

**TP Matlab-Simulink**  
**Robotique Mobile : Partie Modélisation/Commande**  
**Master SAR - M2**

**Intitulé:** *Estimation d'attitude pour les drones multirotors*

L'objectif de ce TP est de mettre en œuvre sous Matlab-Simulink des algorithmes d'estimation d'attitude pour des drones multirotors (voir slides de cours, partie "estimation\_drones"). L'exemple considéré dans ce TP est un drone à quatre hélices (quadricoptère). Le schéma global du simulateur est donné par la Figure ci-dessous. Il s'agit d'un simulateur assez complet (mais cependant simplifié).

## 1 Architecture globale du simulateur:

Le simulateur est composé de plusieurs sous-systèmes principaux:

- Le sous-système bleu "Dynamique" intègre les équations de la dynamique du drone.
- Le sous-système rouge "Capteur/Estimateur" est destiné à l'estimation d'état. **Vous allez dans ce TP principalement travailler sur ce bloc**; nous le détaillerons plus bas.
- Le sous-système vert "Contrôleur" contient la partie calcul des lois de commande. Les commandes utilisées dans ce TP sont des commandes avec consignes en vitesses linéaires et vitesse angulaire autour de l'axe de lacet.
- Le sous-système bleu "Visualisation" est destiné à la visualisation des variables et à l'enregistrement des données. Par soucis de lisibilité, certains graphes ont été désactivés.
- Le sous-système jaune "Vitesse-Vent" permet de générer du vent. Par défaut il génère un vent de vitesse nulle et il n'est pas nécessaire de le modifier dans ce TP.
- Le sous-système vert "Vitesse-Ref" permet de générer une consigne de vitesse de référence linéaire  $v_r$ , ainsi qu'une consigne de vitesse angulaire autour de l'axe de lacet du drone  $\omega_{r,3}$ . En double cliquant sur ce bloc, vous pouvez choisir entre deux types de consignes: une consigne de vol quasi-stationnaire (très faibles vitesses de référence), et une consigne de vol non-stationnaire, avec des vitesses de référence plus importantes.

Le contrôleur implémenté est un contrôleur de vitesse: il a pour but de stabiliser la vitesse linéaire  $v$  du drone à la vitesse de référence  $v_r$  et la troisième composante  $\omega_3$  du vecteur vitesse angulaire  $\omega$  du drone à la vitesse angulaire de référence  $\omega_{r,3}$ . Vous n'aurez pas à intervenir sur ces lois de commande.

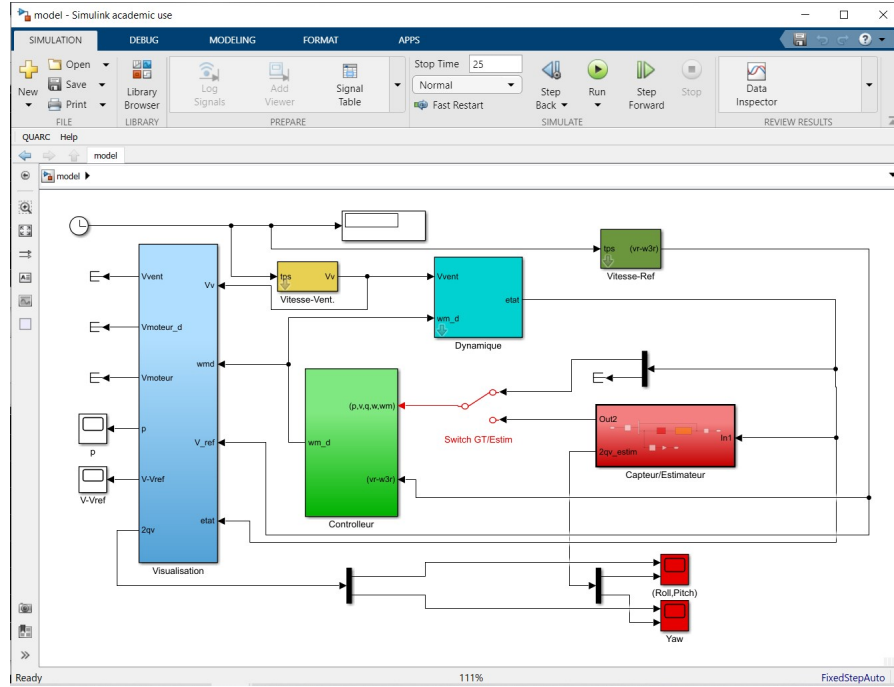


Figure 1: Architecture globale du simulateur

Afin de calculer les lois de commande, il faut évidemment alimenter le contrôleur avec les mesures/estimation de l'état. Le bloc "Switch GT/Estim" (voir figure) permet de switcher entre deux modes: alimentation par la "Ground Truth" (GT), i.e., l'état exact du drone, ou alimentation par la sortie d'un estimateur, i.e. le sous-système "Capteur/Estimateur". Nous allons maintenant détailler ce sous-système.

## 2 Architecture du sous-système "Capteur/Estimateur":

Ce sous-système est illustré par la figure 2 ci-dessous.



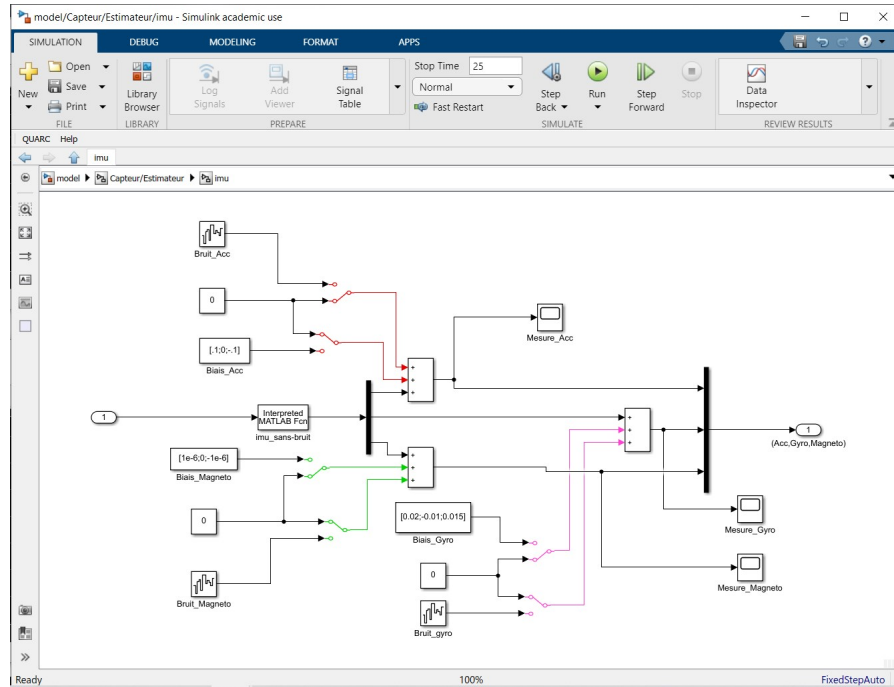


Figure 3: Sous-système "imu"

La sortie du sous-système "imu" est fournie au sous-système "Quaternion\_estim" qui doit renvoyer une estimation d'attitude sous la forme d'un quaternion unitaire (i.e., vecteur de norme 1). Cette estimation sera ensuite agrégée aux "mesures parfaites des autres états pour fournir l'estimée finale. Autrement dit, **dans cette version du simulateur, on ne s'intéresse qu'à l'estimation d'attitude**, i.e., pour les autres états on récupère les valeurs GT.

Une capture d'écran du sous-système "Quaternion\_estim" est fournie ci-dessous.

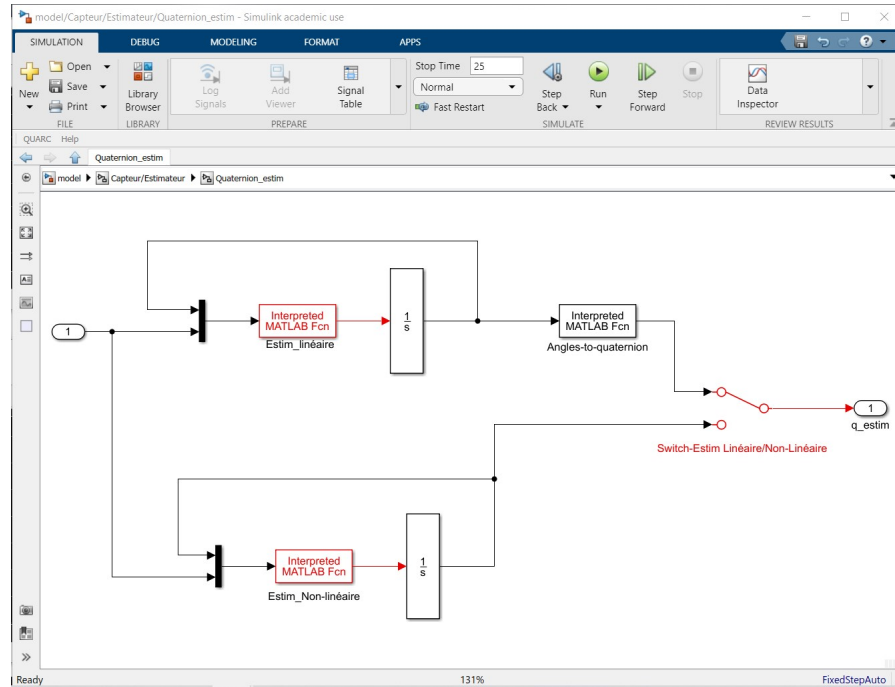


Figure 4: Sous-système "imu"

On peut identifier essentiellement deux parties, qui feront l'objet des deux parties principales de ce TP:

1. Une première partie, avec le bloc "Estim\_linéaire", a pour but de développer une estimation de l'attitude à partir d'un estimateur linéaire. Pour cela, on commence par estimer les angles de Roulis/Tangage/Lacet, puis ces angles sont passés à un bloc "Angles-to-quaternion" qui convertit ces angles en un quaternion unitaire.
2. Une deuxième partie, avec le bloc "Estim\_Non-linéaire", a pour but de développer une estimation de l'attitude à partir d'un estimateur non-linéaire qui va directement fournir un quaternion unitaire.

Le choix de l'estimateur se fait par le biais du switch manuel rouge de la figure.

### 3 Synthèse d'un estimateur linéaire

L'objectif est de fournir un quaternion unitaire à partir des estimateurs d'angles linéaires vus en cours.

1. Mettre en œuvre un estimateur (observateur) linéaire pour l'estimation des angles  $\theta_1, \theta_2, \theta_3$  correspondant aux angles de Roulis, Tangage, Lacet (voir slides de cours). Pour cela, on complètera le fichier "estim\_lin" qui doit définir la dynamique des angles estimés

$\hat{\theta}_1$ ,  $\hat{\theta}_2$ , et  $\hat{\theta}_3$ . On reprendra les éléments vus en cours. Concernant les mesures magnétométriques, on suppose que le capteur mesure le champ magnétique terrestre, dont les coordonnées par rapport au repère inertiel du drone sont données approximativement (valeurs du champ magnétique terrestre à Paris) par

$$m_I = 4.7e - 5 \begin{pmatrix} \cos(65^\circ) \\ 0 \\ \sin(65^\circ) \end{pmatrix}$$

2. Valider dans un premier temps l'estimateur en l'absence de biais de mesure et de bruits de mesure, pour le vol quasi-stationnaire. On profitera de cette validation préliminaire pour tester différents choix de gains de l'estimateur. **Remarque: A ce stade, ne pas encore utiliser l'estimation dans la commande, i.e., garder la GT pour alimenter le contrôleur.**
3. Affiner le choix des gains de l'observateur en ajoutant aux mesures imu des bruits de mesure, et en réglant les gains de façon à avoir un niveau de bruit raisonnable sur les estimées. Analyser l'impact de ces bruits sur les estimées et sur l'asservissement du drone.
4. Utiliser le switch du système principal (Figure 1) pour passer en mode "contrôleur alimenté par l'estimation d'attitude". Interpréter et commenter les résultats.
5. En double-cliquant sur le bloc vert, passer en vol non-stationnaire, et relancer les simulations, dans un premier temps en alimentant le contrôleur par la "Ground Truth" (GT), puis en alimentant le contrôleur par les estimées. Interpréter et commenter les résultats.

## 4 Synthèse d'un estimateur non-linéaire

Les résultats de la section précédente montrent la difficulté d'obtenir une bonne estimation du lacet avec l'approche linéaire. On se propose donc ici de mettre en œuvre un estimateur non-linéaire. Plusieurs solutions ont été vues en cours, et vous êtes libres d'utiliser l'estimateur de votre choix. Dans un premier temps, l'utilisation du filtre complémentaire explicite ("Explicit complementary filter") est recommandée.

On rappelle que l'estimateur doit fournir un quaternion unitaire. On rappelle également (Voir Slides 18-19 de la partie "Commande" consacrée aux drones) qu'une dynamique de rotation

$$\dot{R} = RS(\omega)$$

est équivalente, en quaternions, à la dynamique suivante:

$$\begin{cases} \dot{q}_s &= -\frac{1}{2}\langle q_v, \omega \rangle \\ \dot{q}_v &= \frac{1}{2}(q_s \omega + q_v \times \omega) \end{cases}$$

Par ailleurs, pour éviter les dérives numériques de la norme du quaternion (qui doit rester unitaire), on peut modifier la dynamique ci-dessus de la façon suivante:

$$\begin{cases} \dot{q}_s &= -\frac{1}{2} (\langle q_v, \omega \rangle - q_s(1 - \|q\|^2)) \\ \dot{q}_v &= \frac{1}{2} (q_s \omega + q_v \times \omega + q_v(1 - \|q\|^2)) \end{cases}$$

1. Mettre en œuvre un estimateur non-linéaire avec des mesures imu parfaites, et tester l'estimateur sur les vols quasi-stationnaire et non-stationnaire, dans un premier temps en alimentant le contrôleur avec la Ground-Truth, puis avec les estimées. Interpréter et commenter les résultats. On portera un soin particulier au réglage des gains de l'observateur, en répondant aux questions suivantes: que se passe t-il si l'on augmente trop ces gains? quel ordre de grandeur des gains ne doit-on pas dépasser?
2. Affiner le choix des gains de l'estimateur en ajoutant aux mesures imu des bruits de mesure, et en réglant les gains de façon à avoir un niveau de bruit raisonnable sur les estimées. Analyser l'impact de ces bruits sur l'estimation d'attitude et sur l'asservissement du drone.
3. Ajouter maintenant les biais de mesure, et analyser l'impact de ces biais sur l'estimation d'attitude et sur l'asservissement du drone. Dans un premier temps on ajoutera un seul biais à la fois; puis on regardera l'impact des trois biais réunis.
4. Selon vous, comment pourrait-on réduire l'impact de ces biais?