

# TP expérimentaux

## Commande Robuste $H_\infty$ d'un Hélicoptère 3DDL

Raphael Martens

Antoine Morin

Nicolas Rechatin

Mathieu Sousa

## TABLE DES MATIERES

<b>I. INTRODUCTION</b>	<b>2</b>
<b>II. METHODE PAR FONCTIONS DE PONDERATION</b>	<b>2</b>
1. VALIDATION DE L'EXEMPLE DE COURS	2
2. DETERMINATION DU CORRECTEUR PAR LA METHODE DES PONDERATIONS	4
A) LES FONCTIONS DE PONDERATION	4
B) ETUDE DES COMMANDES	6
C) RECHERCHE DES FONCTIONS DE PONDERATION	6
<b>III. CONCLUSION</b>	<b>10</b>

## I. Introduction

L'objectif de ce TP est de mettre en pratique les connaissances acquises lors des cours de commande robuste. L'étude permettra de mettre en évidence les bénéfices et les limites associés à une commande  $H_\infty$ .

## II. Méthode par fonctions de pondération

### 1. Validation de l'exemple de cours

Dans un premier temps nous allons nous intéresser à la validation de l'exemple de cours. Pour cela nous programmons le script Matlab de la synthèse d'un correcteur  $H_\infty$  avec les propriétés suivantes :

$K_{\text{hinf}} = \text{hinfsyn}(P, 1, 1, 0.5, 30, 0.05)$

Une entrée

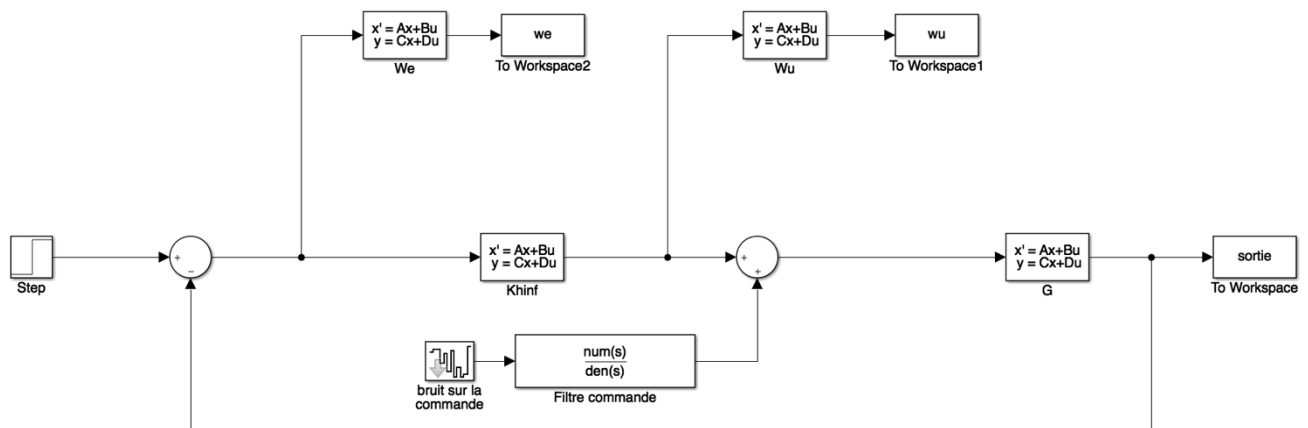
Une sortie

$\gamma_{\min} = 0.5$

$\gamma_{\max} = 30$

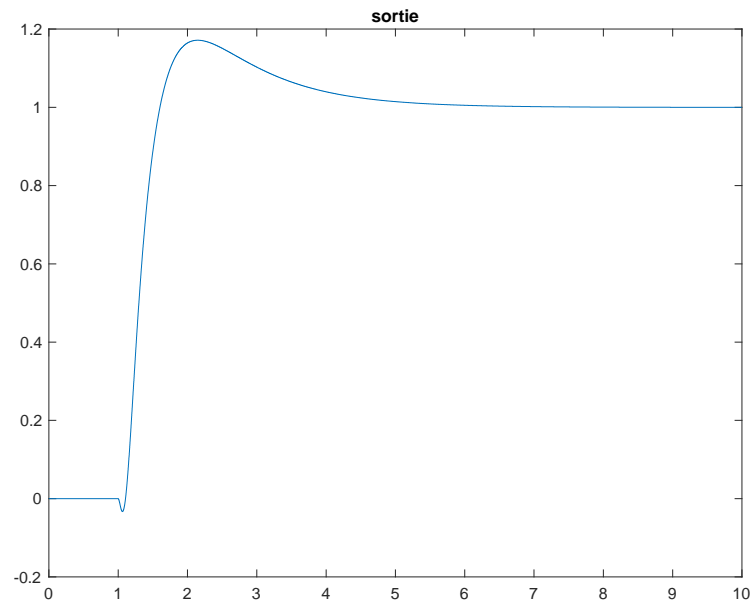
pas = 0.05

La valeur de Gamma obtenu est 1.6653. Le correcteur ainsi synthétisé est ensuite testé de la façon suivante :

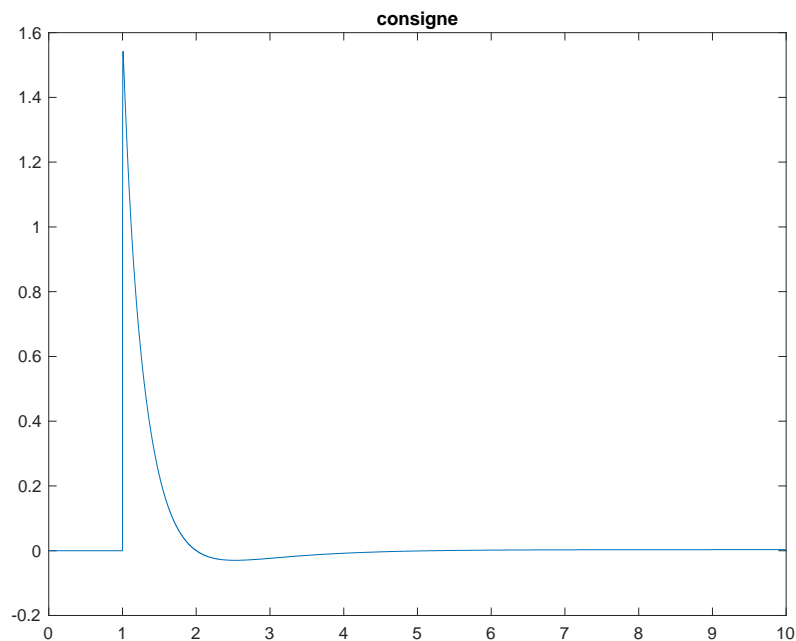


Afin de tester la robustesse de notre commande, nous avons rajouté un bruit sur la commande.

Les résultats obtenus pour un échelon unitaire sont présentés ci-dessous.



Nous observons un bon suivi de la consigne avec un dépassement de 17%.



La consigne obtenue ne présente aucun bruit ce qui est l'intérêt de l'utilisation d'un correcteur  $H^\infty$ .

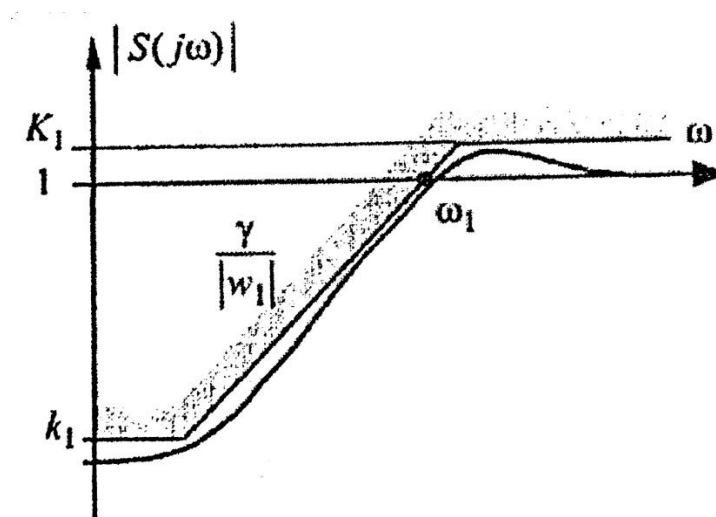
## 2. Détermination du correcteur par la méthode des pondérations

### a) Les Fonctions de pondération

Dans cette deuxième partie le correcteur  $H_\infty$  sera déterminé par l'intermédiaire de la méthode des pondérations. Les trois filtres pondérateurs à régler seront  $W_1(p)$ ,  $W_2(p)$ ,  $W_3(p)$ .

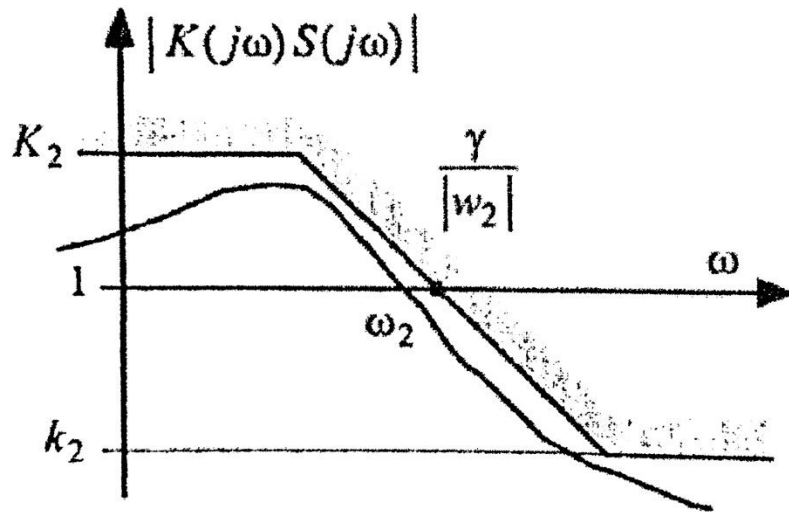
Ces trois filtres pondérateurs doivent avoir les profils suivants :

- $W_1$  :



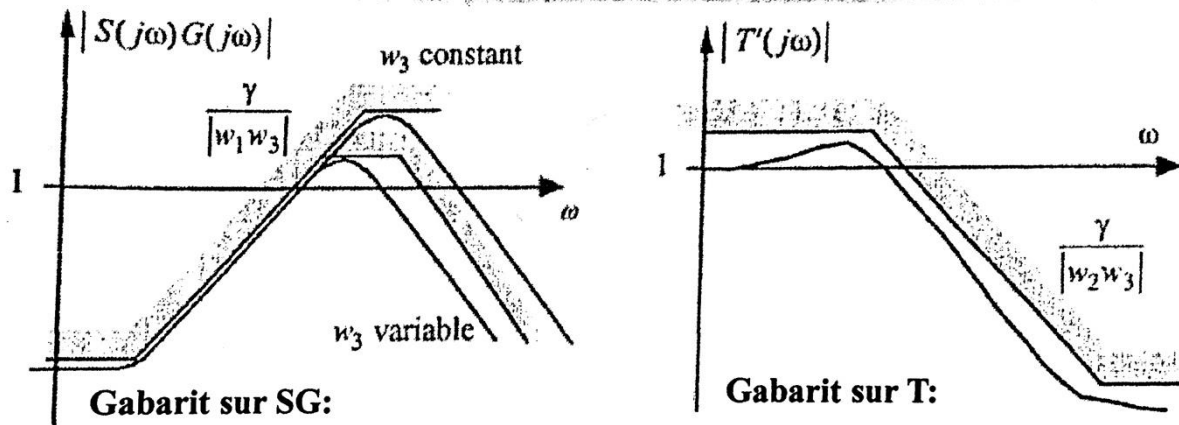
Le rôle du premier pondérateur agissant sur  $S$  est d'assurer la performance de suivi de trajectoire.

- W2 :



Le rôle du deuxième pondérateur agissant sur KS est d'assurer la robustesse et la limitation de la commande.

- W3 :

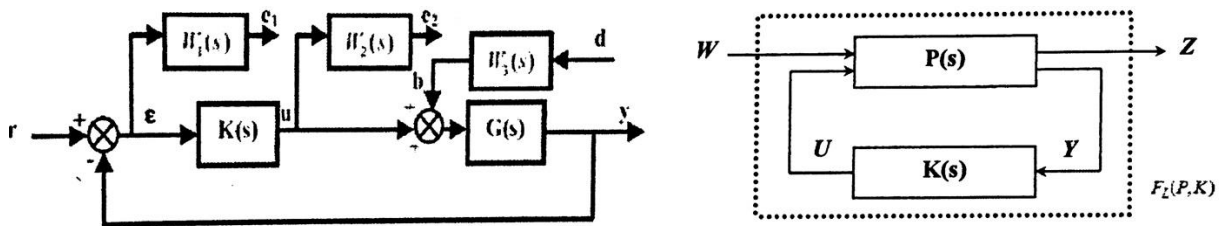


Enfin le rôle du troisième pondérateur agissant sur SG a un rôle de rejet de perturbation.

## b) Etude des commandes

### • sysic

La commande sysic permet de former un groupe de différents systèmes. Cette commande permet donc la mise sous forme d'état augmentée.



### • hinfsyn

La commande hinfsyn permet la synthèse d'un correcteur Hinf avec les options comme suit :

$K_{hinf} = \text{hinfsyn}(P, 1, 1, 0.5, 30, 0.05)$

Une entrée

Une sortie

$\gamma_{min} = 0.5$

$\gamma_{max} = 30$

pas = 0.05

### • nd2sys

La commande nd2sys permet de passer d'une fonction de transfert à un système avec la possibilité de rajouter un gain sur la sortie du système.

## c) Recherche des fonctions de pondération

i)

Dans un premier temps nous cherchons à déterminer les paramètres de la fonction  $W1$ .

Nous savons que la fonction  $W1(s)$  se définit de la façon suivante :

$$\frac{1}{W_e(s)} = \frac{s + w_b \varepsilon}{\frac{s}{M_s} + w_b}$$

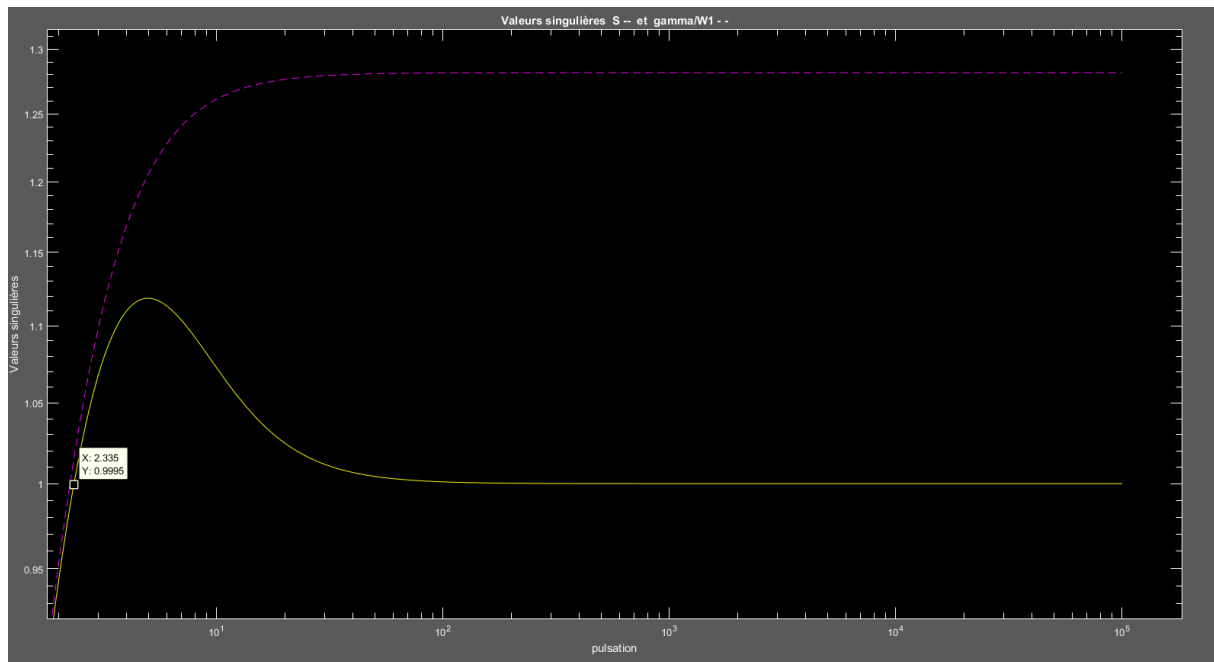
De plus nous savons que le gain  $M_s$  ne doit être inférieur à 2.0. A l'aide de la courbe la courbe traçant la valeur singulière de  $S$  et  $\gamma/W_1$  obtenue précédemment nous indique que l'on doit poser un epsilon très faible et la fréquence de coupure  $W_b$  proche de 1.

Après plusieurs essais, nous avons retenus les paramètres suivants :

```
Ms = 1.5;
Wb = 1.2;
epsilon = 1e-6;

DenW1 = [1 Wb*epsilon];
NumW1 = [1/Ms Wb];
```

Nous obtenons la réponse et les caractéristiques suivantes :



La pseudo bande passante est de 2.335 rad/s.

Nous calculons l'erreur statique

$$20 \log(0.29) = -24.75 \text{ dB}$$

Enfin nous vérifions la condition sur la marge de module.

$$\frac{1}{\max[S(jw)]} = \frac{1}{1.12} = 0.89$$

Les contraintes des deux autres gabarits sont relâchées en les forçant à des valeurs constantes ( $W_3$  proche de 0 et  $W_2$  proche de 1).



ii)

Pour atténuer le bruit en basse fréquence un filtre  $W(3)$  constant est suffisant.

Afin de rapprocher la valeur de  $\gamma$  vers une valeur proche de 1 nous avons procédé à plusieurs essais et nous avons retenu les résultats suivants :

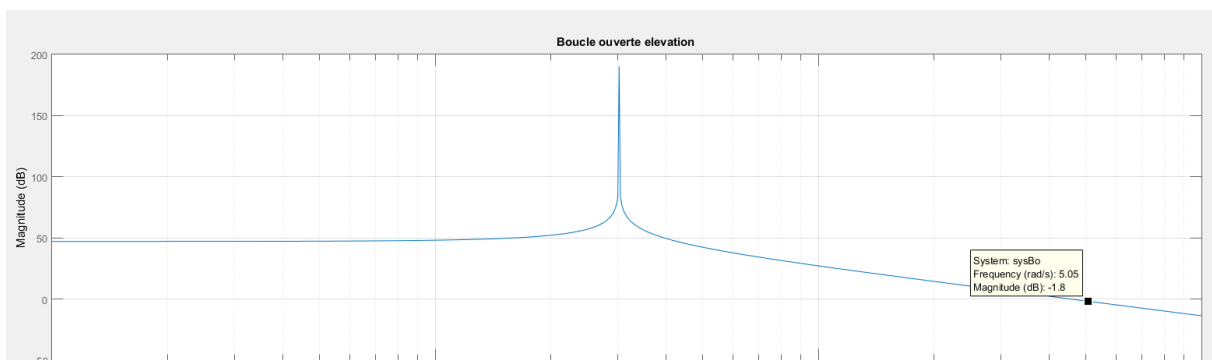
```
NumW3 = 0.05;  
DenW3 = 1;
```

Pour la valeur de  $\gamma$  suivante :

```
Gamma value achieved: 1.0978
```

iii)

Le graphique ci-dessous représente le diagramme de Bode du système en boucle ouverte.



On observe que la pseudo bande passante du système est de 5 rad/s. De plus la courbe traçant la valeur singulière de  $S$  et  $\gamma/W^2$  nous indique que le  $\mu$  doit être fixé à 1.

Nous choisissons une fréquence similaire pour notre filtre  $W_2$ .  
De plus nous prenons un gain unitaire pour le filtre.

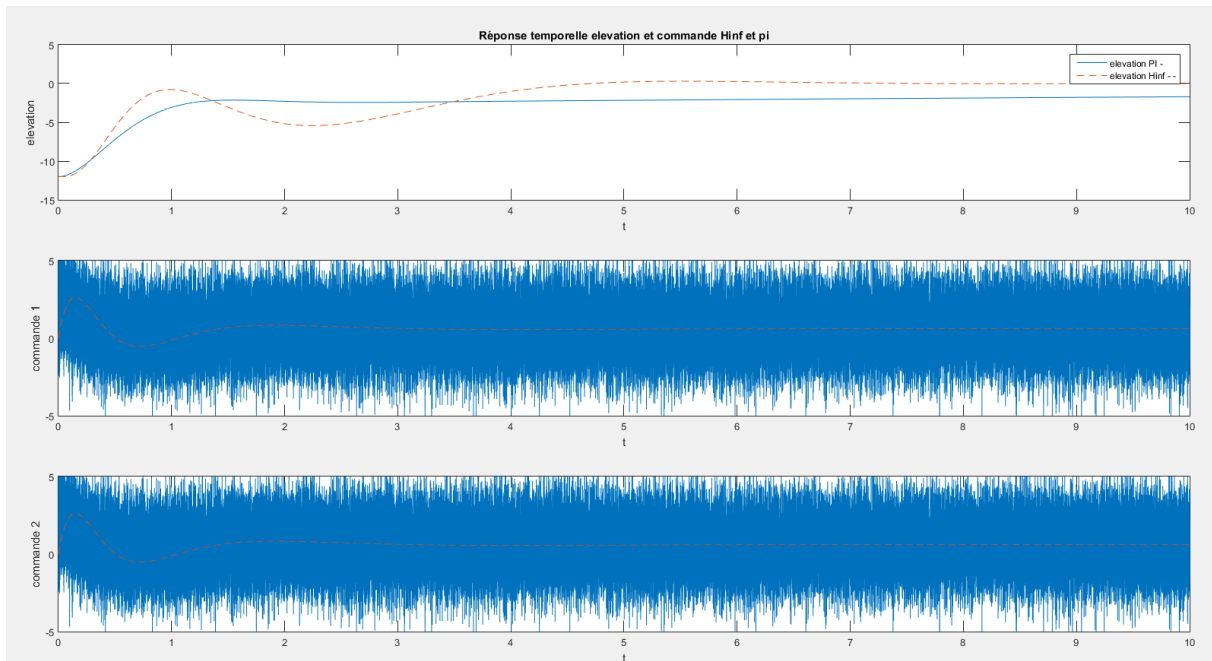
Nous obtenons finalement les paramètres suivants :

```
epsilon1 = 1e-6;  
Wbc= 5;  
Mu = 1;
```

Après avoir fixé  $W_2$  nous réajustons  $W_3$  pour obtenir une réponse temporelle

satisfaisante.

Grace à un gain égal à 0.1 nous obtenons la réponse temporelle suivante :



On observe que la réponse converge en environ 5 secondes contre plus de 30 secondes pour le PI.

De plus la commande n'est plus bruitée ni saturé contrairement à la commande PI.

### III. Conclusion

Cette étude nous a donc permis, via une application directe de notre connaissance, de mieux comprendre les enjeux de la commande  $H^\infty$ . En effet suite à la comparaison et la synthèse des commande PI et  $H^\infty$  nous avons pu observer les avantages et les inconvénients de la commande  $H^\infty$ .

La commande  $H^\infty$  est comme le démontre les derniers résultat plus rapide et plus robuste que la commande PI. La commande  $H^\infty$  semble donc être un bon choix lorsque l'on veut maximiser les performances de la commande ainsi que sa robustesse.

Cependant cette étude nous a également mis en évidence la complexité de la synthèse de la commande  $H^\infty$  tant dans son optimisation que dans le calcul. On notera dans un premier temps que la commande  $H^\infty$  n'est pas utilisable pour tous les types de système ce qui réduit déjà son utilisation. De plus le principal inconvénient de la commande  $H^\infty$  est, au-delà de la complexité dans la synthèse, les lourds calculs quelle requiert. En effet lors de notre étude, nous nous sommes aperçu que les calculs que requiert par la commande  $H^\infty$  sont très grand et nous avons eu des problèmes de gestion de mémoire interne sur les ordinateurs à de nombreuses reprises.

Ainsi nous pouvons conclure que la commande  $H^\infty$  est une commande qui permet d'assurer de très bonne performance, mais qui nécessite de grandes ressources matérielles pour être efficace. Dans le cas des systèmes embarqués, la mémoire interne étant souvent limitante, on pourra se tourner vers d'autres technologies de commande tel que la  $H^\infty$  Loop-Shaping qui permet des performances acceptables mais reste bien plus rapide et simple que la  $H^\infty$  classique à réaliser.