

## Petite introduction à l'utilisation de Matlab/Simulink pour l'Automatique

Ces quelques pages sont consacrées à l'illustration des cours d'automatique (asservissements linéaires continus) à l'aide du logiciel Matlab, et de ses extensions Simulink et Control System Toolbox, dédiées particulièrement à l'automatique.

Ces logiciels sont commercialisés par la société américaine Mathworks. Sans vouloir leur faire une publicité excessive, il faut reconnaître que leur produit est très complet et très agréable à utiliser, ce n'est pas étonnant s'il est une référence dans l'industrie et dans l'enseignement (par exemple, les candidats à l'agrégation de mathématiques utilisent uniquement les logiciels Maple et Matlab pour l'épreuve de modélisation). Le prix de vente aux établissements d'enseignement s'est beaucoup amélioré. Une version étudiante existe à un peu plus de 100 euros TTC, elle est assez difficile à trouver, l'éditeur de cette version est Prentice-Hall.

Un autre logiciel, assez proche de Matlab dans l'esprit et la forme, est disponible gratuitement, il s'agit de Scilab (prononcer ssail-lab!) acronyme de scientific laboratory. Ce logiciel a été élaboré par l'INRIA. Une extension très proche de simulink est disponible et s'appelle scicos. Ce logiciel est téléchargeable à :

<http://www-rocq.inria.fr/scilab/>

Ce type de logiciel constitue une aide puissante pour le professeur faisant son cours d'automatique. Bien sûr, il est possible d'illustrer un cours à l'aide de transparents qui permettent de projeter une courbe qui est la réponse du système étudié à telle excitation, un diagramme de Nichols ou autre. Mais tout cela est désespérément figé.

Supposons une question du genre "Monsieur, si ce paramètre-là augmente, que se passe-t-il?", évidemment le professeur répond, avec ou sans Matlab. Il redessinerait avec plus ou moins de bonheur la courbe modifiée, il n'en pourra donner qu'une vague allure générale à moins de passer tout le temps du cours à calculer les coordonnées des points afin de la dessiner exactement. Un logiciel permet de projeter la courbe voulue exactement et en un clin d'œil. Halte au vague, à l'approximation !

Avant de faire dessiner la courbe au logiciel, on peut toujours, prévoir, estimer son allure, c'est un exercice indispensable. Mais ensuite: précision et gain de temps !

Libéré des contraintes des calculs, il est possible de consacrer plus de temps à l'essence des choses, à l'Automatique elle-même. Fait-on vraiment de l'automatique lorsque l'on a les mains plongées dans le "cambouis du calcul" des pôles d'une fonction de transfert en vue d'obtenir sa transformée de Laplace, lorsque l'on calcule 20 points afin de tracer un lieu de Nichols ? On réalise des opérations nécessaires à l'automatique mais ce n'est qu'accessoire, ce n'est pas de l'Automatique.

Le temps consacré à cette discipline austère est compté. Dans nos ENMM<sup>1</sup> où il est admis que le but n'est pas de former des ingénieurs de conception, champions du calcul mathématique, je pense que l'utilisation raisonnable et bien pensée de tels logiciels permettrait une meilleure compréhension de l'essence de l'automatique en consacrant moins de temps à l'accessoire (le calcul) et plus à l'essentiel : la réflexion sur les principes, les phénomènes.

Voici l'illustration, à l'aide du logiciel Matlab, de ces nouvelles possibilités offertes aux professeurs, il est conseillé de lire activement en refaisant sur un ordinateur équipé de Matlab les manipulations décrites ci-après.

---

<sup>1</sup> ENMM : école nationale de la marine marchande

## **Un soupçon de MATLAB :**

Afin de maîtriser le logiciel "Matlab", nous allons d'abord étudier un système du second ordre :  $H(p) = \frac{10}{s^2 + 2.s + 10}$  (en notation anglo-saxonne, la variable complexe p est remplacée par s).

Les instructions MATLAB sont écrites en gras. Pour avoir des détails, utiliser la commande : "help instruction" (par exemple taper "help nichols").

Ecriture de la fonction de transfert :

Taper les instructions suivantes dans le "workspace" de Matlab

```
num=10  
den=[1 2 10]  
printsys(num,den)
```

Etude de la réponse à un échelon :

```
step(num,den)
```

pour voir la réponse à un échelon pendant 10s

```
t=0:0.1:10 ;  
y=step(num,den,t); le point virgule ";" évite d'afficher le résultat  
plot(t,y)
```

commentaires sur la courbe :

```
title('réponse à un échelon');  
xlabel('temps');  
ylabel('y');
```

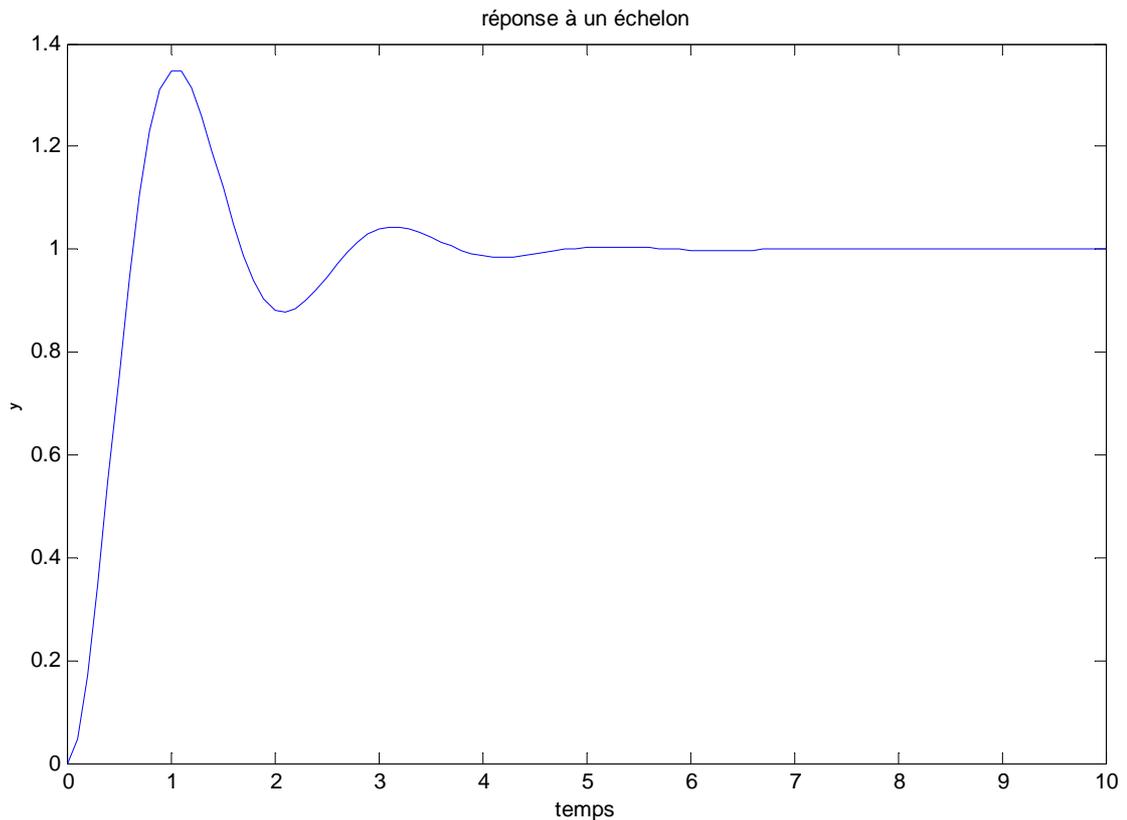


Figure 1 - Réponse du système H à un échelon unité

Pour lire des valeurs sur la courbe :

**[x,y]=ginput(3)** et cliquer avec la souris sur 3 points à mesurer (pas de point virgule sinon commande inopérante ici)

Pour tracer un quadrillage ou une ligne :

**grid**

**line([0 10],[1.05 1.05])**

**line([0 10],[1.05 1.05] , 'color','g')**

*ces instructions permettent de tracer deux lignes horizontales et de faire apparaître le "tuyau des réponses à 5%" :*

**line([0 10],[1.05 1.05])**

**line([0 10],[0.95 0.95])**

*la commande  $line([x1,x2],[y1,y2])$  trace une ligne du point  $(x1,y1)$  au point  $(y1,y2)$ .*

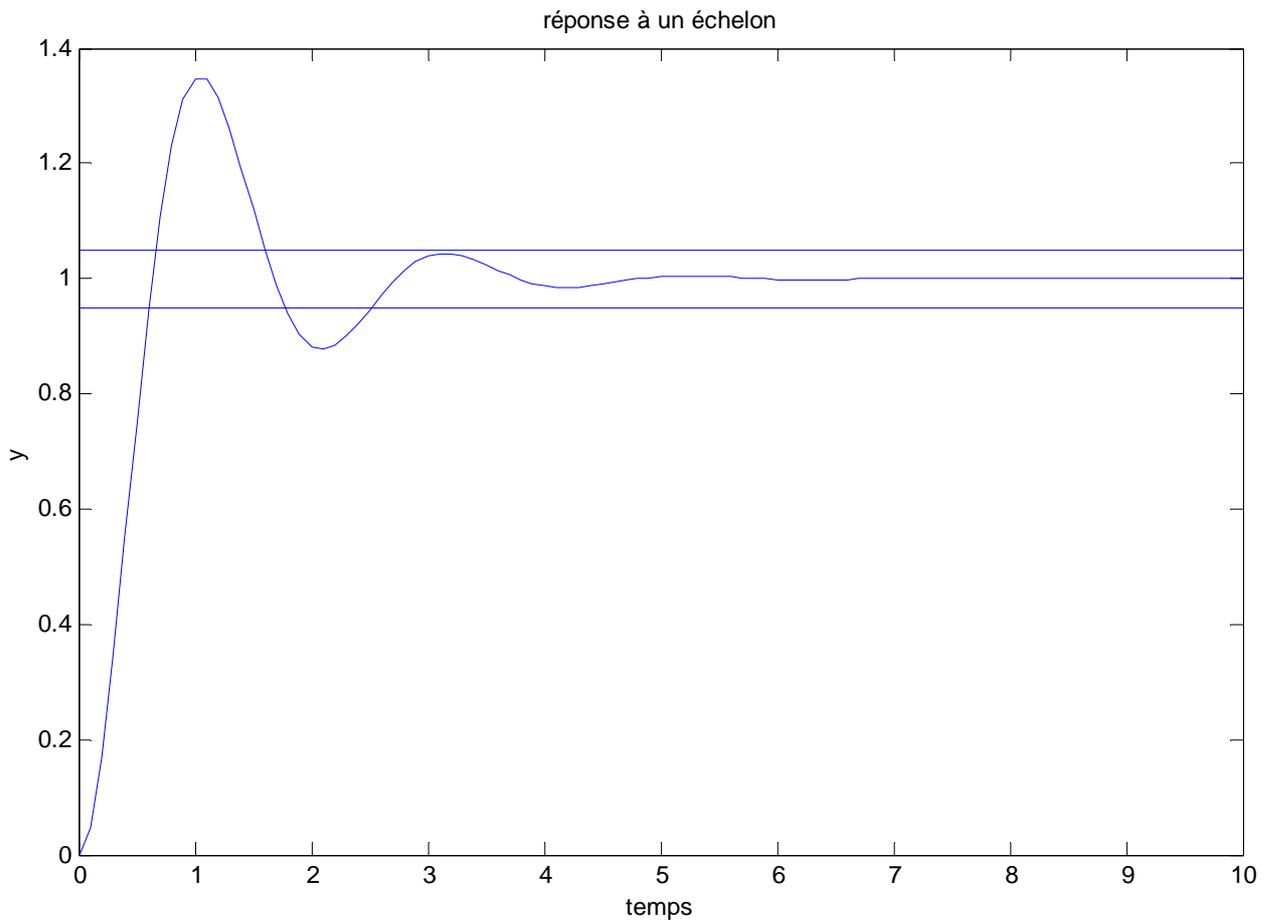


Figure 2 - "Tuyaux de la réponse à 5%" sur la réponse à un échelon

On peut mesurer les coordonnées de points intéressants avec l'instruction *ginput* :

- le temps de réponse à 5%
- le dépassement D%
- le temps de montée au premier pic tpic
- la pseudo période.

### 3) réponse à un signal quelconque

Il faut d'abord définir le temps :

**t=0:0.1:10;**     *la variable t prend des valeurs de 0 à 10 par pas de 0,1*

une rampe :

```
ramp=t;  
y=lsim(num,den,ramp,t);  
plot(t,y)
```

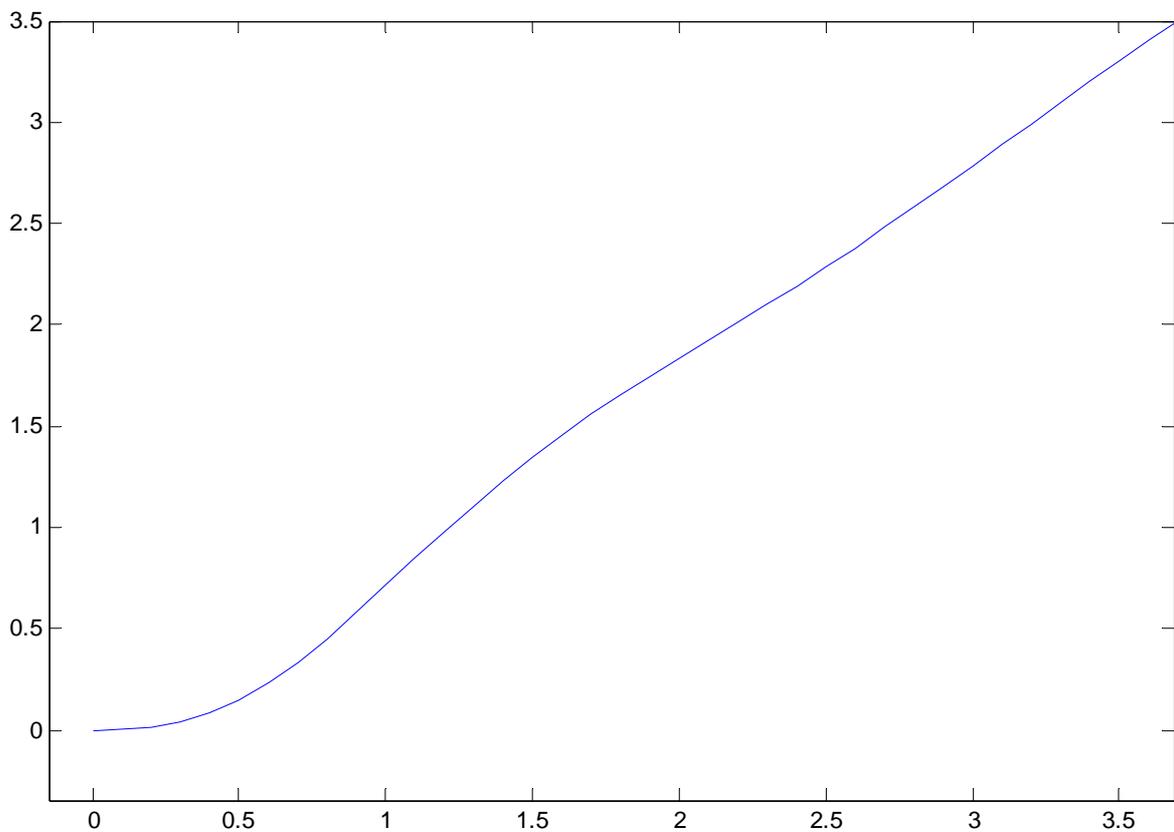


Figure 3 - Réponse de H à une rampe unité, zoom effectué pour observer le départ de la réponse

Une sinusoïde :

```
sinus=sin(t);     (ne pas employer sin comme nom car c'est une fonction de Matlab et cela entraînerait une "plante" du programme)  
z=lsim(num,den,sinus,t);  
plot(t,z)
```

Afin de visualiser la sinusoïde originale et le signal de sortie, taper :

**hold on**

**plot(t,sinus,'r')** 'r' permet d'affecter la couleur rouge à ce tracé pour le différencier du précédent

"hold on" permet de tracer plusieurs courbes dans la même fenêtre. "hold off" annule cette commande.

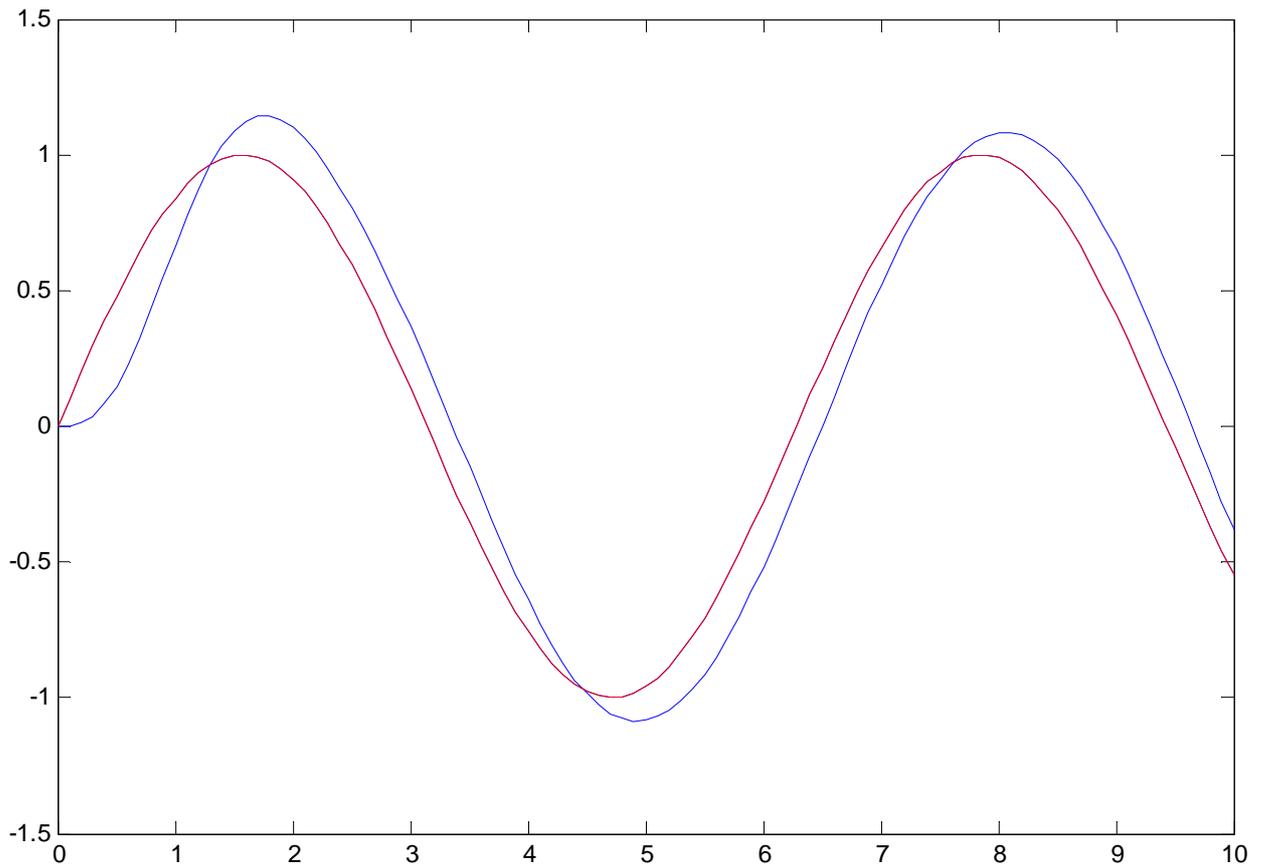


Figure 4 - Réponse de H à une excitation sinusoïdale (en bleu), sinusoïde excitatrice en rouge

#### 4) Etude fréquentielle

**nichols(num,den)**

**ngrid**

**line([-135 -135],[0 -50])** permet de tracer une ligne verticale d'abscisse  $-135^\circ$  pour la visualisation de la marge de phase à  $45^\circ$ .

La fonction zoom on permet d'agrandir une partie de la figure, cliquer a l'aide du bouton droit de la souris et tirer le pointeur en maintenant l'appui afin de délimiter une zone précise.

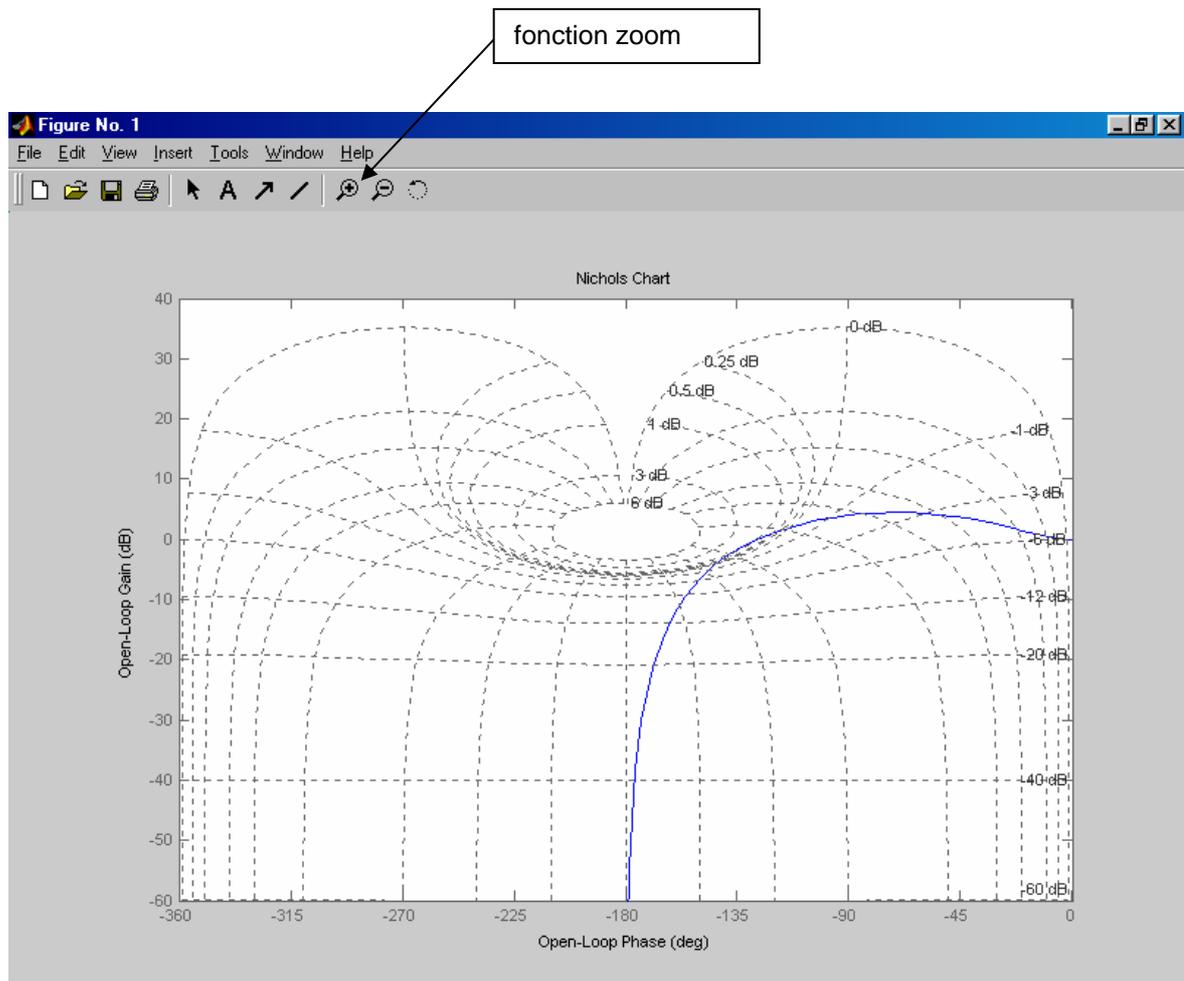


Figure 5 - Réponse fréquentielle de H dans le repère de Black Nichols

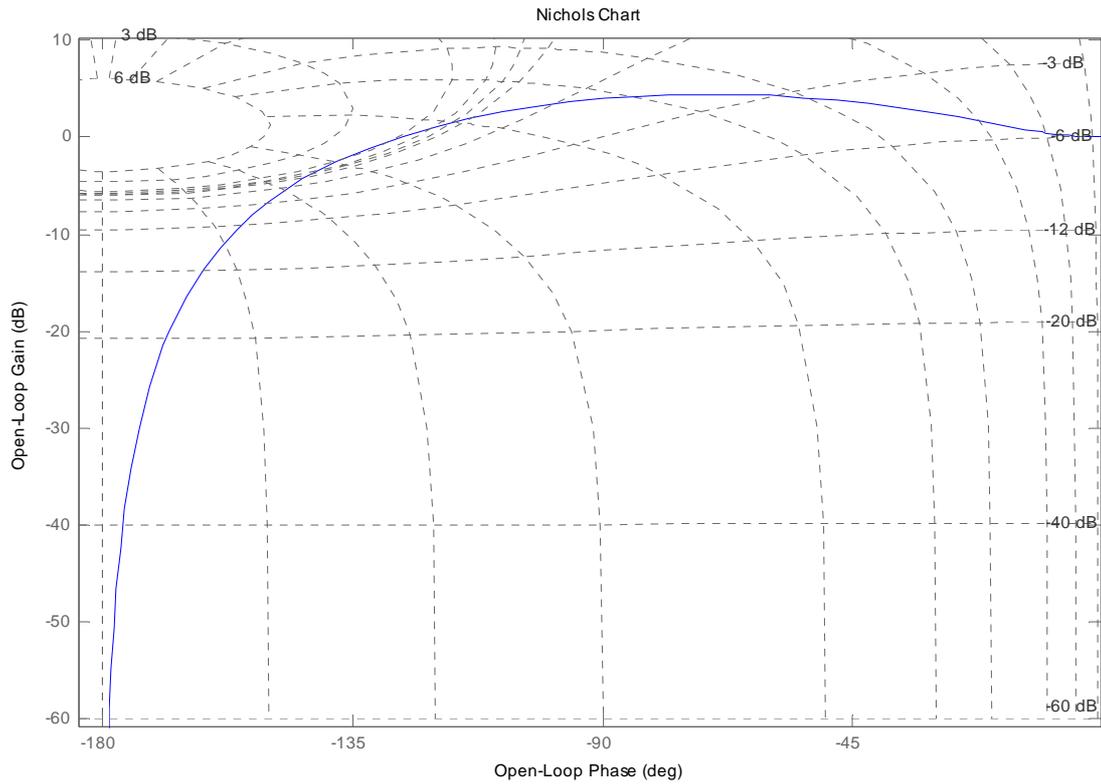


Figure 6 - Zoom sur la réponse fréquentielle de H dans Black Nichols

**bode(num,den)**

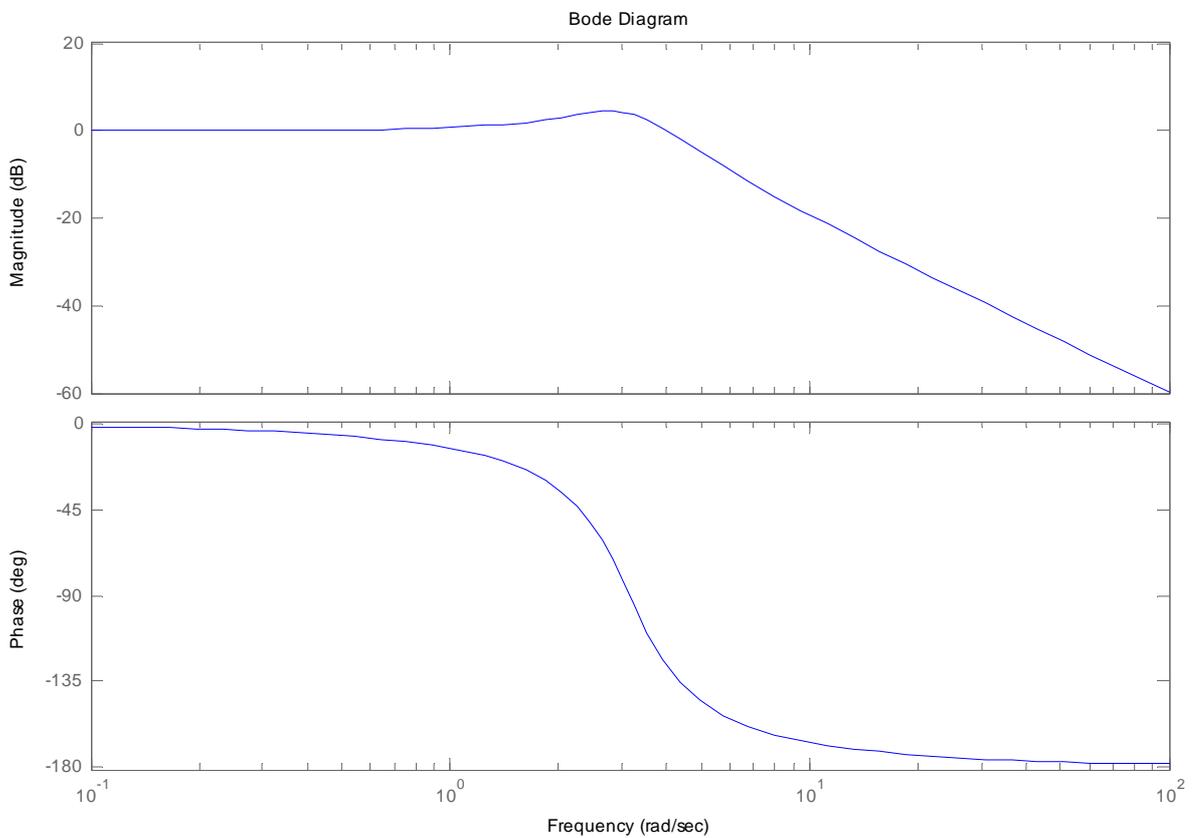


Figure 7 - Diagramme de Bode de H

**[Gm,Pm,Wcg,Wcp] = MARGIN(num,den)** mesure de la marge de phase et de la marge de gain ainsi que des pulsations correspondantes

Gm = Inf ; Pm = 53.1301

Wcg = Inf ; Wcp = 4.0000

**figure** pour ouvrir une nouvelle fenêtre "windows"

pour avoir un diagramme de Bode personnalisé :

```
puls=logspace(0,1,100);  
[ampli,phase,puls]=bode(num,den,puls);  
subplot(211),semilogx(puls,20*log10(ampli)),  
grid  
subplot(212),semilogx(puls,phase)  
grid
```

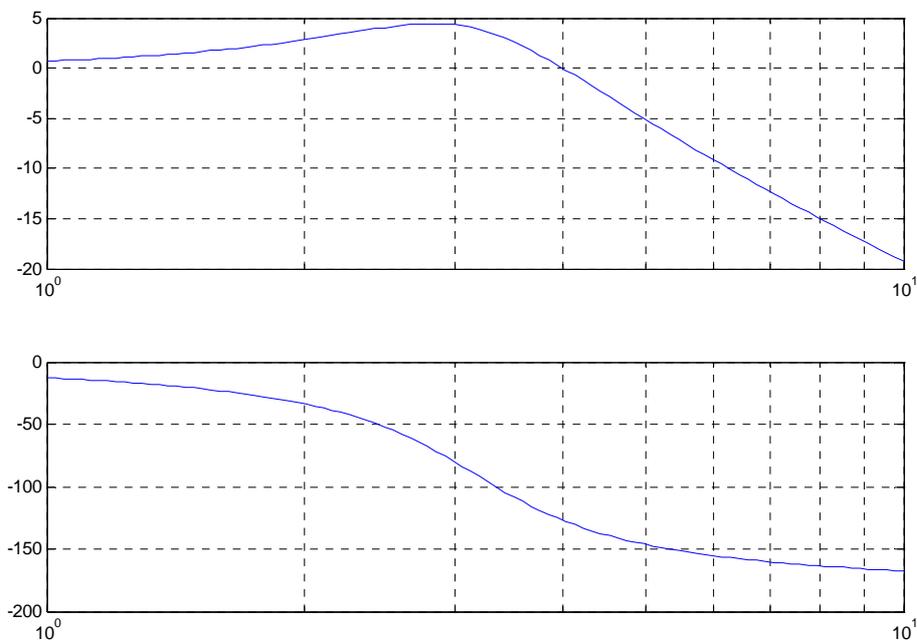


Figure 8 - Diagramme de Bode personnalisé pour H

Mesure de la marge de phase et de la marge de gain ainsi que des pulsations correspondantes :

```
margin(num,den)  
[Gm,Pm,Wcg,Wcp] = margin(num,den)
```

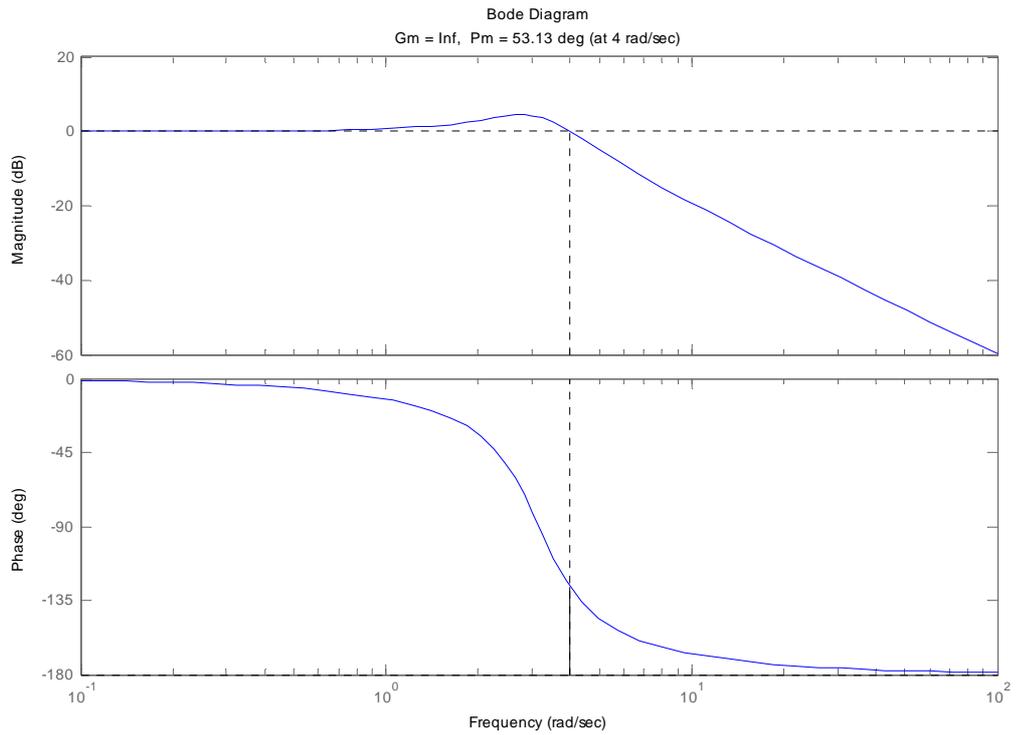


Figure 9 - Diagramme de Bode de H avec la marge de gain  
**figure**  
**nyquist(num,den)**

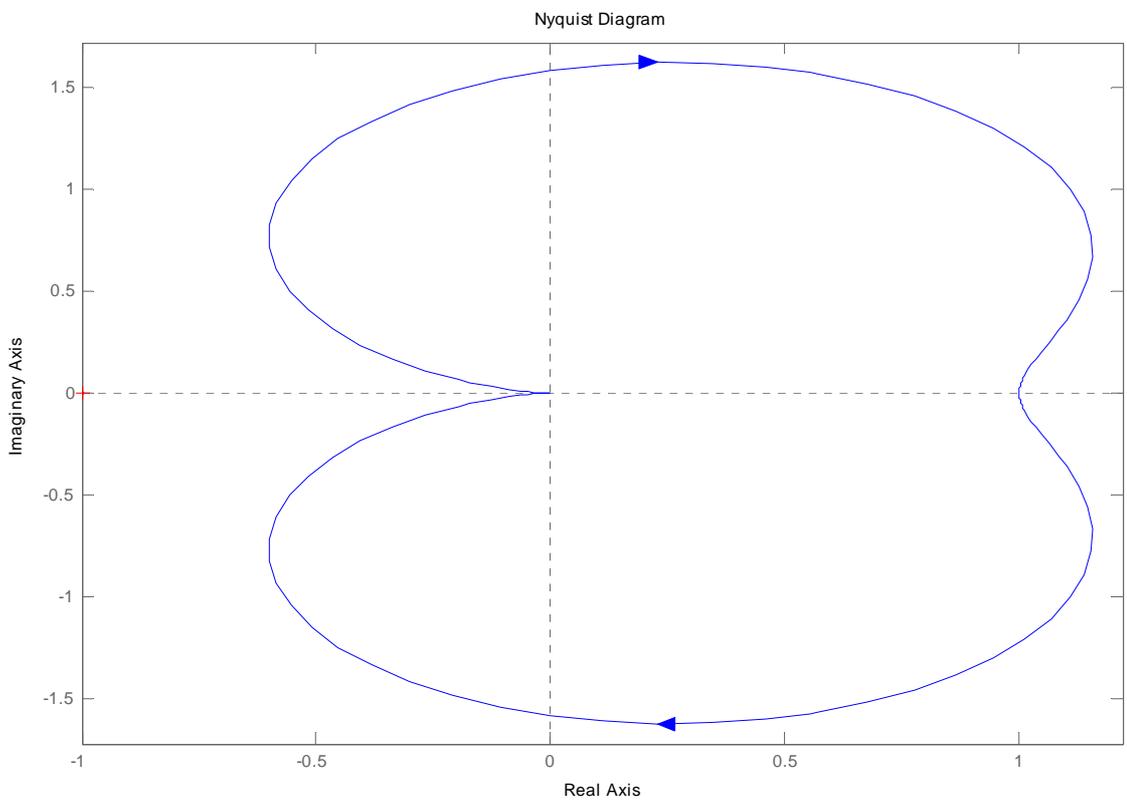


Figure 10 - Diagramme de Nyquist de H

## Lieu des racines

La méthode consiste à regarder où se trouvent les pôles du système bouclé lorsque l'on fait varier le gain du correcteur proportionnel.

Pour afficher le lieu des pôles:

**rlocus(num,den);**

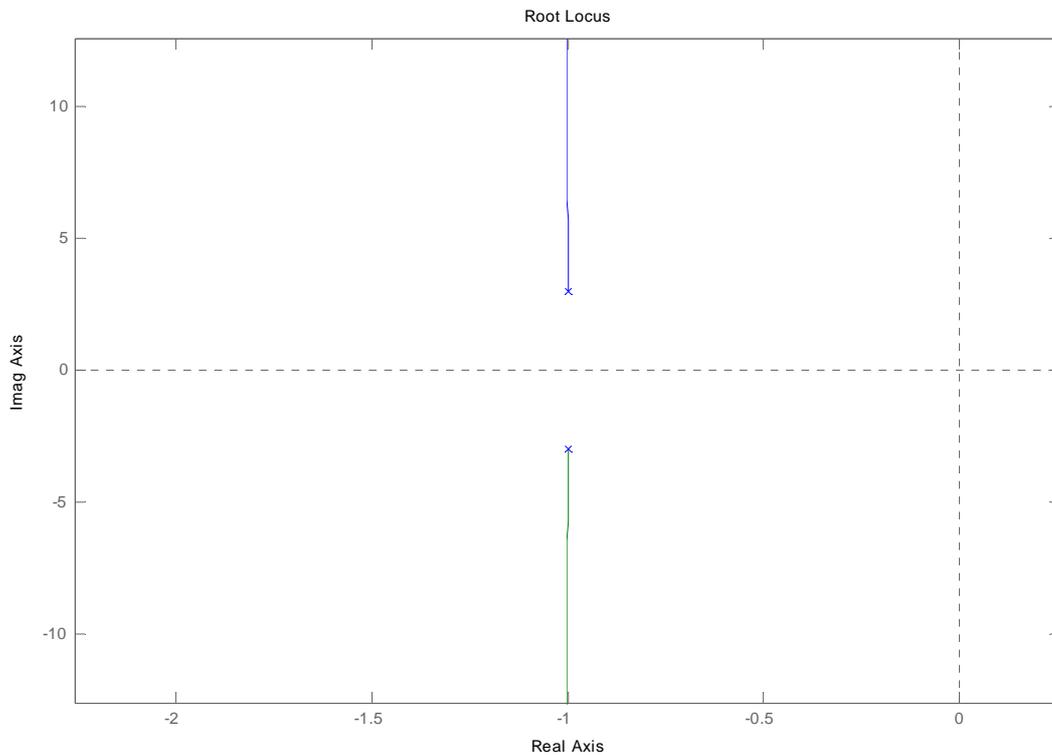


Figure 11 - Lieu des racines de H

Pour préciser une zone sur le lieu des pôles :

**zeta=0.2;**

**wo=5;**

**sgrid(zeta,wo)**

Pour chercher la valeur du gain k permettant d'obtenir les pôles désirés en boucle fermée

**[k,poles]=rlocfind(num,den);**

sélectionner un point sur le graphique, les coordonnées s'affichent dans le "workspace".

Pour construire un système en boucle fermée :

**[numf,denf]=cloop(num,den,-1);**

**step(numf,denf)** *permet de visualiser la réponse en boucle fermée unitaire à un échelon*

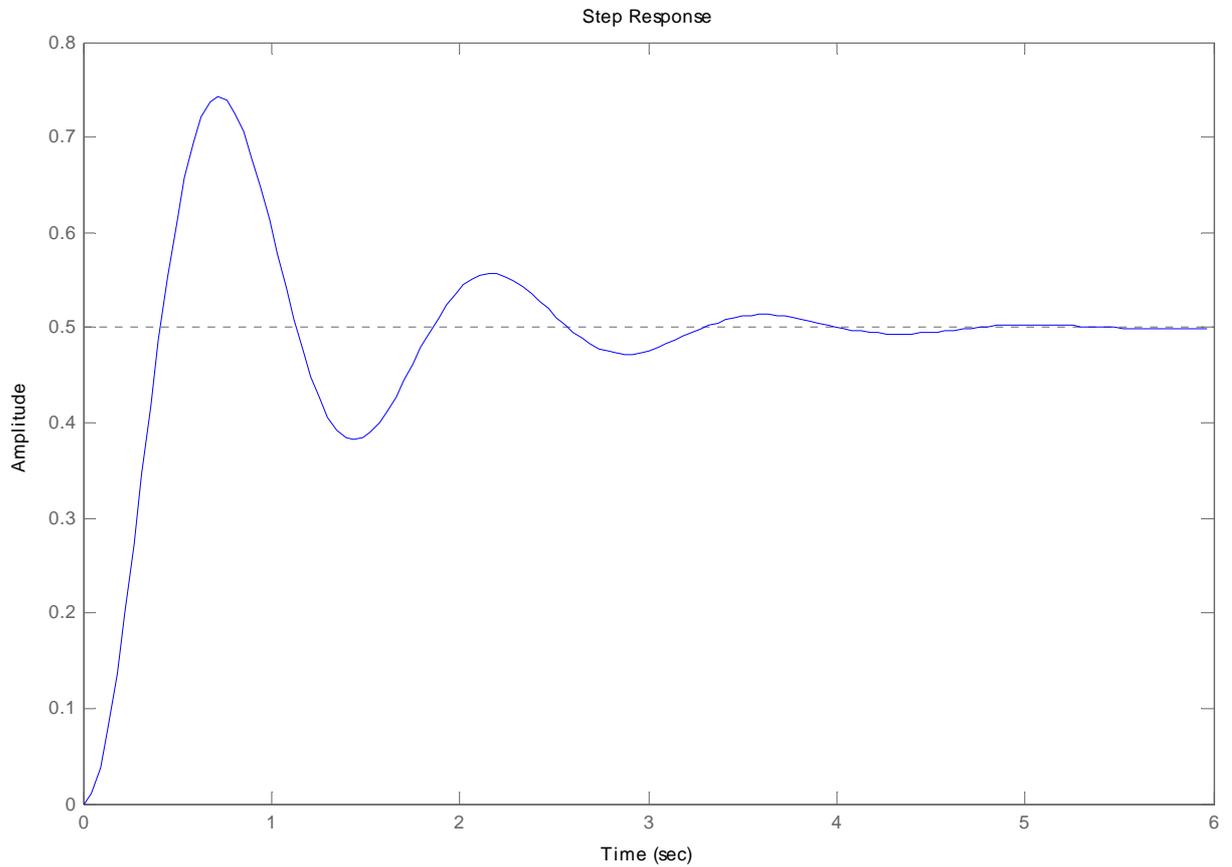


Figure 12 - Réponse du système en boucle fermé unitaire à un échelon

Il est facile de voir l'influence d'un correcteur. Par exemple, si on considère un simple gain de 6, il faut redéfinir le numérateur par l'instruction :

**num=60 (10×6)**

Redéfinir la fonction de transfert en boucle fermée :

**[numf,denf]=cloop(num,den,-1);**

et faire dessiner la réponse à un échelon

**step(numf,denf)**

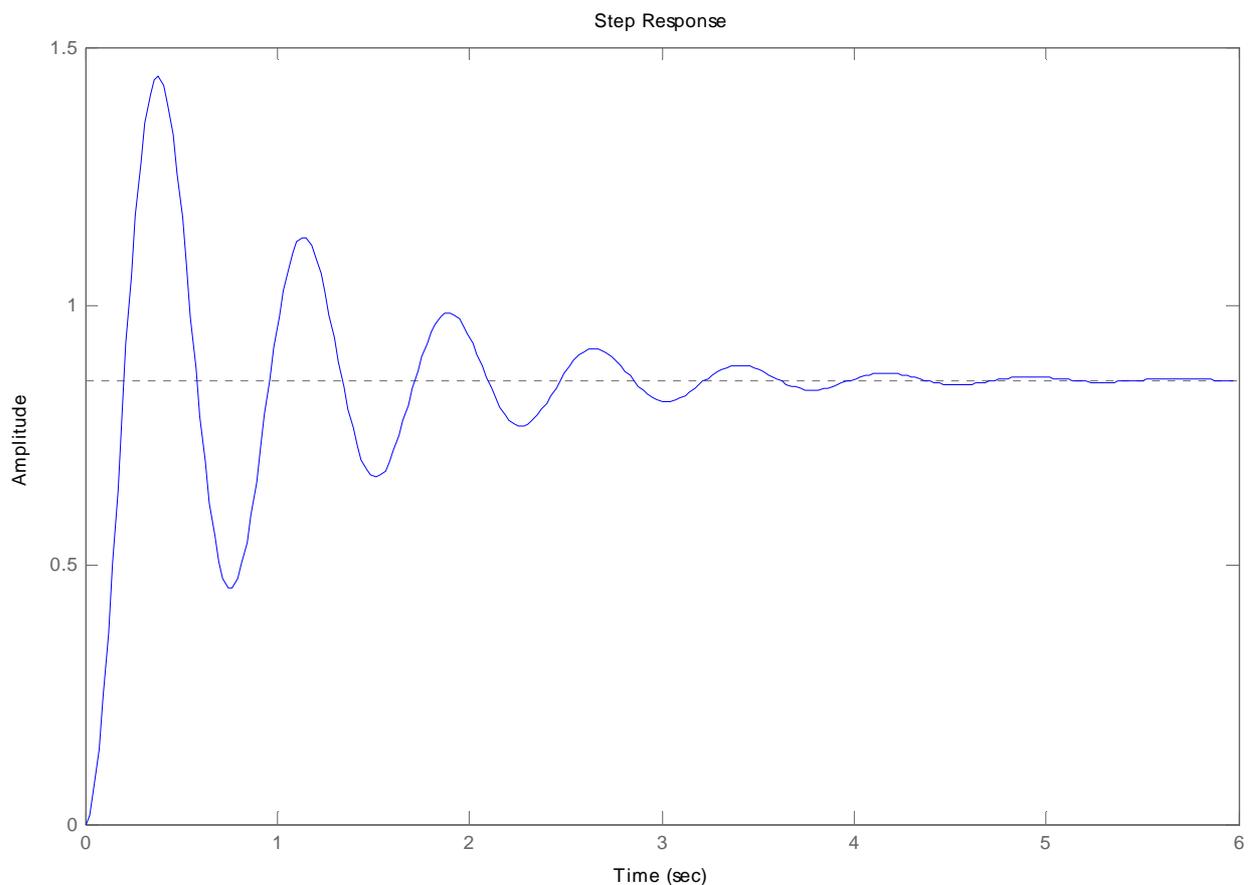


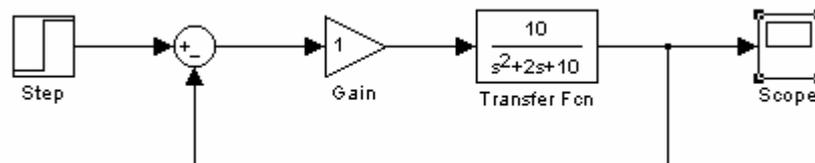
Figure 13 - Réponse de H en boucle fermée à un échelon avec un correcteur constitué d'un amplificateur de gain = 6

Il apparaît donc très facile de montrer l'influence du changement d'un paramètre sur les réponses. Là où il fallait avoir recours à des calculs longs et fastidieux, à des abaques, pour obtenir la réponse d'un système, le logiciel fournit cette réponse de manière aisée et rapide.

Une autre manière de procéder est de recourir à Simulink, extension logicielle de Matlab destinée à faciliter la modélisation de phénomènes physiques. Un schéma se substitue aux équations pour constituer le modèle, mais les équations différentielles représentatives du système physique sont sous-jacentes. Cette manière de résoudre numériquement des systèmes d'équations différentielles est très proche de la manière de penser des ingénieurs et des automaticiens qui raisonnent par schéma bloc, par fonction de transfert. Cela se faisait couramment avec les calculateurs analogiques.

## Analyse de H à l'aide de simulink

Ouvrir Simulink, soit par un clic sur son icône, soit en tapant simulink et en appuyant sur la touche entrée. Cliquer sur l'icône représentant une page blanche en haut à gauche "new model" afin d'ouvrir une nouvelle fenêtre permettant de construire H. Cliquer sur "continuous" dans la fenêtre "Simulink" et traîner à la souris une "transfer Fcn" dans la fenêtre vide. Cliquer sur "Math" et traîner dans la nouvelle fenêtre un "Gain", et un "sum". Cliquer sur "sinks" et traîner un "scope" dans la nouvelle fenêtre. Cliquer ensuite sur "Sources" et traîner un bloc "step" dans la nouvelle fenêtre. Raccorder les éléments comme indiqué ci-dessous.



Ce fichier peut être appelé en double cliquant sur [auto/H](#)

Double cliquer sur "transfer Fcn" et ajuster les coefficients du numérateur et du dénominateur. Le nombre le plus à gauche représente la plus grande puissance de p, créer un espace (barre d'espace), écrire le coefficient suivant (éventuellement 0), etc. Bien faire attention à ce que les coefficients soient entre crochets. Vérifier que la durée de la simulation est bien de 10 s en cliquant sur "simulation", "simulation parameters" (ne toucher à rien d'autre). Lancer la simulation, puis double cliquer sur le scope.

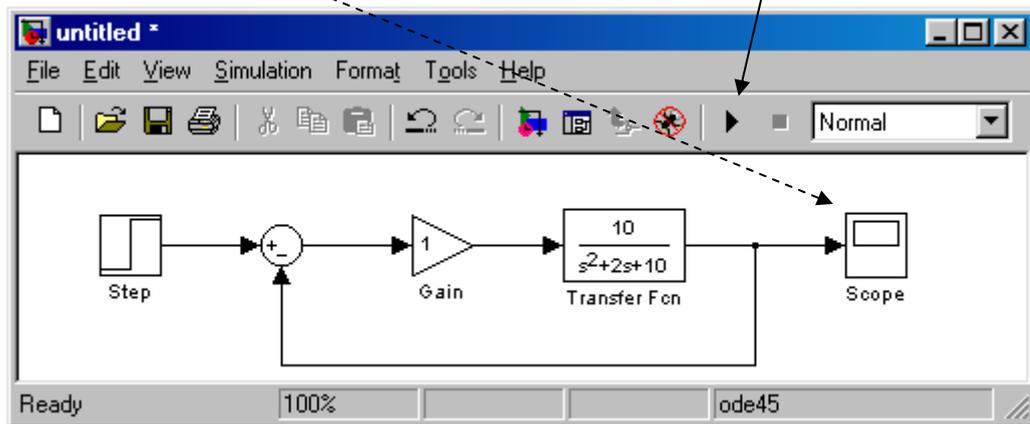


Figure 14 - H représentée par un schéma-bloc Simulink

L'échelon peut être réglé lui aussi en double cliquant dessus. Par défaut, il possède un amplitude de 1 et devient égal à 1 à  $t = 0,5$  s.

En double cliquant sur le scope, on obtient la figure suivante :

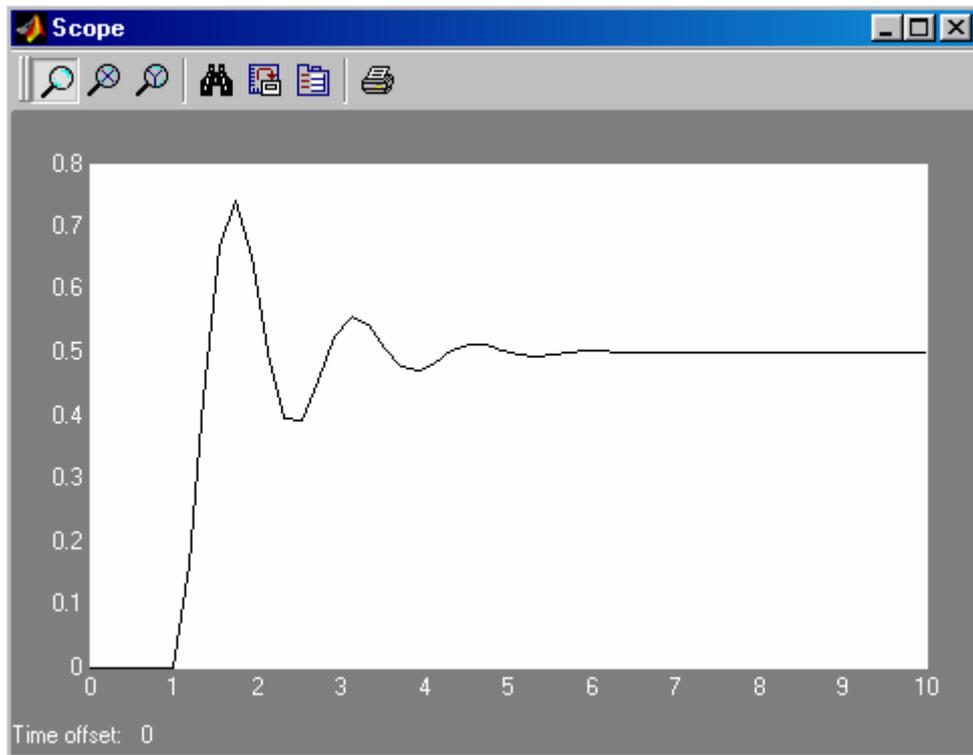


Figure 15 - Réponse de H à un échelon visualisée par un oscilloscope de Simulink

Cliquer éventuellement sur le bouton figurant une paire de jumelles afin d'ajuster la courbe à la taille de l'écran du scope.

Cette figure peut être ajustée à la taille désirée en cliquant avec le bouton gauche sur un coin de la figure et en traînant le coin à la souris en maintenant le bouton enfoncé.

Bien évidemment, la courbe est la même que celle de la Figure 16.

En double cliquant sur l'élément "gain", une boîte de dialogue s'ouvre et il est possible de modifier le gain. Relancer la simulation permet d'obtenir la nouvelle réponse sur le scope si on a pris soin de ne pas fermer cette fenêtre "scope". Il est possible d'ajuster rapidement la courbe à la fenêtre en cliquant sur l'icône représentant une paire de jumelles.

Afin d'étudier le réglage d'un PID et l'influence des différents paramètres, on reprend le schéma précédent sur Simulink. Un clic sur l'élément "gain" le sélectionne. Un appui sur la touche "suppr" le supprime. Il ne reste plus qu'à le remplacer par un élément PID que l'on trouve dans "Simulink Extras" "Additional Linear". On obtient le schéma suivant :

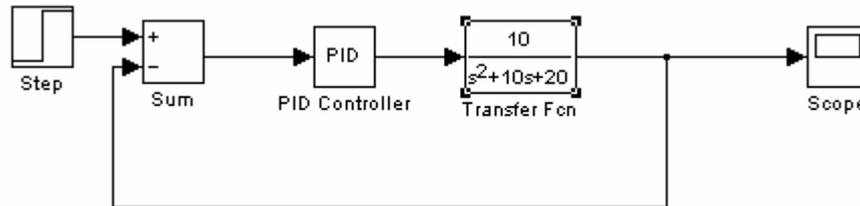


Figure 17 – Schéma-bloc Simulink montrant la fonction de transfert H contrôlée par un PID

En double cliquant sur l'élément PID, il est très facile de modifier les paramètres. Relancer la simulation fait apparaître la nouvelle réponse sur le scope.

On peut donc simuler rapidement en salle de classe n'importe quel système et procéder à la correction du système en boucle fermée que ce soit avec un PID ou à l'aide d'autre régulateur. Les possibilités ainsi offertes au professeur sont immenses.

Il n'a plus besoin d'équipement encombrant et coûteux pour expérimenter et mettre au point ses connaissances, ses cours, ses exercices et problèmes. Un PC équipé du logiciel adéquat suffit.

## Pilote automatique d'un navire

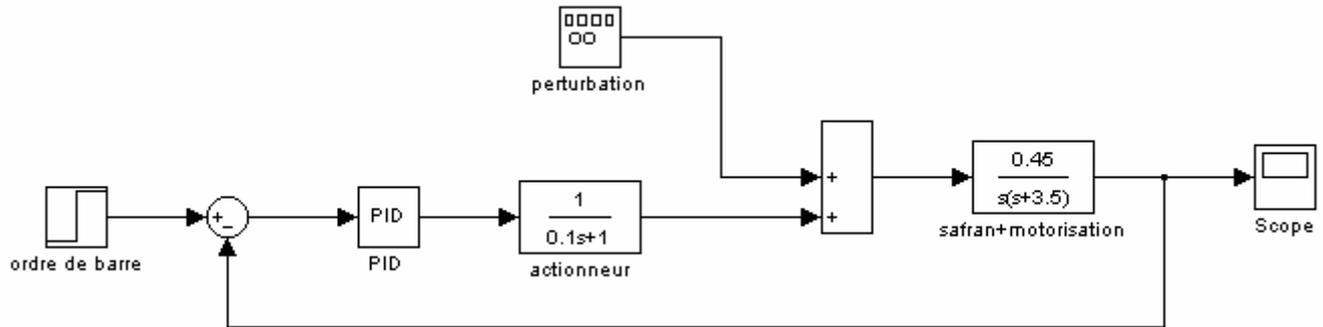


Figure 18 - Modélisation de la barre d'un navire à l'aide de Simulink

Appeler le fichier "pilot\_auto" du répertoire [auto/pilot\\_auto](#). Si Matlab est installé, un double-clic lance la simulation (de l'intérêt de la lecture sur l'écran du PC).

Le safran et sa motorisation ont été modélisés par une fonction de transfert comprenant une intégration au dénominateur (asservissement de position), les coefficients proviennent d'un polycopié d'exercices de l'ENSTA.

Les perturbations sont générées par un générateur de fonctions qui fournit des sinusoïdes de fréquence 0,4 Hz (des vagues agissant sur le safran), leur amplitude a été réglée à 20. Ces valeurs sont modifiables à souhait.

L'actionneur (vannes de télécommande de l'hydraulique du safran) a été modélisé par une fonction de transfert du premier ordre avec une constante de temps de 1/10s.

L'ordre de barre est  $10^\circ$  (babord ou tribord, peu importe ici), il est donné à  $t = 1s$ .

Après avoir déclenché la simulation, l'instruction "plot(y(:,2),y(:,1))" dans le "workspace" de Matlab déclenche le tracé suivant :

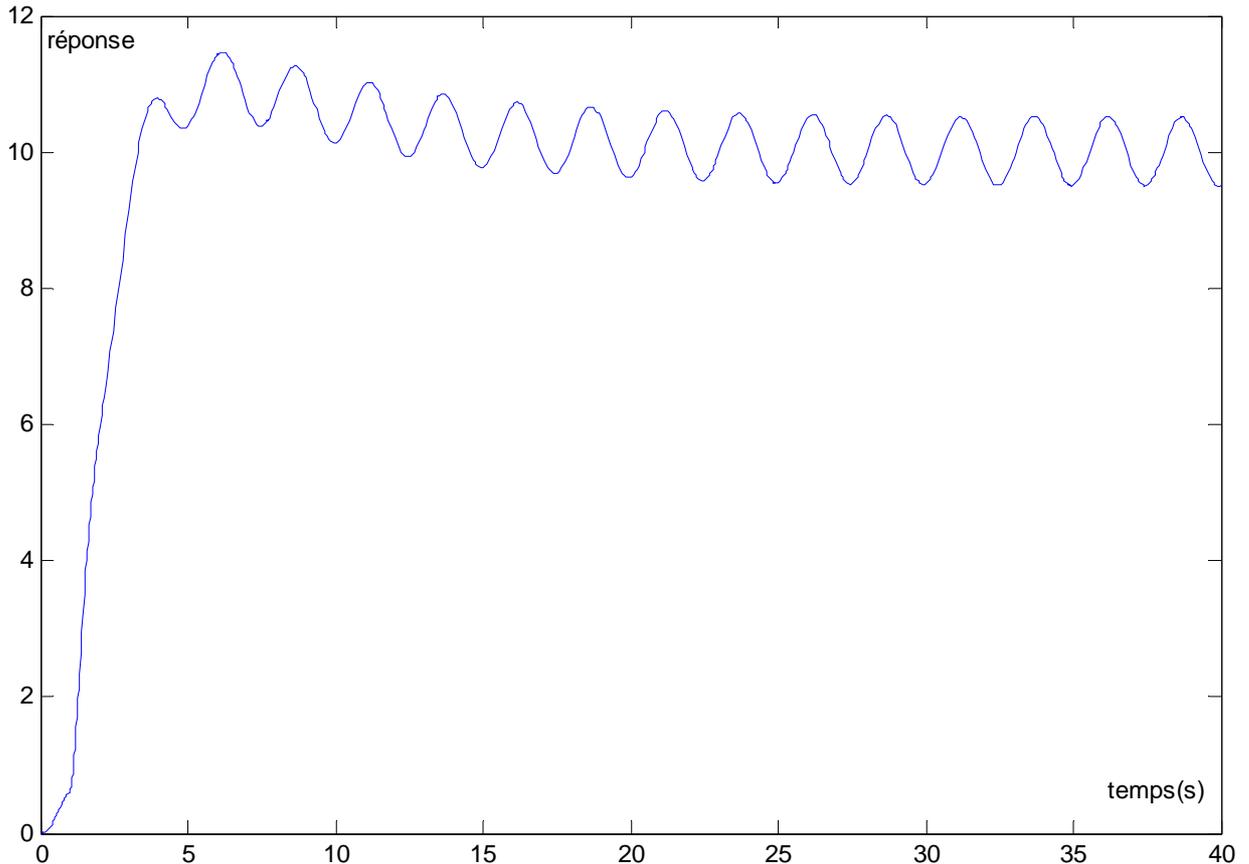


Figure 19 - Réponse de l'asservissement de position "barre de navire" à un ordre de barre de  $10^\circ$  intervenant à  $t = 1s$ , les perturbations sont sinusoïdales

On remarque le filtrage des perturbations, elles sont très atténuées. Néanmoins, la réponse est "bruitée" par ces perturbations.

Un peu de « dérivée » peut améliorer la réponse, notamment si l'amplitude des perturbations augmente.

Dans le domaine des travaux pratiques également, l'utilisation de logiciels comme Matlab/Simulink permet, par exemple, de se familiariser avec les systèmes du premier ordre et du second ordre de manière beaucoup plus enrichissante, ludique et bon marché qu'avec des méthodes classiques nécessitant des oscilloscopes performants (et donc onéreux) et une réserve de composants (résistances, condensateurs, selfs) qui fait souvent défaut dans les salles d'automatique.

L'étude des réponses fréquentielles est très simple avec Matlab, il n'est plus nécessaire de passer des heures à faire des mesures avec des oscilloscopes souvent inadaptés pour obtenir un lieu de Nichols !

La simulation permet de régler des PID en s'affranchissant des difficultés rencontrées lors de manipulation sur des maquettes réelles. Prenons l'exemple des maquettes de régulation de niveau d'eau. Ces dernières sont chères (plus de 10000 euros), la salle de

TP d'automatique n'en possède donc généralement qu'une seule entraînant la nécessité de faire tourner les groupes de TP. Ces maquettes sont généralement vieilles et fonctionnent mal, voire plus du tout, entraînant un gaspillage de temps pour le professeur qui essaye de remettre en état du matériel au lieu d'améliorer et de créer des TP et pour les élèves qui ne peuvent observer un fonctionnement correct.

Un fichier Simulink tournant sur 6 PC permet de réaliser un TP de réglage de PID de manière sûre et agréable pour tous les groupes en même temps.

Eloigner les élèves du concret pour leur faire manipuler une réalité simulée n'est pas si nuisible puisque, de toutes les manières, l'automatique a pour but de modéliser les systèmes à contrôler. Je reste persuadé que le recours à la simulation en TP est un excellent compromis qui permet, malgré tout, de réaliser nos objectifs de formation.