

Master E3A 2ème année ISAS
Électronique, Énergie Électrique, Automatique

UE GNC

Rapport

➤ Fait par :

✓ Sidi DAGNON

➤ Encadreur de TP :

Thésarde Jaber Lara

2023-2024

Dagon,

TABLE DE MATIERE

2023-2024	1
I INTRODUCTION :.....	2
1. Objectif du TP :.....	2
2. Logiciels et outils:	2
II Modélisation et Présentation du Contexte:	2
1. Représentation d'état du système :	2
2. Commande du système :	3
3. Modèle simplifié du Système :	4
III. Application du TP	4
1. Chargement du fichier Simulink:	4
2. Simulation et Analyse des blocs:	5
3. Explication de loi de commande :	6
IV. Contrôle Longitudinal et Latéral	6
1. Forme LPV :	6
2. Modèle polytopiques:	8
3. Commande poly topiques et Stabilité :	9
4. Conditions LMI permettant le calcul des gains de la commande :	12
5. Observateur et structure :	13
V. Contrôle Vertical	15
1. Calcul des k_1, k_2, k_3	15
2. Simuler le modèle vertical	16
VI. Contrôle autonome	18

I INTRODUCTION :

1. Objectif du TP :

Les objectifs de ce TP sont :

- Concevoir un pilote automatique pour un drone par l'approche LPV et l'approche linéarisation exacte
- Se familiariser avec la modélisation LPV polytopique
- Se familiariser avec les outils d'optimisation LMI (Linear Matrix Inequality)
- Utilisation de l'analyse de stabilité au sens de Lyapunov

2. Logiciels et outils:

- *Matlab*
- *Simulink*

II Modélisation et Présentation du Contexte:

1. Représentation d'état du système :

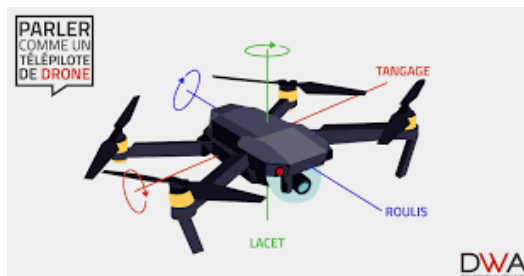
On s'intéresse dans ce TP au mouvement latéral d'un drone ayant des voilures dirigeables selon des angles dans deux directions perpendiculaires.

Le modèle non linéaire du drone est exprimé sous une forme de représentation d'état à 12 variables qui correspondent aux positions, vitesses, positions angulaires et vitesses angulaires, qui sera contrôlé via 4 commandes. Les 4 commandes du système sont respectivement :

- La poussée T ,
- Les 2 angles de rotation α et β ,
- La commande du roulis $M\psi$.



Angle de Roulie



Les variables d'état du système sont donc représentées comme :

$$x = \begin{pmatrix} x \\ u \\ y \\ v \\ z \\ \omega \\ \phi \\ p \\ \theta \\ q \\ \psi \\ r \end{pmatrix} = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \\ x_9 \\ x_{10} \\ x_{11} \\ x_{12} \end{pmatrix} \Rightarrow \dot{x} = \begin{pmatrix} \dot{x} \\ \dot{u} \\ \dot{y} \\ \dot{v} \\ \dot{z} \\ \dot{\omega} \\ \dot{\phi} \\ \dot{p} \\ \dot{\theta} \\ \dot{q} \\ \dot{\psi} \\ \dot{r} \end{pmatrix} = \begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \\ \dot{x}_5 \\ \dot{x}_6 \\ \dot{x}_7 \\ \dot{x}_8 \\ \dot{x}_9 \\ \dot{x}_{10} \\ \dot{x}_{11} \\ \dot{x}_{12} \end{pmatrix} = \begin{pmatrix} x_2 \\ \dot{x}_2 \\ x_4 \\ \dot{x}_4 \\ x_6 \\ \dot{x}_6 \\ x_8 \\ \dot{x}_8 \\ x_{10} \\ \dot{x}_{10} \\ x_{12} \\ \dot{x}_{12} \end{pmatrix}$$

Dans ce système, on a Trois Variables qui représente la position, 3 pour la vitesse et 6 pour les positions et vitesses angulaires.

Et le modèle du système est représenté par une représentation d'état comme suite :

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = x_{12}x_4 - x_{10}x_6 - g \sin x_9 - \frac{T}{m} \cos \beta \sin \alpha \\ \dot{x}_3 = x_4 \\ \dot{x}_4 = x_8x_6 - x_{12}x_2 + g \sin x_7 \cos x_9 + \frac{T}{m} \sin \beta \\ \dot{x}_5 = x_6 \\ \dot{x}_6 = x_{10}x_2 - x_8x_4 + g \cos x_7 \cos x_9 - \frac{T}{m} \cos \beta \cos \alpha \\ \dot{x}_7 = x_8 \\ \dot{x}_8 = \frac{I_{yy} - I_{zz}}{I_{xx}} x_{12}x_{10} + \frac{1}{I_{xx}} (-T \sin \beta (AB \cos \alpha + GA)) \\ \dot{x}_9 = x_{10} \\ \dot{x}_{10} = \frac{I_{xx} - I_{yy}}{I_{zz}} x_{12}x_8 + \frac{1}{I_{zz}} (-GA.T \cos \beta \sin \alpha) \\ \dot{x}_{11} = x_{12} \\ \dot{x}_{12} = \frac{I_{xx} - I_{yy}}{I_{zz}} x_8x_{10} + \frac{1}{I_{zz}} (AB.T \sin \beta \sin \alpha) + \frac{1}{I_{zz}} M_\psi \end{cases}$$

2. Commande du système :

D'après la représentation d'état du système qui a été faite dans la section précédente, on peut déduire les commandes qui régissent notre système modélisé. Mais étant toujours dans l'idée de simplifier le système, on suppose que notre système est commandé par des angles assez faibles. Ceci nous permet en effet de pouvoir, simplifier les termes de cosinus et sinus pour des cas de faibles amplitudes. On obtient donc comme commande :

$$\begin{cases} u_1 = -\frac{T}{m}\alpha \\ u_2 = \frac{T}{m}\beta \\ u_3 = -\frac{T}{m} \\ u_4 = \frac{T_{AB}}{I_{zz}}\alpha\beta \\ u_5 = \frac{1}{I_{zz}}M_\psi \end{cases}$$

:

3. Modèle simplifier du Système :

En considérant les commandes simplifiées de notre système, on peut réécrire la représentation d'état du système comme suite :

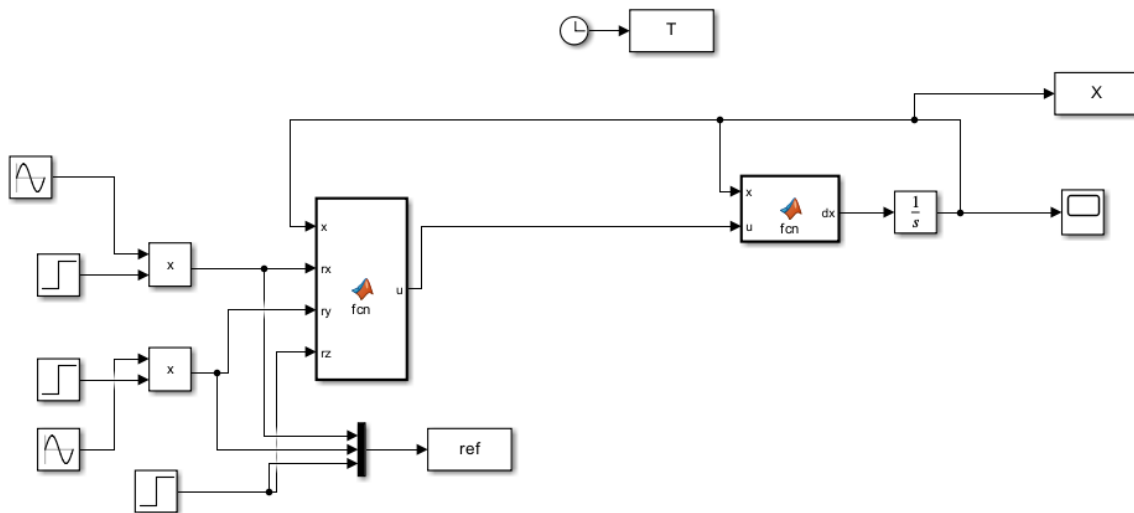
$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = x_{12}x_4 - x_{10}x_6 - g \sin x_9 + u_1 \\ \dot{x}_3 = x_4 \\ \dot{x}_4 = x_8x_6 - x_{12}x_2 + g \sin x_7 \cos x_9 + u_2 \\ \dot{x}_5 = x_6 \\ \dot{x}_6 = x_{10}x_2 - x_8x_4 + g \cos x_7 \cos x_9 + u_3 \\ \dot{x}_7 = x_8 \\ \dot{x}_8 = a_1x_{12}x_{10} + b_1u_2 \\ \dot{x}_9 = x_{10} \\ \dot{x}_{10} = a_2x_{12}x_8 + b_2u_1 \\ \dot{x}_{11} = x_{12} \\ \dot{x}_{12} = a_3x_8x_{10} + u_4 + u_5 \end{cases}$$

Avec bien sûr les termes :

$$\begin{cases} a_1 = \frac{I_{yy} - I_{zz}}{I_{xx}} \\ a_2 = \frac{I_{zz} - I_{xx}}{I_{yy}} \\ a_3 = \frac{I_{xx} - I_{yy}}{I_{zz}} \end{cases} \text{ et } \begin{cases} b_1 = \frac{m}{I_{zz}}(AB + GA) \\ b_2 = -\frac{m}{I_{yy}}GA \end{cases}$$

III. Application du TP

1. Chargement du fichier Simulink:



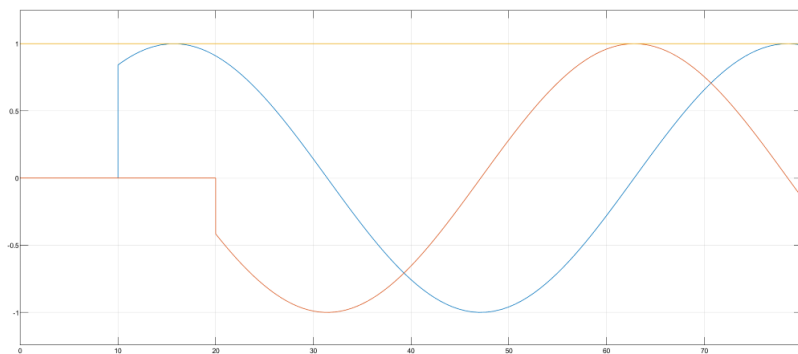
2. Simulation et Analyse des blocs:

- Analyse de blocs :

Dans le modèle Simulink nous avons en réalité, 2 principales blocs :

- **Bloc de Commande :** Pour représenter la commande du système, un bloc type fonction a été utilisé. Il décrit une fonction qui prend en paramètres les références du systèmes et l'état du système. Ensuite la fonction permet de donner en sortie la commande du système.

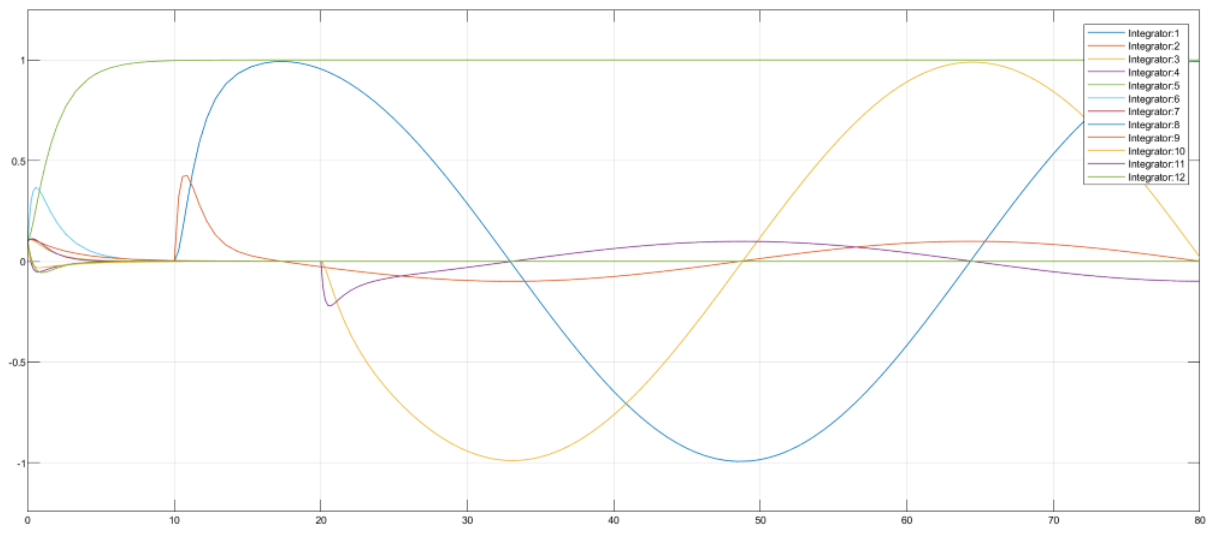
La Référence dans ce contexte, c'est comme la consigne qu'on veut atteindre pour le système. On a trois différentes références pour chaque axe de position. Les références rx et ry sont des produits de signaux de sinus et d'échelon et la référence rz est juste un échelon. :



- **Bloc de système :** Le système dans ce contexte est représenté aussi par un bloc de type fonction qui prend en entrée l'état et la commande du système. Il donne ensuite en sortie la dérivée de l'état qui permet de modéliser le système sous forme d'état.

- **Simulation :**

A la sortie du bloc de système, on place un intégrateur qui permet de visualiser les 12 variables d'états qui définissent notre système. Ils représentent les positions, les vitesses et les angles.



3. Explication de loi de commande :

Cette loi de commande utilise des rétroactions proportionnelles pour ajuster les commandes de contrôle du système en fonction des erreurs entre les états actuels et les références, tout en prenant en compte les caractéristiques dynamiques du système, les moments d'inertie, et les effets de la gravité. Les constantes de gain jouent un rôle crucial dans la stabilité et la réponse du système.

IV. Contrôle Longitudinal et Latéral

1. Forme LPV :

- **Long :**

$$\dot{x} = f(x, u) = \begin{bmatrix} x_2 \\ -g \sin(x_9) \\ x_{10} \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ u_1 \\ 0 \\ b2 * u_1 \end{bmatrix}$$

$$f(0,0) = 0 \text{ alors } \dot{x} = Ax$$

$$\dot{x} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{-g \sin(x_9)}{x_9} & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_9 \\ x_{10} \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \\ 0 \\ b_2 \end{bmatrix} * u_1$$

$$\rho(t) = \frac{g \sin(x_9)}{x_9} \text{ alors } r = 2$$

La forme est quasi-LPV.

• **Lat :**

$$\dot{x} = f(x, u) = \begin{bmatrix} x_4 \\ g \sin(x_7) \\ x_8 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ u_2 \\ 0 \\ -b_1 * u_2 \end{bmatrix}$$

$$f(0,0) = 0 \text{ alors } \dot{x} = Ax$$

$$\dot{x} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{g \sin(x_7)}{x_7} & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_3 \\ x_4 \\ x_7 \\ x_8 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \\ 0 \\ -b_1 \end{bmatrix} * u_2$$

$$\rho_2(t) = \frac{g \sin(x_7)}{x_7} \text{ alors } r = 2$$

La forme est quasi-LPV.

2. Modèle polytopiques:

- **Long :**

$$\rho_{min}(t) \leq \rho(t) \leq \rho_{max}(t)$$

$$H_1(t) = \frac{\rho(t) - \rho_{min}(t)}{\rho_{max}(t) - \rho_{min}(t)} \quad et \quad H_2(t) = \frac{\rho_{max}(t) - \rho(t)}{\rho_{max}(t) - \rho_{min}(t)}$$

$$A_1 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & -\rho_{max}(t) & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad et \quad A_2 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & -\rho_{min}(t) & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$Et \quad B_1 = B_2 = B = \begin{bmatrix} 0 \\ 1 \\ 0 \\ b_2 \end{bmatrix}$$

La forme Poly Topique est Globale car $\rho(t)$ est un sinus cardinal et est borné :

$$-0.21 \leq \frac{\sin(x_9)}{x_9} \leq 1$$

On peut maintenant réécrire le système sous forme de modèle d'état comme suite :

$$\dot{X}_{Long} = \sum_{i=1}^2 H_i * (A_i x + B_i u)$$

- **Lat :**

$$\rho_{2min}(t) \leq \rho_2(t) \leq \rho_{2max}(t)$$

$$H_1(t) = \frac{\rho_2(t) - \rho_{2min}(t)}{\rho_{2max}(t) - \rho_{2min}(t)} \quad et \quad H_2(t) = \frac{\rho_{2max}(t) - \rho_2(t)}{\rho_{2max}(t) - \rho_{2min}(t)}$$

$$A_1 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & \rho_{2max}(t) & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \text{ et } A_2 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & \rho_{2min}(t) & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\text{Et } B_1 = B_2 = B = \begin{bmatrix} 0 \\ 1 \\ 0 \\ -b_1 \end{bmatrix}$$

La forme Poly Topique est Globale car $\rho_2(t)$ est un sinus cardinal et est borné :

$$-0.21 \leq \frac{g \sin(x_7)}{x_7} \leq 1$$

On peut maintenant réécrire le système sous forme de modèle d'état comme suite :

$$\dot{X}_{Lat} = \sum_{i=1}^2 H_i * (A_i x + B_i u)$$

3. Commande poly topiques et Stabilité :

- **Commande Poly Topique :**

La loi de commande poly topique est donnée par :

$$U = - \sum_{i=1}^2 H_i * (K_i x)$$

Avec x l'état en longitudinal et l'état en latéral.

En boucle fermée, les systèmes sont décrits donc par les équations :

- **Long :**

$$\dot{X}_{Long} = \sum_{i=1}^2 H_i * (A_i - B_i K_i) X_{Long}$$

- **Lat :**

$$\dot{X}_{Lat} = \sum_{i=1}^2 H_i * (A_i - B_i K_i) X_{Lat}$$

- Posons : $\sum_{i=1}^2 H_i * (A_i - B_i K_i) = \varphi$

- **Stabilité des systèmes :**

- **Théorie**

La stabilité de ces systèmes, est étudié avec la condition de stabilité de Lyapunov. En réalité ce théorème évalue juste l'énergie du système pour conclure sur sa stabilité.

Pour cela, on définit une fonction quadratique de Lypunov :

$$V(x) = x^T P x$$

La fonction V, satisfait les condition : $V > 0$, $V(0)=0$; $V(+\infty)=+\infty$

Et aussi la matrice $P=P^T > 0$

La dérivée de cette fonction doit est négative pour que les systèmes soient stables.

$$\dot{V}(x) = \dot{x}^T P x + x^T P \dot{x}$$

En remplaçant et développant cette relation, on aboutit à :

- **Long**

$$\dot{V}(x)_{Long} < 0 \Leftrightarrow \sum_{i=1}^2 H_i * ((A_i - B_i K_i)^T P + P * (A_i - B_i K_i))$$

- **Lat**

$$\dot{V}(x)_{Lat} < 0 \Leftrightarrow \sum_{i=1}^2 H_i * ((A_i - B_i K_i)^T P + P * (A_i - B_i K_i))$$

La condition suffisante pour que la dérivée de ces fonctions de Lypunov soit négative :

$$((A_i - B_i K_i)^T P + P * (A_i - B_i K_i)) < 0 \text{ alors } \dot{V}(x) < 0$$

Mais en réalité ce problème ne peut pas être résolu car on a un cas de non-linéarité de type bilinéarisé : $P * (B_i K_i)$. Le gain K n'est pas connu et la matrice P aussi n'est pas

connue.

- **Lemme Congruence :** pour pouvoir notre problème d'optimisation, on est ramené à utiliser le lemme congruence sur l'équation précédente. Et on trouve :

$$(P)^{-1}(A_i - B_i K_i)^T + (A_i - B_i K_i) * (P)^{-1} < 0$$

Remarque : Pour chaque pair (A_i, B_i) on a 2 équations LMI. Dans notre cas on aura deux équations LMI car $B_1 = B_2$

- **Pratique**

Cette stabilité est vérifiée en pratique en vérifiant s'il existe un gain K tel que les parties réelles des pôles des matrices d'évolution en boucle fermée sont négatives :

$$A_i - B_i K_i$$

A l'aide de Matlab on peut résoudre ce problème. Le code est fourni en annexe de ce rapport et aussi en document attaché.

Résultat :

```
La valeur du K1Long est:-13.37502      -53.36728      186.7738      75.41066
La valeur du K2Long est:-13.37502      -53.36728      186.7738      75.41066
La valeur du mode 1 est:
-9.2269 + 0.0000i
-6.2709 + 3.5175i
-6.2709 - 3.5175i
-0.2748 + 0.0000i

La valeur du mode 2 est:
-10.6879 + 6.6876i
-10.6879 - 6.6876i
-0.3338 + 0.2484i
-0.3338 - 0.2484i
```

```

La valeur du K1Lat est:5.64884      15.2769      57.2573      -3.68604
La valeur du K2Lat est:11.5273     32.1358     90.6309     -12.6153
La valeur du mode 1 est:
-3.3856 + 4.1053i
-3.3856 - 4.1053i
-4.3727 + 0.0000i
-0.4471 + 0.0000i

La valeur du mode 2 est:
-9.4168 + 0.5263i
-9.4168 - 0.5263i
-0.3435 + 0.3856i
-0.3435 - 0.3856i

```

Commentaire :

On peut bien observer sur ce résultat que tous les pôles du système ont une partie réelle négative. ; Donc on peut conclure pour dire que le système est stable au sens de Lyapunov sur la longitude et aussi sur la latitude.

4. Conditions LMI permettant le calcul des gains de la commande :

Dans cette partie, on met l'accent théorique sur les conditions qu'on a utilisé plus haut pour démontrer la stabilité.

On avait défini les condition de stabilité au sens de Lyapunov comme suite :

$$\dot{V}(x)_{Lat} < 0 \Leftrightarrow \sum_{i=1}^2 H_i * ((A_i - B_i K_i)^T P + P * (A_i - B_i K_i))$$

$$((A_i - B_i K_i)^T P + P * (A_i - B_i K_i)) < 0 \text{ alors } \dot{V}(x) < 0$$

• Lemme Congruence :

Pour pouvoir résoudre notre problème d'optimisation, on a utilisé le lemme congruence sur l'équation précédente. Et on trouve :

$$(P)^{-1}(A_i - B_i K_i)^T + (A_i - B_i K_i) * (P)^{-1} < 0$$

Alors en développant on arrive à :

$$(P)^{-1} * A_i^T + A_i * (P)^{-1} - (P)^{-1} * K_i^T * B^T - B * K_i * (P)^{-1} < 0$$

On pose : $Q=(P)^{-1}$ et $\bar{K}_i = K_i * (P)^{-1}$ alors :

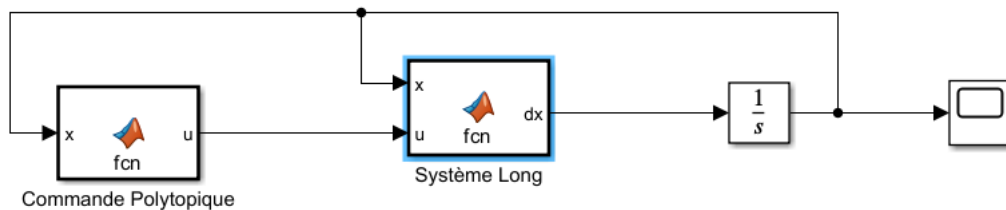
$$Q * A_i^T + A_i * Q - \bar{K}_i^T * B^T - B * \bar{K}_i < 0$$

Les conditions LMI se déduisent donc comme :

$$Q > 0$$

$$Q * A_i^T + A_i * Q - \bar{K}_i^T * B^T - B * \bar{K}_i < 0$$

5. Observateur et structure :



```
function u = fcn(x)

    g=9.8;
    if x(3)==0
        ro=1
    else
        ro=(g*sin(x(3))/x(3));
    end
    Ro1max=g;
    Ro1min=0.21*g;
    h1=(ro-Ro1min)/(Ro1max-Ro1min);
    h2=(Ro1max-ro)/(Ro1max-Ro1min);

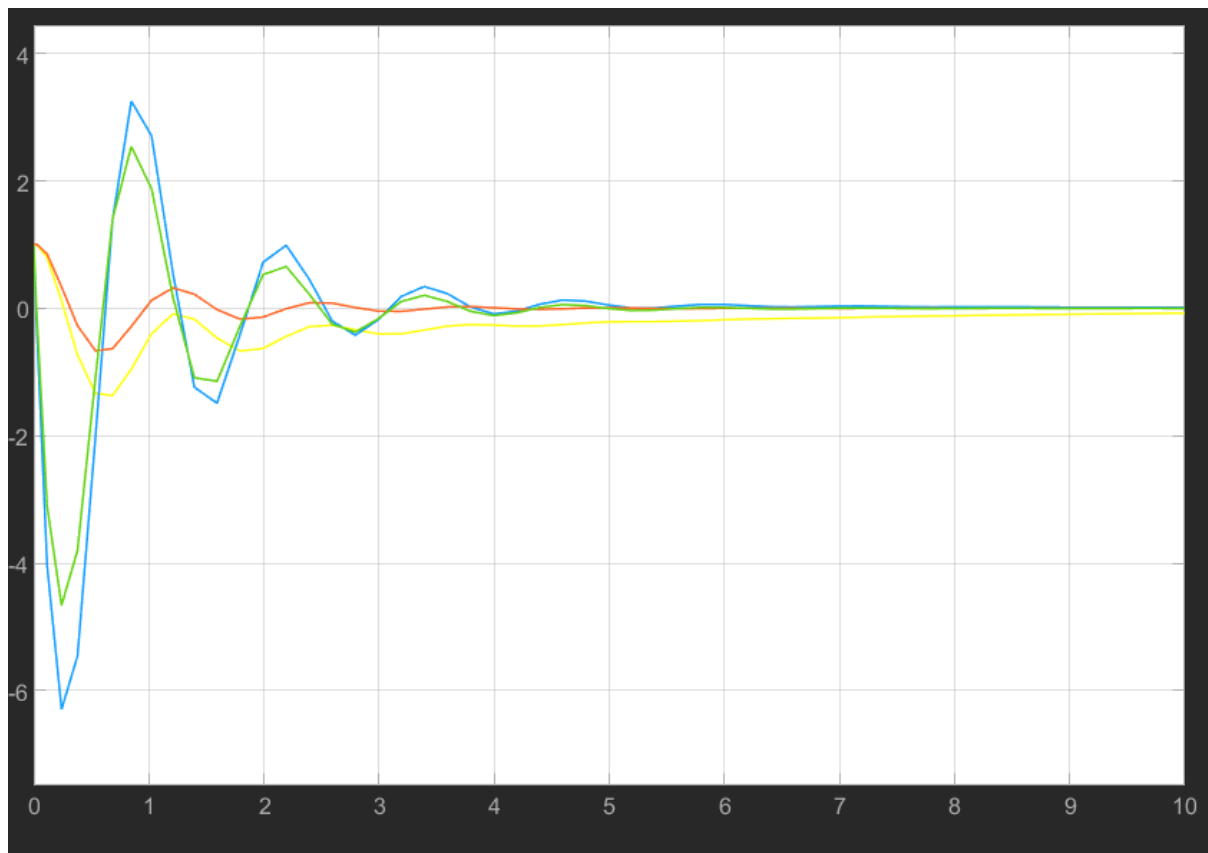
    K1Long = [-1.96018      -9.60731      35.603      15.0988];
    K2Long=[-4.5747      -20.6802      66.1937      29.9611];

    u =-h1*K1Long*x-h2*K2Long*x ;
```

```

1  function dx = fcn(x,u)
2
3      dx=zeros(4,1);
4      g=9.8;
5      if x(3)==0
6          ro=1
7      else
8          ro=(g*sin(x(3))/x(3));
9      end
10
11      Ro1max=g;
12      Ro1min=0.21*g;
13      A1Long=[0 1 0 0; 0 0 -Ro1max 0; 0 0 0 1; 0 0 0 0];
14      A2Long=[0 1 0 0; 0 0 -Ro1min 0; 0 0 0 1; 0 0 0 0];
15      B1Long=[0;1;0;1];
16      C1Long=[1; 0; 0; 1];
17
18      h1=(ro-Ro1min)/(Ro1max-Ro1min);
19      h2=(Ro1max-ro)/(Ro1max-Ro1min);
20      %A=zeros(4,4);
21      A=h1*A1Long+h2*A2Long;
22      dx=A*x+B1Long*u;
23

```



Commentaire :

D'abord quand on représente la structure simple de du système on peut essayer de remarquer si la commande poly topique seule appliquer, le système converge.

Sur le graphe on observe que pour chaque état du système, il convergence. On peut donc essayer d'appliquer la structure de l'observateur pour voir si on peut estimer l'état du système.

Calcul du gain de l'observateur :

Theorem 3 Le système en boucle fermée avec une commande par retour d'état estimé est asymptotiquement stable s'il existe deux matrices symétriques Q et P_2 , des gains F_i et R_i et un scalaire η tel que les LMIs suivantes sont vérifiées :

$$\begin{pmatrix} QA_i^T + A_iQ - B_iF_j - F_j^T B_i^T & B_iF_j & 0 \\ F_j^T B_j^T & -2\eta Q & \eta I \\ 0 & \eta I & A_i^T P_2 + P_2 A_i - R_i C - C^T R_i^T \end{pmatrix} < 0$$
$$i, j = 1, 2$$

Les gains sont obtenus par :

$$K_i = F_i Q^{-1}$$
$$L_i = P_2^{-1} R_i$$

V. Contrôle Vertical

1. Calcul des k1, k2, k3...

$$\begin{cases} \dot{x}_5 = x_6 \\ \dot{x}_6 = x_2 x_{10} - x_4 x_8 + g \cos(x_7) \cos(x_9) + u_3 \\ \dot{x}_{11} = x_8 \\ \dot{x}_{12} = a_3 x_8 x_{10} + \frac{1}{I_{zz}} M_3 + u_4 \end{cases}$$

$$u_3 = -x_2 x_{10} + x_4 x_8 - g \cos(x_7) \cos(x_9) - k_1 x_5 - k_2 x_6$$
$$u_4 = -a_3 x_8 x_{10} - \frac{1}{I_{zz}} M_3 - k_3 x_{11} - k_4 x_{12}$$

$$\dot{x} = f(x, u) = \begin{bmatrix} x_6 \\ -k_1 x_5 - k_2 x_6 \\ x_8 \\ -k_3 x_{11} - k_4 x_{12} \end{bmatrix} = A_1 = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ -k_1 & -k_2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -k_3 & -k_4 \end{bmatrix} \begin{bmatrix} x_5 \\ x_6 \\ x_8 \\ x_{11} \\ x_{12} \end{bmatrix}$$

Les valeurs propres de cette matrices sont calculer par : $\det(A-\lambda I)=0$.

Alors :

$$\det \begin{bmatrix} -\lambda & -k_1 & 1 & 0 & 0 & 0 \\ 1 & -\lambda & 0 & -k_2 & 0 & 0 \\ 0 & 0 & -k_3 - \lambda & 0 & 0 & 0 \\ 0 & 0 & 0 & -k_4 - \lambda & 0 & 0 \\ 0 & 0 & 0 & 0 & -\lambda & -k_4 \\ 0 & 0 & 0 & 0 & 0 & -\lambda - k_3 \end{bmatrix} = 0$$

Pour garantir la stabilité il faudra choisir alors des valeurs positives pour ces paramètres. Ceci conduira à des valeurs propres négative

```

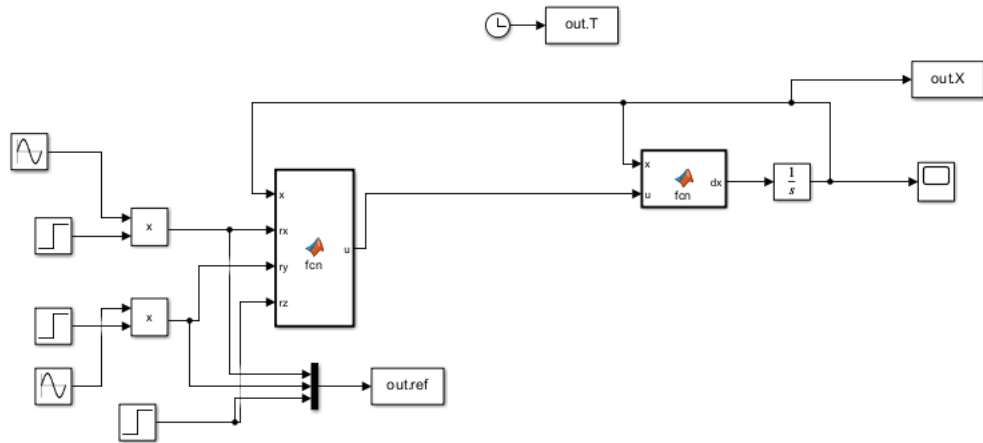
1 % Définition symbolique des variables
2 k1 =1;
3 k2 =1;
4 k3=1;
5 k4=10;
6
7 % Définition de la matrice A1
8 A1 = [0, 1, 0, 0, 0; -k1, -k2, 0, 0, 0; 0, 0, 0, 0, 0; 0, 0, 1, 0, 0; 0, 0, 0, -k3, -
9
10 % Calcul des valeurs propres
11 eigenvalues = eig(A1);
12 disp(eigenvalues)
13
14
```

Command Window

```

>> untitled6
-0.5000 + 0.8660i
-0.5000 - 0.8660i
-10.0000 + 0.0000i
0.0000 + 0.0000i
0.0000 + 0.0000i
```

2. Simuler le modèle vertical



```

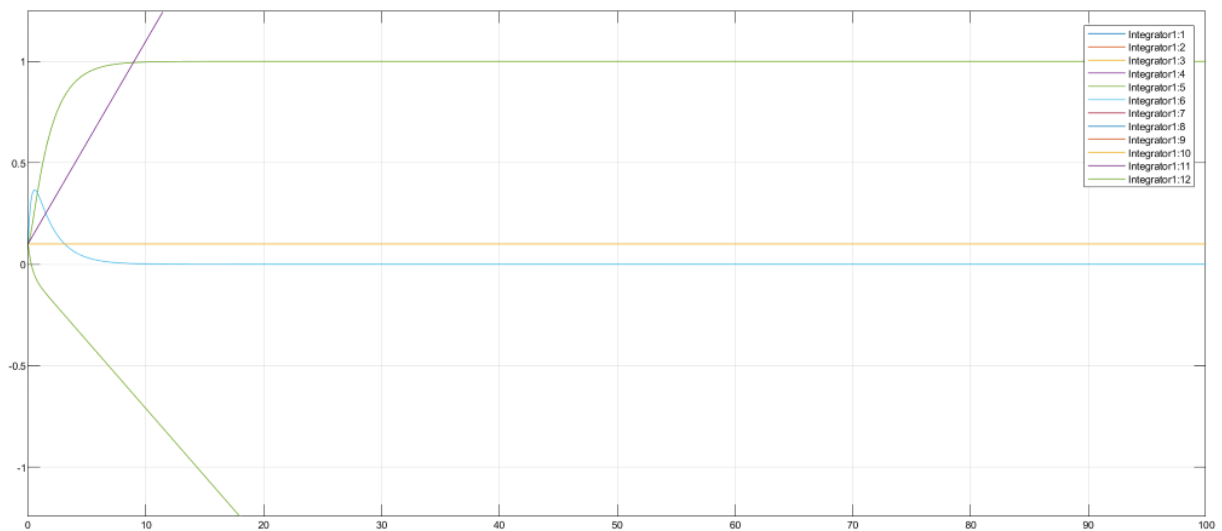
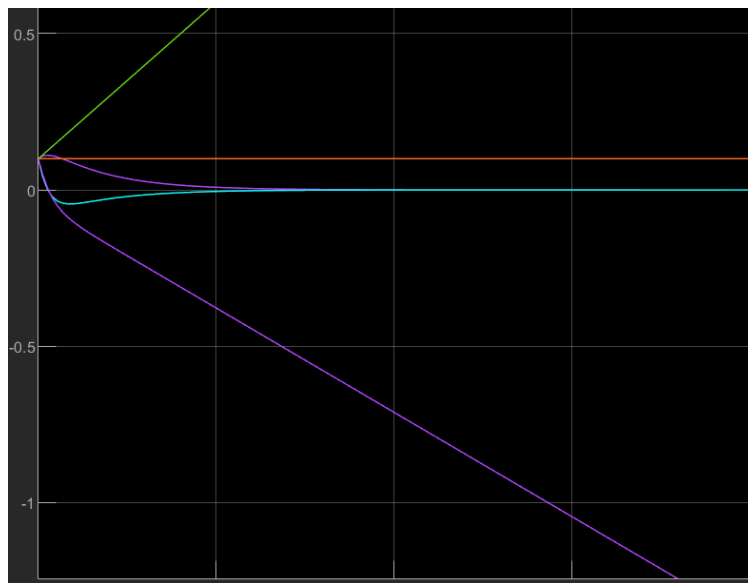
u(1) = 0;
u(2) = 0;
u(3) = -(x(10)*x(2)-x(8)*x(4)+g*cos(x(7))*cos(x(9)))-k5*(x(5)-rz)-k6*x(6);
u(4) = 0;
u(5) = 0;
u(6) = -a3*x(8)*x(10)-k11*x(11)-k12*x(12);

```

```

17      dx = zeros(12,1);
18
19      dx(1) = 0;
20      dx(2) = 0;
21      dx(3) = 0;
22      dx(4) = 0;
23      dx(5) = x(6);
24      dx(6) = x(10)*x(2)-x(8)*x(4)+g*cos(x(7))*cos(x(9))+u(3);
25      dx(7) = 0;
26      dx(8) = 0;
27      dx(9) = 0;
28      dx(10) = 0;
29      dx(11) = x(8);
30      dx(12) = a3*x(8)*x(10)+u(6)+u(4);
31

```

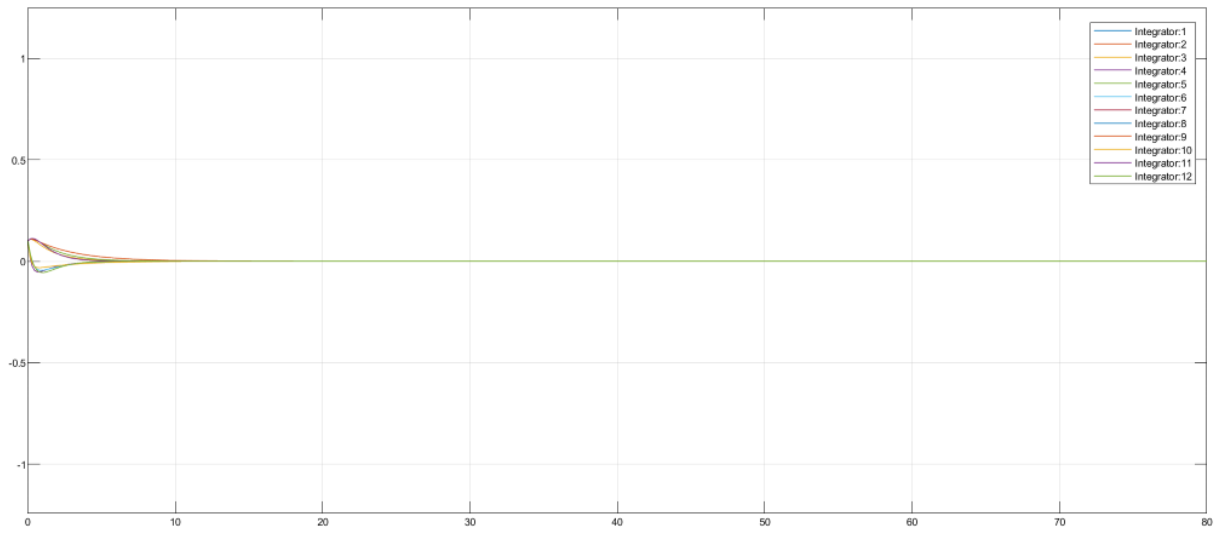


Commentaire : On peut bien remarquer que le système converge pour les deux cas avec des consignes nulles ou en constante

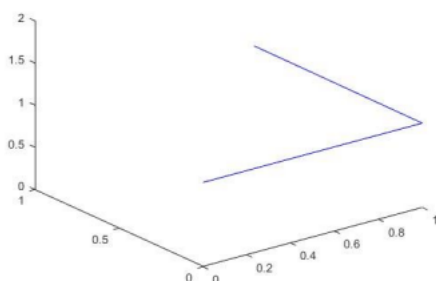
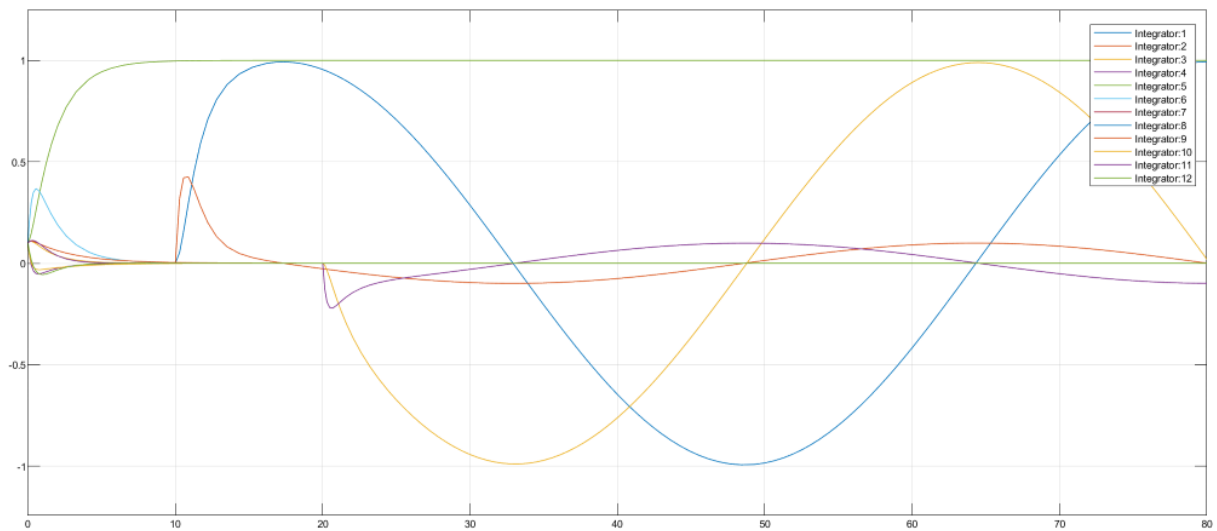
VI. Contrôle autonome

Simulations :

Pour consigne nulle



Pour consigne constante :



VII. Conclusion

La réalisation de ce TP a permis d'aborder plusieurs aspects clés de la conception de pilotes automatiques pour un drone en utilisant deux approches distinctes : l'approche LPV (Linear Parameter Varying) et l'approche de linéarisation exacte. Les principaux objectifs du TP étaient de se familiariser avec la modélisation LPV polytopique, les outils d'optimisation LMI (Linear Matrix Inequality), et l'analyse de stabilité au sens de Lyapunov.

Dans le cadre de l'approche LPV, nous avons exploré la modélisation du drone en tant que système dont les paramètres varient linéairement avec le temps. Cela a permis de concevoir un pilote automatique capable de s'adapter aux variations de paramètres du drone, améliorant ainsi sa robustesse et ses performances dans des conditions dynamiques changeantes.

En conclusion, ce TP a offert une expérience pratique enrichissante dans la conception de pilotes automatiques pour des systèmes dynamiques tels que les drones. Les compétences acquises en modélisation LPV, optimisation LMI et analyse de stabilité au sens de Lyapunov sont des atouts essentiels pour l'ingénierie des systèmes dynamiques complexes.