oscille toujours autour de zéro.

Le pic apparent à 50 s est causé par le changement de cap, en donnant une consigne de -35 degrés, et nous remarquons que en dépit de couplage existant phi parvient à atteindre ou à osciller autour de zéro, même dans un environnement perturbé.

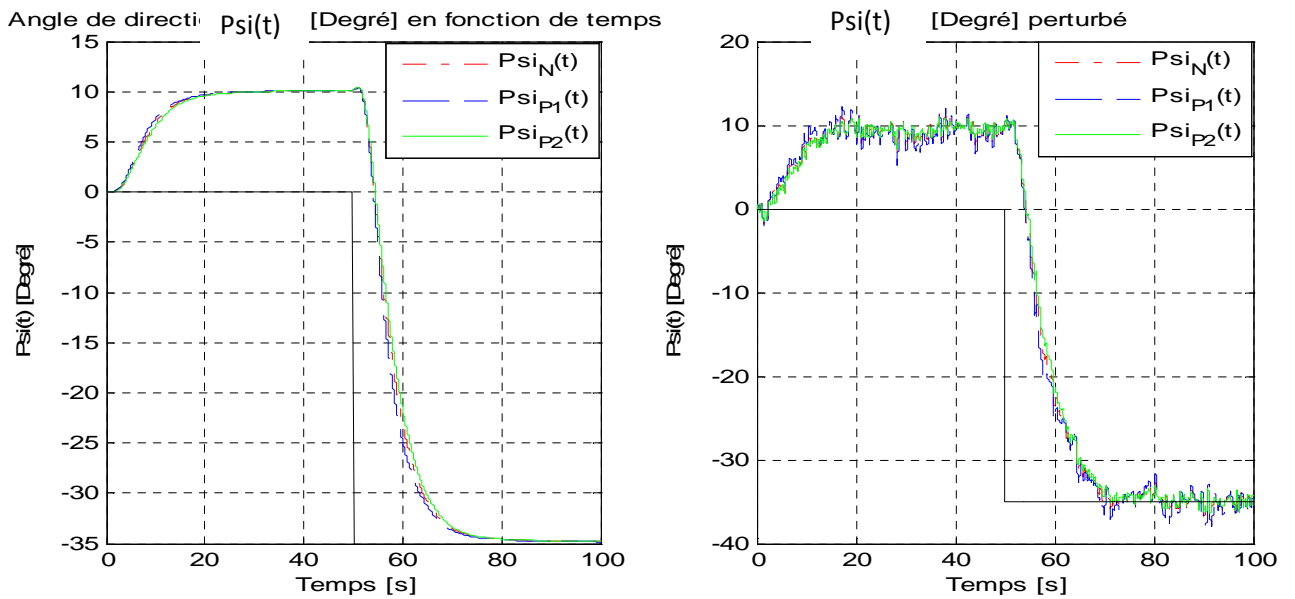


Figure 3.9 : Angle de Lacet en fonction de temps.

Malgré que la consigne du cap est nulle, l'angle de direction  $\psi$  est influencé par la commande des ailerons puisqu'ils sont fortement couplés, et à l'instant 50 s la consigne des ailerons est mise à zéro et celle de l'empennage de direction est réglée à  $-35$  degrés, et c'est ainsi que le drone poursuit le cap à  $-35^\circ$  avec une précision admissible de l'ordre de 2 degrés pour les modèles les plus perturbés.

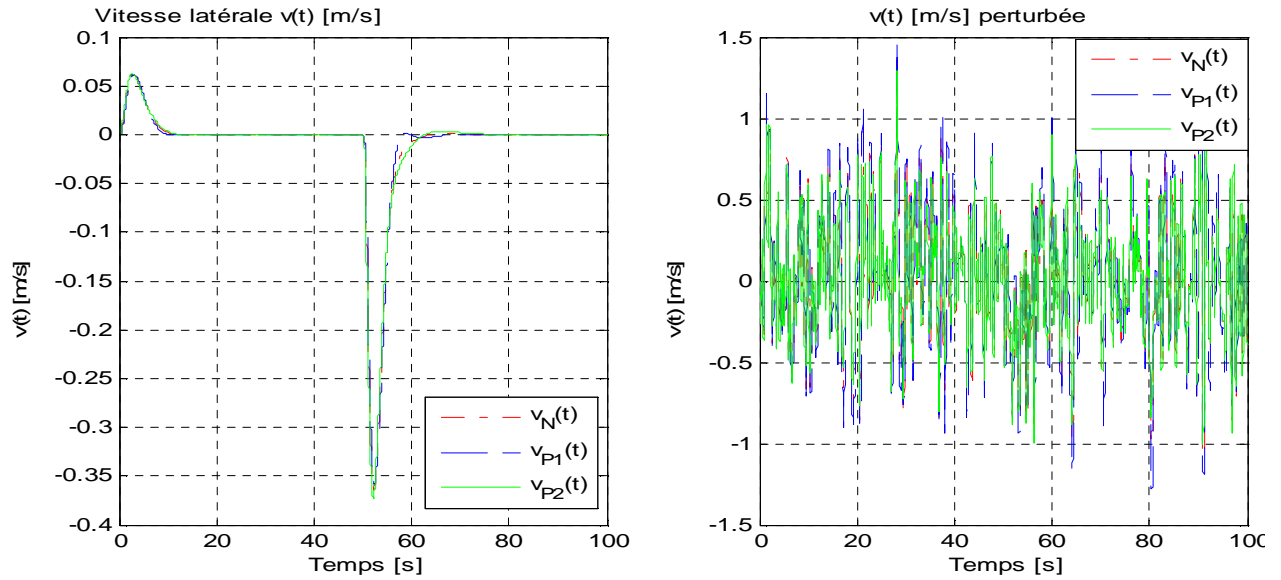


Figure 3.10 : Vitesse latérale en fonction de temps.

La vitesse de translation latérale est minimisée de sorte à éviter des dérives latérales, mais néanmoins un environnement agité cause de sérieuses dérives et pour remédier, le correcteur maintient cette vitesse autour de zéro en dépit des perturbations de hautes fréquences.

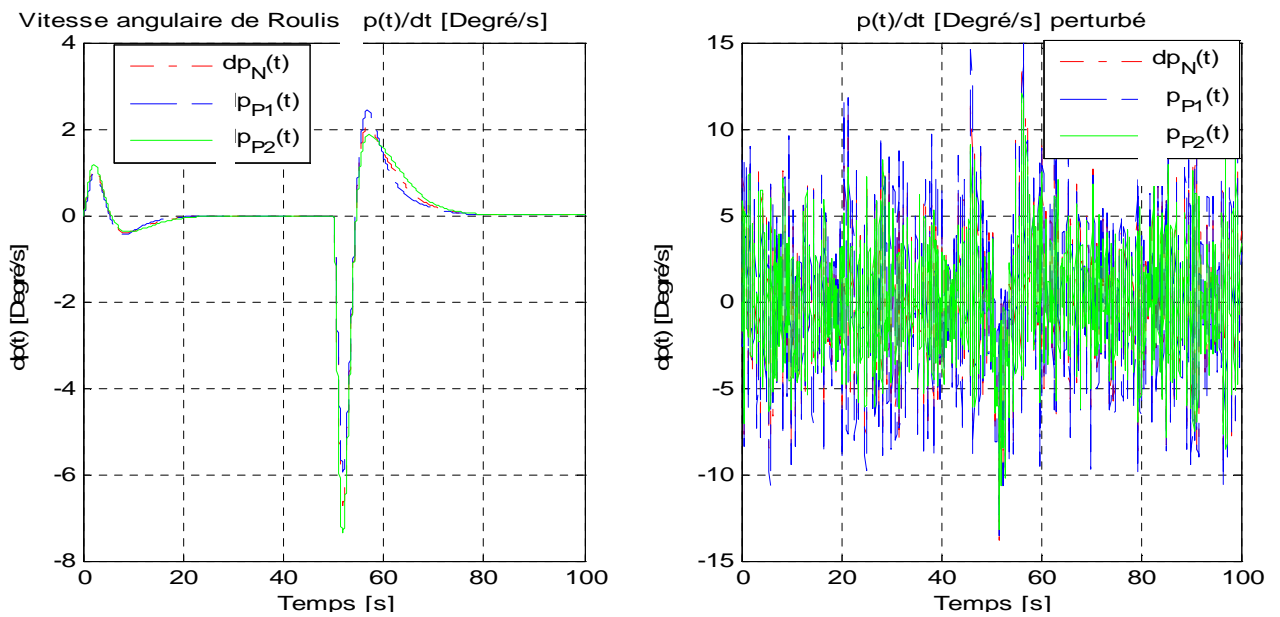


Figure 3.11 : Taux de variation de vitesse angulaire de Roulis en fonction de temps.

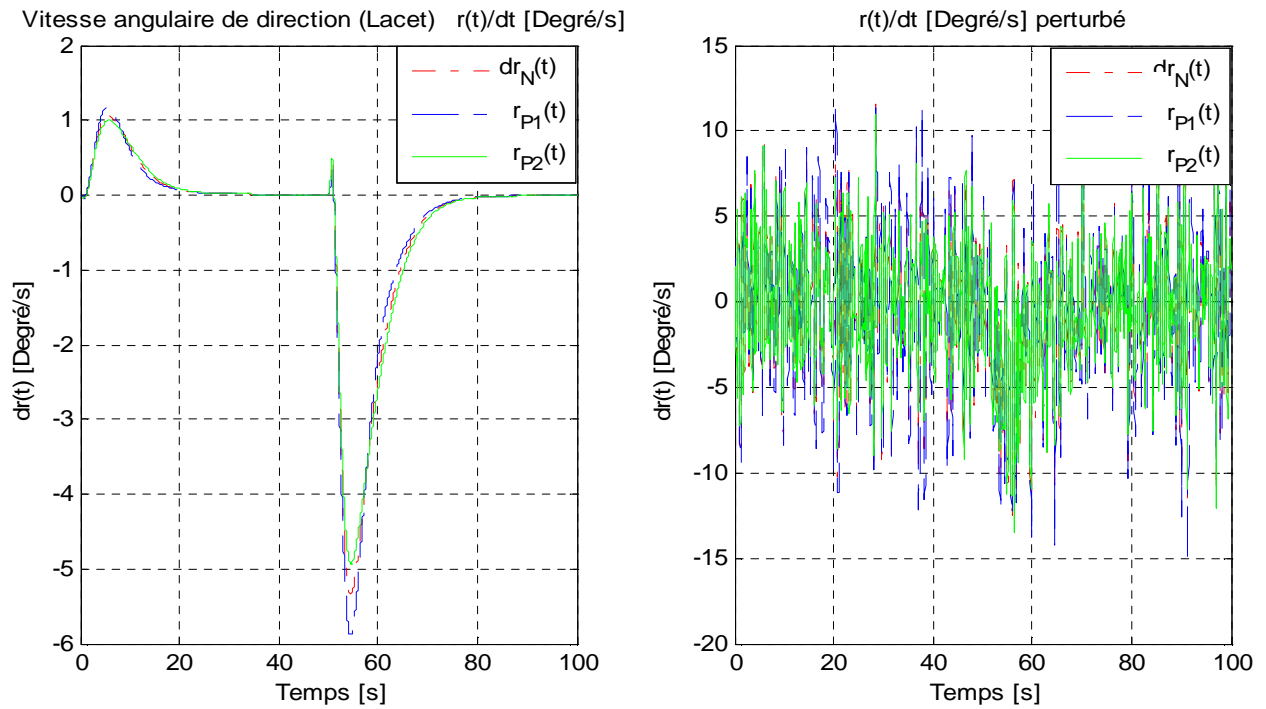


Figure 3.12 : Taux de variation de la vitesse angulaire du Lacet en fonction de temps.

Mêmes remarques pour les vitesses angulaires peuvent être faites, puisque le vol est supposé ; précédemment symétrique.

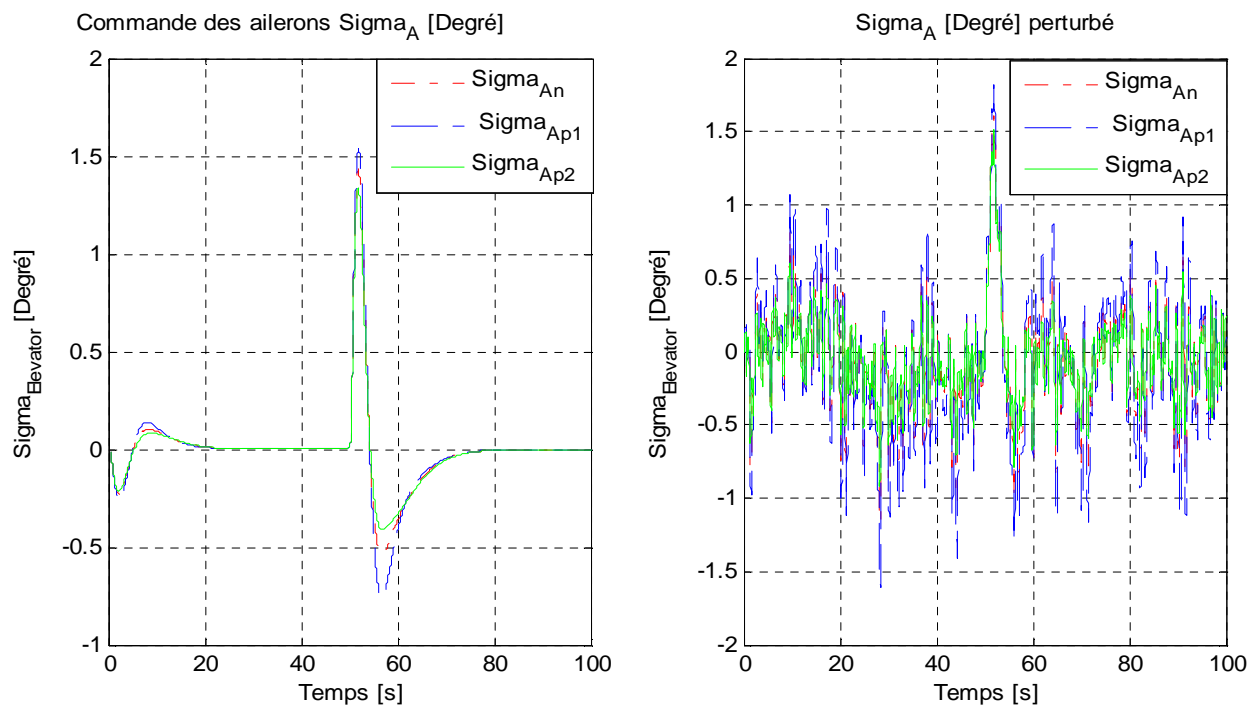


Figure 3.13 : Commande des Ailerons en fonction de temps.

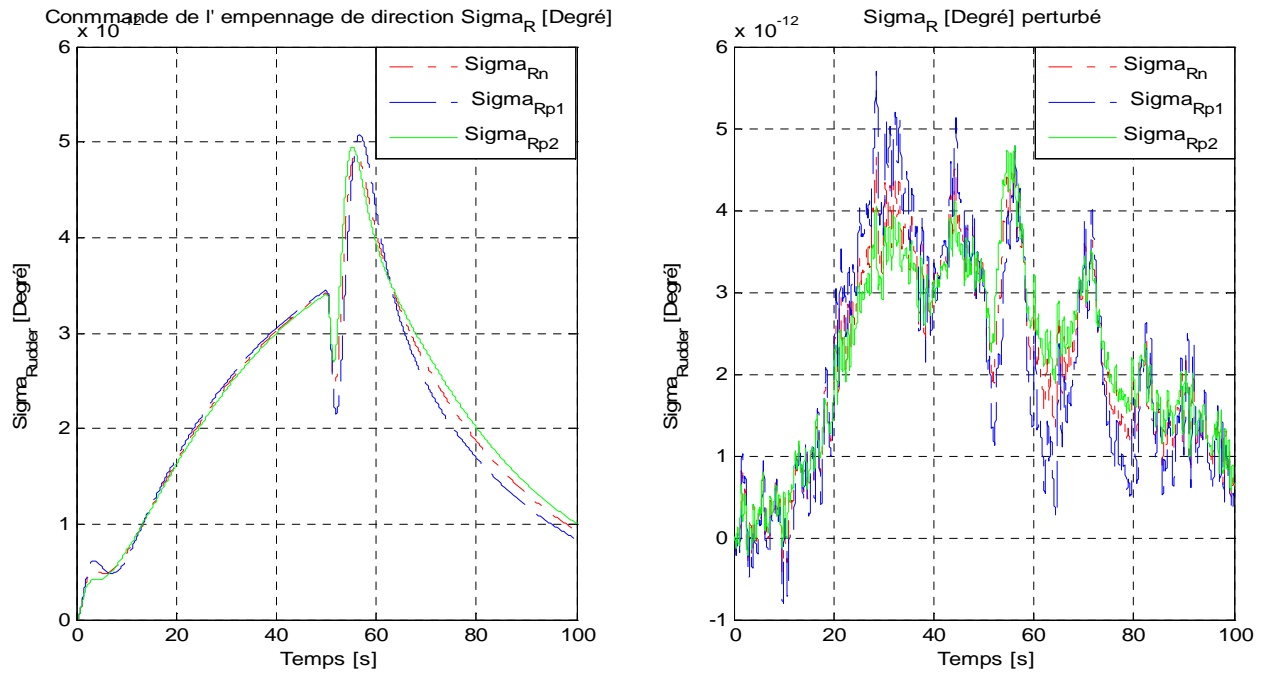


Figure 3.14 : Commande de l'empennage de direction en fonction de temps.

Les deux commandes latérales sont réalisables puis que leur apport énergétique reste dans le domaine de fonctionnement, mais néanmoins la déflexion de l'empennage de direction est nettement très inférieure à celle des ailerons.

### Conclusion

La simulation nous a permis de confirmer l'efficacité de correcteur sur les trois modèles nominal et perturbés en absence ou en présence des perturbations atmosphériques et les variations paramétriques, les modèles en boucle fermée sont toujours stables, ce qui prouve la robustesse de correcteur synthétisé.

CONCLUSION

GENERALE

### Conclusion générale

La conception d'une loi de commande pour des systèmes évoluant dans des situations aussi complexes ; tel qu'un drone ; est une tâche très difficile, et pour surmonter les diverses contraintes qui peuvent être rencontrées nous avons procédé comme suit :

- 1) Déterminer des modèles d'état (nominal et perturbés).
- 2) Synthétiser un correcteur optimal.
- 3) Utiliser les paramètres de du correcteur comme conditions initiales.
- 4) Choisir une fonction objectif selon le cahier de charge (critères de performance).
- 5) Construire un nouveau correcteur robuste en optimisant la fonction objectif avec les algorithmes génétiques.

Le fait de minimiser la fonction objectif, les normes  $H_2$  de la fonction de sensibilité  $S$  et les normes  $H_\infty$  de la fonction de sensibilité complémentaire  $T$ , nous a permis de trouver un compromis entre la performance et la robustesse.

La simulation a mis en évidence l'efficacité de la procédure suivie, elle a respecté les limitations imposées par les actionneurs de drone ce qui permet la réalisabilité et l'implémentation cette commande.

Il est souhaitable de vérifier l'efficacité de cette procédure en l'implémentant sur un modèle non-linéaire avec l'algorithme de poursuite d'une mission (Algorithme de Navigation).

ANNEXE

## Annexe:

La qualité de commande va être obtenue par le choix de la stratégie de commande, mais à conditions d'avoir les moyens de construction de la commande, durant ce qui suit nous proposons de construire une commande à base de la stratégie LQG (Linear Quadratic Gaussian) ; en premier temps ; qui est l'union de la stratégie LQR (Linear Quadratic Regulator) qui consiste à avoir une commande optimale sous forme d'un retour d'état, et un observateurs permettant d'avoir l'estimé de vecteur d'état.

### A.1. Commande Linéaire Quadratique (LQR):

#### A.1.1. Formulation de problème LQR :

Pour étudier cette stratégie de commande, nous proposons le système décrit par ces équations d'état qui est à la fois commandable et observable :

Nous avons dans cette configuration :

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t) \\ y(t) = Cx(t) + Du(t) \\ z(t) = Gx(t) + Hu(t) \end{cases}$$

Tels que :

$x(t)$  : Vecteur d'état, de dimension  $n \times 1$ .

$u(t)$  : Vecteur de commande, de dimension  $m \times 1$ .

$y(t)$  : Vecteur de sortie, de dimension  $q \times 1$ . Signaux disponibles en sortie (mesurables).

$z(t)$  : Vecteur sortie commandé, de dimension  $l \times 1$ . C'est l'ensemble des aux signaux qui doivent être minimisés sur le plus petit intervalle de temps possible.

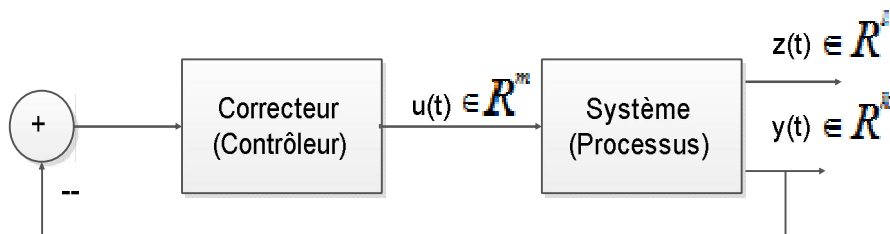


Figure A.1 : Schématisation de problème

La synthèse de correcteur consiste en la recherche d'un *matrice de gain*  $K$ , telle que la commande par un retour d'état stabilise le système et minimise à la fois. Et le problème de la commande en LQR est souvent posé comme suit :

$$\min J_{LQR} = \frac{1}{2} \int_0^{\infty} (z(t)'Qz(t) + u(t)'Ru(t)]dt$$

sujet à:

$$\dot{x} = Ax + Bu$$

$$z = Gx + Hu$$

$$u = -Kx$$

Tels que :

Q et R sont deux matrices symétriques définies positives dont l'ordre est  $l \times l$  et  $m \times m$  respectivement. Nous proposons par la suite de résoudre ce problème posé avec la méthode de principe de minimum (*Principe de Pontriaguine*).

### A.1.1.1. Principe du minimum de Pontriaguine :

Le principe du minimum de Pontriaguine est posé comme suit :

Soit le système d'équation d'état :

$$\dot{x} = f(x, u, t)$$

et le critère de performance :

$$J(x, u, t) = h(x, t)|_{t_0}^{t_f} + \frac{1}{2} \int_{t_0}^{t_f} g(x, u, t) dt$$

En appliquant le multiplicateur *Lagrange*, nous ramenons ce problème à un autre problème sans contraintes, et il devient :

$$J(x, u, t) = h(x, t)|_{t_0}^{t_f} + \frac{1}{2} \int_{t_0}^{t_f} (g(x, u, t) + L^T (f(x, u, t) - \dot{x})) dt$$

Nous définissons l'Hamiltonien du système :

$$H(x, u, p, t) = g(x, u, t) + L^T f(x, u, t)$$

Où  $L$  est appelé *état-adjoint* (du au multiplicateur de Lagrange). Le principe du minimum de Pontriaguine énonce que la trajectoire optimale minimise l'Hamiltonien du système. Autrement dit :

$$H(\tilde{x}, \tilde{u}, \tilde{L}) \leq H(x, u, L) \quad \forall u \in U$$

Le long de la trajectoire optimale, on dispose d'un certain nombre d'équations permettant de résoudre le problème de commande optimale. Ces équations sont généralement établies en utilisant le calcul variationnel (calcul des variations). La solution de problème

## Annexe

---

est donnée par un jeu d'équations, appelées équations canoniques de Hamilton, qui régissent les dynamiques de l'état d'une part et de l'état adjoint d'autre part :

- Etat

$$\frac{\partial H(x, u, L, t)}{\partial L} = \dot{x}$$

- Etat adjoint

$$\frac{\partial H(x, u, L, t)}{\partial x} = -\dot{L}$$

- Et le minimum de  $H(x, u, t)$  par rapport à  $u(t)$  est:

$$\frac{\partial H(x, u, L, t)}{\partial u(t)} = 0$$

Les équations provenant des conditions dites terminales, sont appelées les équations de transversalité.

$$\frac{\partial h(x, t)}{\partial x(t)} \Big|_{t_f} = L(t_f)$$

Maintenant que le principe de Pontriaguine est expliqué, résolvons le problème de la commande LQR entamé précédemment tel que :

$$\min J_{LQR} = \frac{1}{2} \int_0^{\infty} (z(t)' Q z(t) + u(t)' R u(t)) dt$$

sujet à:

$$\dot{x} = Ax + Bu$$

$$z = Gx + Hu$$

$$u = -Kx$$

Nous proposons de remplacer la commande  $z$  avec leurs expressions dans le critère à minimiser pour l'exprimer en fonction de  $x$  et  $u$ .

$$\min J_{LQR} = \frac{1}{2} \int_0^{\infty} ((Gx + Hu)^T Q (Gx + Hu) + u^T R u) dt$$

sujet à:

$$\dot{x} = Ax + Bu$$

Après une simple opération de distribution, nous obtenons :

## Annexe

---

$$\min J_{LQR} = \frac{1}{2} \int_0^{\infty} (x^T G^T Q G x + x^T G^T Q H u + u^T H^T Q G x + u^T H^T Q H u + u^T R u) dt$$

sujet à:

$$\dot{x} = Ax + Bu$$

En adaptant la méthode de Lagrange, nous pourrions exprimer l'Hamiltonien :

$$H(x, u, t) = \frac{1}{2} (x^T G^T Q G x + x^T G^T Q H u + u^T H^T Q G x + u^T H^T Q H u + u^T R u) + L^T (Ax + Bu)$$

En partant sur le principe de minimum, nous obtenons en dérivant l'Hamiltonien l'expression de la commande optimale  $u$  :

$$\frac{\partial H(x, u, t)}{\partial u} = \frac{1}{2} (0 + H^T Q G x + H^T Q G x + 2H^T Q H u + 2R u) + B^T L = 0$$

Rappelons que  $L = Px$ , et de ce fait :

$$u = -(H^T Q H + R)^{-1} (H^T Q G + B^T P) x$$

$$\dot{x} = \frac{\partial H(x, u, t)}{\partial x} = Ax + Bu$$

$$\dot{L} = \frac{\partial H(x, u, t)}{\partial x} = -\frac{1}{2} (2G^T Q G x + G^T Q H u + G^T Q H u) - A^T L$$

Il faut toujours garder dans nos esprits que notre but est d'avoir ; à la fin ; une commande à retour d'état tel que  $= -Kx$  , et il devient nécessaire de prendre  $L = Px$  donc.

$$\begin{aligned} \dot{L} &= \dot{P} x + P \dot{x} \\ &= \dot{P} x + P A x + P B u \\ &= \dot{P} x + P A x - P B (H^T Q H + R)^{-1} (H^T Q G + B^T P) x = -(G^T Q G x + G^T Q H u + A^T P x) \\ &= \dot{P} x + P A x - P B (H^T Q H + R)^{-1} (H^T Q G + B^T P) x \\ &\quad = -(G^T Q G x - G^T Q H (H^T Q H + R)^{-1} (H^T Q G + B^T P) x + A^T P x) \\ &\quad \left( \dot{P} + P A + A^T P + G^T Q G - (P B + G^T Q H) (H^T Q H + R)^{-1} (H^T Q G + B^T P) \right) x = 0 \end{aligned}$$

Nous arrivons à exprimer l'expression de  $K$  :

$$K = (H^T Q H + R)^{-1} (H^T Q G + B^T P)$$

Et  $P$  est l'unique solution de l'équation de Riccati

$$\dot{P} + P A + A^T P + G^T Q G - (P B + G^T Q H) (H^T Q H + R)^{-1} (H^T Q G + B^T P) = 0$$

## Annexe

Cette équation différentielle doit être résolue à chaque instant pour déterminer  $P(t)$ , et si les matrices  $A$ ,  $B$ ,  $G$ ,  $H$ ,  $Q$  et  $R$  sont constant et  $t_f$  tend vers l'infini, alors la matrice  $P(t)$  solution de l'équation de Riccati est une matrice indépendante de temps, donc la différentielle algébrique connue sous le nom de *l'équation algébriques de Riccati*:

$$PA + A^T P + G^T Q G - (PB + G^T Q H)(H^T Q H + R)^{-1}(H^T Q G + B^T P) = 0$$

La commande se fait donc par :

- 1- Formulation de problème.
- 2- Choix approprié des matrices  $Q$  et  $R$  (la difficulté se pose sur ce choix).
- 3- Résolution de l'équation de Riccati.
- 4- Calcul de la matrice de gain  $K$ .
- 5- Et enfin, détermination de la commande  $u = -Kx$ .

La figure A.2 montre schéma global de ce type de commande.

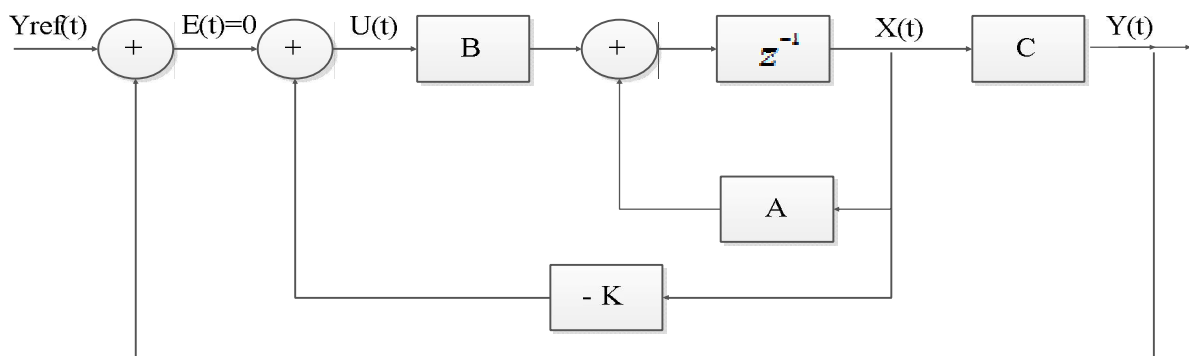


Figure A.2: Schéma général de la commande LQR

Sans restriction, les pondérations peuvent être choisies symétriques. Elles sont généralement choisies diagonales. Ainsi, on se ramène au choix de  $n$  scalaires pour l'état et de  $p$  scalaires pour la commande. Voici une méthode simple de choix et de modification des pondérations en vue d'aboutir à un correcteur satisfaisant.

1. Au départ, on choisit généralement des pondérations égales aux matrices identité.
2. Dans une seconde étape, on accélère ou décélère globalement le système en multipliant la matrice  $Q$  par un scalaire  $\lambda$  (accélération avec  $\lambda > 1$  et décélération avec  $\lambda < 1$ ), jusqu'à obtenir une dynamique moyenne adaptée.
3. Dans le cas où certains états auraient des dynamiques trop lentes par rapport à d'autres, on peut choisir d'augmenter la pondération de  $Q$  correspondant aux premiers.
4. Dans le cas où certains actionneurs seraient trop sollicités par rapport à

d'autres, on peut choisir d'augmenter la pondération de  $R$  leur correspondant.

Les étapes 2, 3 et 4 peuvent être réitérées dans l'ordre souhaité jusqu'à obtenir un correcteur satisfaisant le cahier des charges.

Durant la mise en place de correcteur LQR, nous rencontrons souvent des difficultés, car fréquemment le vecteur d'état n'est que partiellement disponible en sortie, donc la simplicité de la commande LQR est à la fois un avantage et un handicap, et pour y remédier, il faut lui associer un observateur de Kalman, et un filtre pour la mise en forme des signaux des capteurs en sortie, et donc LQR devient LQG pour une commande *Linière Quadratique Gaussienne*.

### **A.2. Commande linière quadratique gaussienne :**

Par rapport à la commande LQR, la commande LQG présente l'intérêt de s'appliquer à des systèmes dont l'état n'est pas mesuré (développée au début de la seconde moitié du 20ème siècle et appliquée lors du programme spatial Apollo pour la stabilisation de lanceurs), elle est apparue comme la première méthode générale pour l'asservissement des systèmes multi-variables. De ce fait, elle a connu un grand succès comme en témoigne les nombreuses publications sur le sujet. Depuis la fin du 20ème siècle, la commande  $H_\infty$  apparaît comme un sérieux concurrent pour l'asservissement robuste des systèmes multi-variables. Néanmoins, la commande LQG n'en demeure pas moins un standard industriel.

La synthèse LQG consiste donc à rechercher, à partir de cette mesure partielle, un régulateur qui minimise un critère quadratique de nature stochastique.

La commande LQG associe un observateur d'état basé sur le filtre de Kalman à la commande LQR. L'observateur est un système dynamique, dont les entrées sont constituées de l'entrée et de la sortie du système à commander, et qui a pour fonction de reconstruire l'état du système.

#### **A.2.1. Formulation de problème :**

Soit le système dynamique stochastique d'équation d'état :

$$\begin{aligned}\dot{x}(t) &= Ax(t) + Bu(t) + Mw(t) \\ y(t) &= Cx(t) + v(t)\end{aligned}$$

## Annexe

Où le bruit d'état  $w$  et le bruit de mesure  $v$  sont des bruits blancs centrés, de **variance**  $var(w) = E\{w^T w\} = W \geq 0$ ,  $var(v) = E\{v^T v\} = V > 0$  et la covariance des deux bruits  $N = cov(w, v) = E[w, v^T]$ . Le problème LQG consiste à minimiser le critère :

$$J(x_0, t_0, u) = \lim_{t_f \rightarrow \infty} E \left( \frac{1}{t_f} \int_{t_0}^{t_f} (x^T Q x + u^T R u) dt \right)$$

Où  $Q \geq 0$  et  $R > 0$ . Du fait des entrées des bruits d'état  $w$  et de sortie  $v$ , les grandeurs  $u$  et  $x$  sont des grandeurs stochastiques. Comme critère, il est ainsi naturel de s'intéresser à l'espérance d'une intégrale. Comme pour la commande LQ.

### A.2.2. Théorème de séparation :

La solution de ce problème de commande optimale de processus stochastique est bien connue sous le nom de théorème de séparation. Ce théorème énonce que la solution du problème est composée de deux parties :

- un observateur de Kalman permettant de donner l'estimée  $\hat{x}$  de  $x$  qui est non biaisée et à variance minimale (de la différence  $\min E([x - \hat{x}]^T [x - \hat{x}])$ ).
- la commande par retour d'état  $u = -K\hat{x}$  à appliquer au système, où  $K$  est calculé en considérant le problème LQ correspondant (en enlevant  $v$  et  $w$  de l'équation d'état et l'espérance dans le critère).

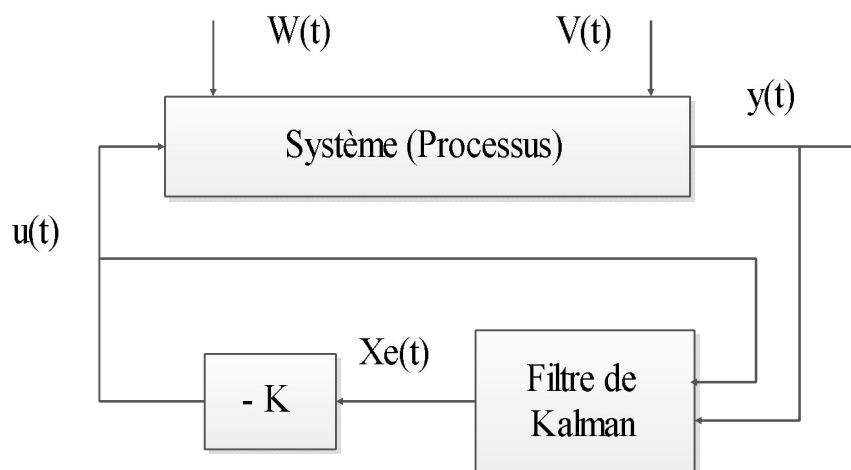


Figure A.3: Schéma de séparation de problème LQG

Ce théorème est intéressant dans la mesure où il donne la solution du problème complet comme étant la réunion des solutions de deux problèmes plus simples: celui de la commande LQ et celui de l'estimation de Kalman.

### A.3. Structure de la commande LQG :

L'estimée optimale  $\hat{x}$  est donnée par l'observateur d'équation d'état :

$$\dot{\hat{x}} = A\hat{x} + Bu + L(y - C\hat{x})$$

où le gain de Kalman est :

$$L = (N + PC^T)V^{-1}$$

Avec  $P$  la solution de l'équation algébrique de Riccati :

$$PA^T + AP - PC^TW^{-1}CP + V = 0$$

La commande étant donnée par  $u = -K\hat{x}$ , on peut réécrire les équations de la commande d'entrée  $y$  et de sortie  $u$  :

$$\begin{aligned}\dot{\hat{x}} &= (A - BK - LC)\hat{x} + Ly \\ u &= -K\hat{x}\end{aligned}$$

Le suivi d'une consigne  $y^d$  se fera par la loi de commande  $u = C(s)(y^d - y)$  où la fonction de transfert du correcteur est :

$$C(s) = K(sI_n - A + BK + LC)^{-1}L$$

Ses équations d'état sont :

$$\begin{aligned}\dot{\hat{x}} &= (A - BK - LC)\hat{x} + L Er \\ u &= -K\hat{x}\end{aligned}$$

Où  $Er = y^d - y$ . Notons que ce correcteur LTI a le même ordre que le processus.



# REFERENCES

## REFERENCES

1. Robust Flight Control. A Design Challenge / J. Magni, S. Bennani, J. Terlouw. –London, Berlin, New York etc., 1997. – 649 p.
2. Pratt R.W. Flight Control Systems: Practical issues in design and implementation / R. W. Pratt. – Edited by IEE Control Engineering Series 57, IEEE Publ., Padstowe, Cornwall, 2000. – 382 p.
3. Reigelsperger W. C. Application of Multivariable Control Theory to Aircraft Control Laws, final report: Multivariable Control Design Guidelines / W. C. Reigelsperger // Honeywell Technology Center, Lockheed Martin Skunk Works. May 1996. – 286 p.
4. Tunik A.A. Integrated Navigation, Guidance and Control Systems for Small Unmanned Aerial Vehicles / A.A. Tunik, I.K. Ahn, H. Ryu, V. Larin // Proceedings of the World Congress Aviation in the XXI-st Century. – Kyiv. – 2003. – P. 5.15–5.25.
5. Tunik A.A. Platform airship's robust stabilization in stochastic atmosphere / A.A. Tunik, I.K. Ahn, H. Ryu, C.H. Lim, H.C. Lee // Proceedings of the KSAS Fall Annual Meeting. Sejong Univ., Seoul. – 2001. – P. 293–299.
6. Tunik A.A. Robust Optimization of Control Law of Flight in Stochastically Disturbed Atmosphere/ A.A. Tunik, I.K. Ahn, C.H. Yeom // Proceedings of Millennium Conference on Air and Space Sciences, Technology and Industries. – Hankook Aviation University. – Apr. 28, 2000. – Seoul, Korea. – P. 122–133.
7. Roskam J. Airplane flight dynamics and automatic flights control Part II / J. Roskam // Design, Analysis and Research Corporation (DARcorporation). – 1998, 780 p.
8. Roskam J. Airplane aerodynamics and performance / J. Roskam, C. T. Lan // Design, Analysis and Research Corporation (DARcorporation). – 1997, 840 p.

9. Rauw M. The Flight Dynamics and Control Toolbox / M. Rauw. – MathWorks Company, 2000. – 263p.
10. McLean D. Automatic Flight Control Systems / D. McLean – Englewood: Prentice Hall Inc., 1990. – 593 p.
11. Aerosonde Robotic aircraft Web site: [www.aerosonde.com](http://www.aerosonde.com)
12. Aerosim blockset; Web site : [www.aerosonde.com](http://www.aerosonde.com)
13. Tunik A. A. Structured Parametric Optimization of Multivariable Robust Control based on Genetic Algorithms / A.A. Tunik, M.A. Touat // Proceedings of the National Aviation University. – №2. – 2008. – P. 10-17.
14. Touat M.A. Robust optimization of multivariable control system of uav lateral motion via genetic algorithms / M.A. Touat // Електроніка та системи управління. – 2009. – №1 (19). – С. 129 – 136.
15. Tunik A. A. The Parameter Optimization of Aircraft's Control Law from the Viewpoint of Some Airworthiness Requirements / A.A. Tunik, J.C. Kim, C.S. Yoo // Proceeding of the 12 Korea Automatic Control Conference „97 KACC”. – ICASE Publ. Seoul. – 1997. – P.1651–1654.
16. Tunik A.A. Parametric optimization procedure for robust flight control system design / A.A. Tunik, H. Ryu, H.C. Lee // KSAS International Journal. – 2001. – Vol. 2. – № 2. – P. 95–107.
17. Finn A. Miniature UAV's & Future Electronic Warfare / A. Finn, K. Brown, T. Lindsay // EW & Radar Division, DSTO, Edinburgh, SA 5111. – 8 p.
18. Huang Z.J. Integration of MEMS inertial sensors – based on GNC of a UAV / Z. J. Huang, J.C. Fang. // International Journal of Information technology. – Vol. 11. – №10. – 2005. – P. 123-132.
19. Alison K.B. GPS/INS uses Low-Cost MEMS IMU / K.B. Alison // NAVSYS Corporation. IEEE A&E System Magazine. – September 2005. – P. 3-10.
20. Advances in Unmanned Aerial Vehicles // Edited by Kimon P. Valavanis. – State of the Art and the Road to Autonomy, Springer. – 2007, 960 p.

21. Safonov M.G. Stability margins of diagonally perturbed multivariable feedback systems / M.G. Safonov // IEE Proc. D: Control theory and Applications. – 1982. – Nov. – P.251–256.
22. Doyle J. State–space solution to standard  $H_2$  and  $H_\infty$  control problems / J. Doyle, K. Glover, P. Khargonekar, B. Francis // IEEE Trans. on Automatic Control. – 1989. – Vol. 34. – № 8. – P. 831–847.
23. Kwakernaak H. Robust control and  $H_\infty$ – optimization / H. Kwakernaak // Tutorial paper. Automatica. – 1993. – Vol. 29. – № 2. – P. 255–273.
24. Schoemig E. Mixed  $H_2/H_\infty$  control of multi-model plants/ E.Schoemig, M. Sznaier // Journal of Guidance, Control and Dynamics, No.3, May-June, 1995, – P. 523–531.
25. Zhou K. Essential of robust control / K. Zhou, J. Doyle // Prentice Hall. May 25, 1999. 425 p.
26. Kwakernaak H. Linear Optimal Control Systems / H. Kwakernaak, R Sivan // Wiley Interscience, a Division of John Wiley & Sons, Inc. New York .ChiChester Brisbane Toronto. – 1972. – 563 p.
27. Kirk D.E. Optimal Control Theory, An Introduction / D.E. Kirk // Dover Publications, Inc. Mineola, New York. – 1998. – 472 p.
28. Zbigniew M. Genetic Algorithms + Data Structures = Evolution Programs / M. Zbigniew // Third edition, Revised and Extended Edition. Springer. – 1996. – 388 p.
29. Mitscell Melani. An introduction to genetic algorithms. 143 p.