

# **MODÉLISATION ET COMMANDE MULTIVARIABLES**

**GEI-720**

*Jean de Lafontaine*

## **PROJET no 3**

### **Asservissement de la dynamique d'un aéronef avec la méthode de placement des pôles**

#### **Contenu**

1	Description du Projet .....	2
1.1	Objectif.....	2
1.2	Modèle de la dynamique et conditions initiales .....	2
1.3	Actionneurs et capteurs .....	2
1.4	PARTIE A : Design du régulateur .....	2
1.5	PARTIE B : Design de l'observateur .....	3
1.6	PARTIE C : Couplage de l'observateur et du régulateur .....	4

# 1 DESCRIPTION DU PROJET

## 1.1 OBJECTIF

Dans ce projet, le modèle non linéaire de la dynamique de translation et de rotation d'un aéronef sera asservi avec la méthode de placement des pôles, sujet à des critères de performance et utilisant un régulateur et un observateur d'états. Les outils MATLAB/Simulink seront utilisés pour obtenir les gains du régulateur et de l'observateur.

## 1.2 MODÈLE DE LA DYNAMIQUE ET CONDITIONS INITIALES

Le modèle Simulink de la dynamique non linéaire est dans le fichier **AVION\_LONGITUDINAL.mdl** du Projet no 2. Le modèle linéaire a été obtenu dans le Projet no 2 utilisant les fonctions **TRIM** et **LINMOD**. Le modèle est linéarisé à une vitesse de 80 m/s pour un vol horizontal et c'est ce modèle qui sera utilisé dans la conception.

Dans le Projet no 2, la dynamique de l'avion avait été excitée par une perturbation sur le gouvernail de profondeur. Dans le présent projet, la perturbation sur le gouvernail de profondeur sera enlevée et remplacée par une perturbation causée par des conditions initiales qui ne sont pas au point d'équilibre, i.e. une vitesse initiale de 85 m/s. Les autres variables d'état  $\alpha_e, \theta_e, q_e$  seront initialisées au point d'équilibre:

$$\text{etat\_ini} = [85, \alpha_e, \theta_e, q_e.]$$

## 1.3 ACTIONNEURS ET CAPTEURS

Pour la conception du régulateur, on ferme la boucle de commande sur l'actionneur du gouvernail de profondeur (angle  $\delta$ ). Ainsi, la matrice  $B$  du régulateur devient :

$$\underline{B}_\delta = \underline{B}(:, 1)$$

i.e. la première colonne de la matrice originale  $B$ . La propulsion reste inchangée à l'équilibre  $a_e$ .

Pour l'observateur, on suppose que l'unique sortie mesurée est la vitesse angulaire de tangage  $q$ , mesurées avec un gyromètre bruité. La déviation standard du bruit sera de :

$$\sigma = 1 \text{ mrad/s.}$$

On mesure en rad/s. La matrice de mesure est donc :

$$\underline{C}_{\text{mes}} = [0 \ 0 \ 0 \ 1].$$

## 1.4 PARTIE A : DESIGN DU RÉGULATEUR

Les critères de performance sont :

- temps de stabilisation à 2% de  $t_s \leq 8 \text{ s}$
- facteur d'amortissement de  $\sqrt{2}/2 = 0.707$ .

**P3-1 :** Calculer et justifier la position des pôles désirés du régulateur  $\underline{P}$ .

**P3-2 :** Déterminer la commandabilité du système avec l'actionneur  $\underline{B}_\delta$  (et la mesure  $\underline{C}_{\text{mes}}$  quand requise) en utilisant les 3 méthodes présentées dans les notes de cours.

**OPT :** Faire le design du régulateur (calcul de la matrice  $\underline{K}$ ) avec la méthode longue des notes de cours. (Voir algorithme à la page 5-18.) Vérifier que le résultat est identique à celui obtenu ci-dessous.

**P3-3 :** Calculer la matrice de gain  $\underline{K}$  du régulateur avec la fonction MATLAB **place**. Vérifier qu'on obtient bien  $\text{eig}(\underline{A} - \underline{B}\delta\underline{K}) = \underline{P}$ .

**P3-4 :** Valider le design du régulateur par simulation sur le modèle non linéaire **AVION\_LONGITUDINAL**. Dupliquer le modèle non linéaire dans Simulink. Asservir le premier avec le régulateur et mettre ses sorties de 1 à 16. Laisser la deuxième copie du modèle non linéaire comme tel, non asservi, et mettre ses sorties de 17 à 32. Il servira de comparaison par rapport au système régulé. Renommer le modèle ainsi obtenu et son fichier à **AVION\_CONTROL** pour éviter la confusion entre les deux modèles. IMPORTANT : Voir les notes de cours page 5-15 pour s'assurer que le régulateur ne reçoive que les  $\Delta\hat{x}$  et pour s'assurer de prendre en compte que la sortie du régulateur est  $\Delta u$ .

**P3-5 :** Comparer les sorties et vérifier la performance du régulateur. Remettre au minimum les sorties suivantes :

- vitesse (asservie et non asservie)
- vitesse angulaire de tangage (asservie et non asservie)
- Vérifier que le temps de stabilisation est bien de 8 s ou moins sur les graphiques.

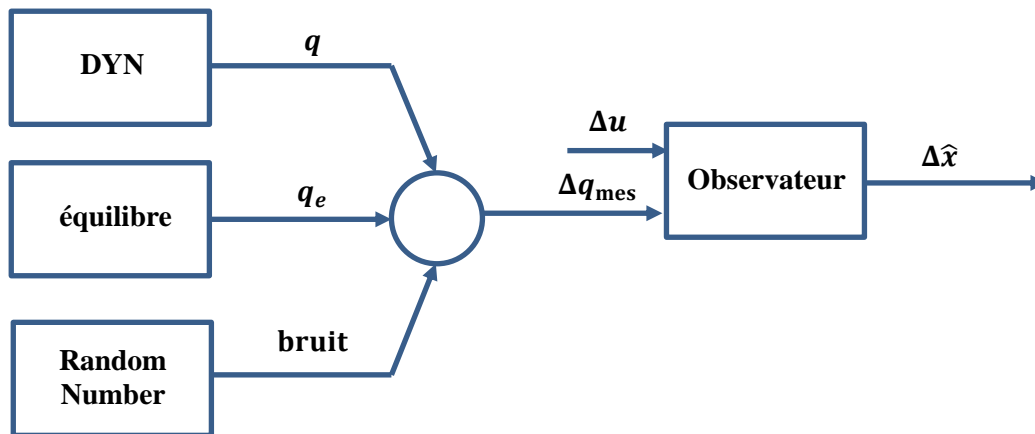
## 1.5 PARTIE B : DESIGN DE L'OBSERVATEUR

Dans la partie B, l'observateur d'état est ajouté au système asservi de **AVION\_CONTROL** pour créer un nouveau modèle appelé **AVION\_OBSERV** (le système non linéaire non asservi peut être supprimé). Dans ce modèle, le régulateur utilise toujours les vrais états du simulateur. L'observateur est ajouté en boucle ouverte pour valider sa performance, sans qu'il soit connecté au régulateur pour l'instant.

On doit créer et ajouter le modèle du capteur qui donnera les mesures  $\Delta y_{\text{mes}} = \Delta q_{\text{mes}}$  selon l'équation :

$$\Delta q_{\text{mes}} = q - q_e + \text{bruit}$$

où le bruit a une déviation standard de 1 mrad. Le modèle est donc :



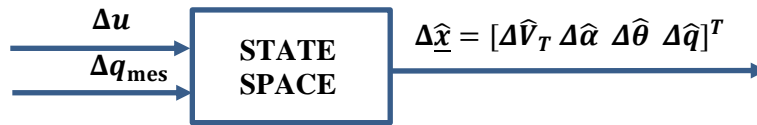
**P3-6 :** Donner et justifier le choix des pôles de l'observateur  $\underline{P}_e$ .

**P3-7 :** Déterminer l'observabilité du système avec

- la fonction MATLAB **obsv(A, C)**
- la fonction MATLAB **ctrb(?, ?)** utilisant la dualité de **obsv** et de **ctrb** (trouver ce que sont les ?).

**OPT :** Vérifier que le système est observable peu importe la variable d'état unique que l'on mesure, par exemple avec  $\underline{C}_{\text{mes}} = [1 \ 0 \ 0 \ 0]$ ,  $\underline{C}_{\text{mes}} = [0 \ 1 \ 0 \ 0]$  ou  $\underline{C}_{\text{mes}} = [0 \ 0 \ 1 \ 0]$ . Un seul capteur assure l'observabilité des 4 états.

- P3-8 : Faire le design de l'observateur (calcul de la matrice  $\underline{K}_e$ ) avec la fonction MATLAB [place](#). Vérifier qu'on obtient bien  $\text{eig}(\underline{A} - \underline{K}_e \underline{C}_{\text{mes}}) = \underline{P}_e$ .
- P3-9 : Valider le design de l'observateur par simulation sur le modèle non linéaire asservi. Comme annoncé plus haut, renommer le modèle et son fichier à [AVION\\_OBSERV](#). Pour réaliser l'observateur, utiliser un bloc « STATE SPACE » de Simulink avec les matrices  $A_{\text{obs}}, B_{\text{obs}}, C_{\text{obs}}, D_{\text{obs}}$  de l'observateur :



Les équations des matrices  $A_{\text{obs}}, B_{\text{obs}}, C_{\text{obs}}, D_{\text{obs}}$  de l'observateur sont obtenues à partir des équations du modèle de l'observateur aux pages 5-21 à 5-23 des notes de cours :

$A_{\text{obs}}$  : matrice d'état de l'observateur calculée avec la matrice  $\underline{K}_e$

$B_{\text{obs}}$  : noter dans le diagramme ci-dessus et dans les équations que l'observateur a 2 entrées,

- (1) l'entrée de commande (connue)  $\Delta u$
- (2) l'entrée de la mesure  $\Delta q_{\text{mes}}$

On doit donc construire la matrice  $B_{\text{obs}}$  à partir des équations de l'observateur (pages 5-21 à 5-23).

$C_{\text{obs}}$  : noter que les sorties de l'observateur sont les états en radians du système, donc attention que la sortie du système virtuel que constitue l'observateur est différente de la sortie du système non linéaire original et de la sortie unique du capteur utilisé à l'entrée de l'observateur.

Donner les équations analytiques des matrices  $A_{\text{obs}}, B_{\text{obs}}, C_{\text{obs}}, D_{\text{obs}}$  de l'observateur. Mettre les sorties  $\Delta \hat{V}_T \Delta \hat{\alpha} \Delta \hat{\theta} \Delta \hat{q}$  de 17 à 20 dans le simulateur Simulink.

Simuler avec Simulink pour démontrer que l'observateur reconstruit bien les 4 états. Ne pas oublier d'initialiser le système non linéaire aux conditions initiales  $\text{etat\_ini} = [85, \alpha_e, \theta_e, q_e]$  et d'initialiser l'estimateur d'état aux conditions à l'équilibre (donc conditions initiales nulles) parce que l'observateur d'état n'a aucune idée de l'état initial du système. On le suppose donc à l'équilibre.

Remettre au minimum les comparaisons d'états suivantes :

- (a) la vitesse :  $\hat{V}_T$  et  $V_T$
- (b) l'angle de tangage:  $\hat{\theta}$  et  $\theta$  (comparer des radians avec radians ou degrés avec degrés).

NOTE : Plus les pôles de l'observateur  $\underline{P}_e$  sont loin à gauche, plus l'observateur convergera rapidement vers les états mais plus le bruit du gyromètre sera amplifié. Il y a donc un compromis à faire entre la rapidité de convergence et l'amplification du bruit. Une autre façon de voir ce compromis est que, plus le gain de l'observateur est élevé, plus le modèle dynamique virtuel embarqué va « suivre » les mesures au dépend du modèle dynamique et il va donc aussi suivre le bruit dans ces mesures. Le compromis est de trouver le gain qui filtre le bruit de la mesure tout en conservant le contenu dynamique de la mesure. (Ce compromis est fait de façon optimale dans le Chapitre 6.)

## 1.6 PARTIE C : COUPLAGE DE L'OBSERVATEUR ET DU RÉGULATEUR

Dans la section précédente, le régulateur recevait toujours les états du système dynamique. Dans cette dernière partie, le régulateur utilisera les variables d'états estimées par l'observateur.

- P3-10 : Faire une copie de [AVION\\_CONTROL](#) sous le nom [AVION\\_OBSCTL](#) et y ajouter l'observateur validé dans [AVION\\_OBSERV](#). Relier ensuite les sorties de l'observateur à l'entrée du régulateur. Valider par simulation l'asservissement complet du système. Remettre les graphiques des variables asservies  $V_T$ ,  $\theta$  et  $q$  et de leur valeur estimée par l'observateur, chaque paires sur le même graphique.