

# TD1 et TD2 - Analyse de la robustesse de contrôleurs (systèmes avec paramètres incertains et délais)

Thao Dang

16 octobre 2012

Nous illustrons l'étude de la robustesse de contrôleurs au travers de l'exemple du contrôleur d'orientation de robots LEGO. Le contrôleur de distance peut être analysé de la même manière.

## Contrôleur d'orientation

Les équations différentielles qui décrivent la dynamique du robot sont :

$$\dot{x} = (v_g + v_d)\cos(\theta)/2 \quad (1)$$

$$\dot{y} = (v_g + v_d)\sin(\theta)/2 \quad (2)$$

$$\dot{\theta} = (v_d - v_g)/l \quad (3)$$

où  $x$  et  $y$  sont les coordonnées de la position courante le robot ;  $\theta$  est son orientation ;  $v_g$  et  $v_d$  sont les vitesses des roues à gauche et à droite ;  $l$  est la distance entre les deux roues.

Nous contrôlons **l'orientation** du robot par un contrôleur de type PI dont la fonction de transfert est

$$H_{\theta}(s) = \frac{ki_{\theta}}{s} + kp_{\theta} \quad (4)$$

où  $ki_{\theta}$  est le coefficient de l'action proportionnelle et  $kp_{\theta}$  est le coefficient de l'action intégrale.

Comme nous utilisons la différence entre les vitesses des roues pour contrôler l'orientation  $\theta$  du robot, nous avons la variable de commande  $u_{\theta} = (v_d - v_g)$ . De l'équation (3) on obtient :

$$\dot{\theta} = u_{\theta}/l$$

ce qui donne en Laplace  $s\theta(s) = u_{\theta}(s)/l$  (en supposant que les conditions initiales sont 0). Alors, la fonction de transfert de  $\theta$  est

$$H_{\theta}(s) = \frac{\theta(s)}{u_{\theta}(s)} = \frac{1}{ls}$$

Le schéma de commande est comme suit. La sortie  $u_{\theta}$  du contrôleur  $H_{\theta}(s)$  est connectée à l'entrée du bloc  $H_{\theta}(s)$  correspondant à la dynamique de l'orientation du robot. L'entrée du contrôleur  $C_{\theta}(s)$  est la différence entre le retour de la sortie  $\theta$  de  $H_{\theta}(s)$  et l'orientation désirée  $\theta^*$ . L'évolution de  $\theta$  contrôlée par la commande  $u_{\theta}$  peut donc être représenté par un système en boucle fermée dont la fonction de transfert est :

$$\begin{aligned} H(s) &= \frac{C_{\theta}(s)H_{\theta}(s)}{1 + C_{\theta}(s)H_{\theta}(s)} \\ &= \frac{\frac{kp_{\theta}}{l}s + \frac{ki_{\theta}}{l}}{s^2 + \frac{kp_{\theta}}{l}s + \frac{ki_{\theta}}{l}} \end{aligned}$$

## Discrétisation du contrôleur

Pour implanter le contrôleur en temps-continu ci-dessus, il faut le discrétiser. Si on choisit une période d'échantillonnage  $T$  et une approximation du pre-

mier ordre

$$s = \frac{z - 1}{zT}$$

où  $T$  est la période d'échantillonnage, on obtient la fonction de transfert en temps discret  $H(z)$ .

## Système avec des paramètres incertains

Nous allons étudier la robustesse du contrôleur d'orientation vis-à-vis aux paramètres tels que  $T$ ,  $ki_\theta$ ,  $kp_\theta$ ,  $l$ . Afin d'utiliser les critères pour les systèmes en temps continu, nous utilisons la transformation suivante qui ramène le cercle unitaire en  $z$  vers le demi-plan gauche en  $\eta$

$$z = \frac{1 - \eta}{1 + \eta}$$

La fonction caractéristique en  $\eta$  est :

$$p(\eta) = (4 + 2\frac{kp_\theta}{l}T + \frac{ki_\theta}{l}T^2)\eta^2 + (-2\frac{kp_\theta}{l}T - 2\frac{ki_\theta}{l}T^2)\eta + \frac{ki_\theta}{l}T^2 \quad (5)$$

Nous pouvons maintenant traiter la variable  $\eta$  comme la variable  $s$  dans les fonctions de transfert en Laplace.

## 1 Questions 1

En utilisant les fonctions matlab dans le fichier **hurwitz.m** donné, calculer la matrice de Hurwitz pour déterminer la stabilité du contrôleur pour des différentes valeurs de la période d'échantillonnage  $T$  et des coefficients  $kp_\theta$  et  $ki_\theta$  du PI.

**Réponse.** *En matlab, après donner des valeurs numériques (à tester) aux paramètres  $a$ ,  $b$  et  $T$ , on définit le polynôme  $p$  ci-dessus par un vecteur de ses coefficients :*

$$p = [(4 + 2\frac{kp_\theta}{l}T + \frac{ki_\theta}{l}T^2) \quad (-2\frac{kp_\theta}{l}T - 2\frac{ki_\theta}{l}T^2) \quad \frac{ki_\theta}{l}T^2]$$

Puis, on calcule sa matrice de Hurwitz :

$$H = \text{hurwitz}(p, \text{deg}P)$$

où  $\text{deg}P = 2$  est le degré de  $p$ .

Pour utiliser le critère de Hurwitz, on regarde les mineurs principaux de la matrice  $H$ . Pour assurer la stabilité de  $p$ , il faut que ces mineurs soient positifs.

Pour tester un intervalle de valeurs d'un paramètre, on peut inclure le test ci-dessus dans une boucle "for" qui itère sur des valeurs sélectionnées à partir d'une grille sur l'intervalle. Notons que la complexité de cette méthode est exponentielle en nombre de paramètres.

## 2 Questions 2

Quand il y a plusieurs paramètres, la méthode de tester des valeurs sur une grille dans l'espace de paramètres peut devenir très coûteuse. On souhaite donc trouver une condition analytique pour la robustesse du système en utilisant le critère Bialas.

On note les coefficients du polynôme  $p$  comme suit :

$$(4 + 2\frac{kp_\theta}{l}T + \frac{ki_\theta}{l}T^2) = A$$

$$(-2\frac{kp_\theta}{l}T - 2\frac{ki_\theta}{l}T^2) = B$$

$$\frac{ki_\theta}{l}T^2 = C$$

Pour étudier la stabilité du polynôme  $p$ , il suffit d'étudier le polynôme "normalisé" suivant :

$$\pi(\eta) = \eta^2 + \frac{B}{A}\eta + \frac{C}{A}$$

Pour utiliser le **critère Bialas**, on décompose  $\pi$  en deux polynômes comme suit :

$$\pi(\eta) = (\eta^2 + 2\sqrt{C/A}\eta + \frac{C}{A}) + (-2\sqrt{C/A})\eta + B/A\eta \quad (6)$$

$$= (\eta + \sqrt{C/A})^2 + (-2\sqrt{C/A} + B/A)\eta \quad (7)$$

Notons

$$\pi_0(\eta) = (\eta + \sqrt{C/A})^2$$

,

$$\pi_1(\eta) = (-2\sqrt{C/A} + B/A)\eta = \eta$$

et

$$q = (-2\sqrt{C/A} + B/A)$$

Alors,

$$\pi(\eta) = \pi_0(\eta) + q\pi_1(\eta)$$

On peut maintenant utiliser le **critère Bialas** puisque  $A > 0$  et  $C > 0$ , le polynôme  $\pi_0$  est toujours stable. En considérant  $q = (-2\sqrt{C/A} + B/A)$  comme paramètre, utiliser le **critère Bialas** pour trouver la condition de stabilité robuste.

**Réponse.** On va utiliser le script matlab suivant.

```
%% Programme matlab
l=0.2; %axe inter-roue

% Parametres du controleur d'orientation
alpha=1/l;
omega=1;
ksi=0.005;

ki_teta=omega*omega/alpha;
kp_teta=2*ksi*omega/alpha;

%periode d'echantillonage
```

```

T = 0.000001;

A = 4*l + 2*kp_teta*T + ki_teta*T*T;
B = -2*kp_teta*T - 2*ki_teta*T*T;
C = 2*ki_teta*T*T;

% coefficients de p1
c01 = 0;
c11 = 1;

% coefficients de p0
c00 = C/A;
c10 = 2*sqrt(C/A);
c20 = 1;

q = -(2*sqrt(C/A) - B/A)
qinv = 1/q;

p0 = [c20  c10  c00];
p1 = [c11  c01];

H0 = hurwitz(p0,2)
H1 = hurwitz(p1,1)

H = -inv(H0)*H1

lambda = eig(H)

```

A partir des valeurs propres

lambda

calculé par le programme, on déduit les conditions suivantes pour les paramètres  $a$ ,  $b$ ,  $T$ .

$$q_{min} \leq (-2\sqrt{C/A} + B/A) \leq q_{max}$$

où  $q_{max} = \lambda_{max}^+$  (la valeur propre positive la plus grande) et  $q_{min} = \lambda_{min}^-$  (la valeur propre négative la plus petite). Ensuite on doit tenir compte des conditions sur les valeurs choisies de  $A, B, C$  dans le programme.

### 3 Questions 3

Appliquer le théorème Kharitonov pour étudier la stabilité par rapport à la distance  $l \in [0.19, 0.21]$  (les autres paramètres du contrôleur sont fixés, par exemple :  $ki_\theta = 0.4, kp_\theta = 0.2, T = 0.0001$ ). Pour vérifier la stabilité des polynômes de Kharitonov, utiliser le programme matlab **hurwitz.m**.

**Réponse.** On reprend le polynôme (5) :

$$p(\eta) = (4 + 2\frac{kp_\theta}{l}T + \frac{ki_\theta}{l}T^2)\eta^2 + (-2\frac{kp_\theta}{l}T - 2\frac{ki_\theta}{l}T^2)\eta + \frac{ki_\theta}{l}T^2$$

Et on note ses coefficients par  $q_2, q_1$  et  $q_0$ , c-a-d

$$\pi(\eta) = q_2\eta^2 + q_1\eta + q_0$$

Pour chaque coefficient  $q_i$ , on détermine un intervalle  $[q_{im}, q_{ip}]$  qui contient toutes ses valeurs. Puis, on calcule les 4 polynômes de Kharitonov (voir le script matlab suivant).

```
l=0.2; %axe inter-roue
l1=0.19;
l2=0.21;
```

```
% Parametres du controleur d'orientation
% On utilise les coefficients de PI calcule' avec la valeur nominale de l
alpha=1/l;
omega=1;
ksi=1;
```

```
%omega=1;
%ksi=0.005;
```

```

ki_teta=0.4;
kp_teta=0.2;

%periode d'echantillonnage
T = 0.00001;

q2m = 4 + 2*kp_teta*T/l2 + 2*ki_teta*T/l2;
q2p = 4 + 2*kp_teta*T/l1 + 2*ki_teta*T/l1;
q1m = -2*T*kp_teta/l1 -2*T*T*ki_teta/l1;
q1p = -2*T*kp_teta/l2 -2*T*T*ki_teta/l2;
q0m = T*T*ki_teta/l2;
q0p = T*T*ki_teta/l1;

Femin=[ q2p 0 q0m];
Femax=[ q2m 0 q0p];
Fomin=[ 0 q1m 0];
Fomax=[ 0 q1p 0];

Pmm = Femin + Fomin;
Ppm = Femax + Fomin;
Pmp = Femin + Fomax;
Ppp = Femax + Fomax;

H1 = hurwitz(Pmm,2)
H2 = hurwitz(Ppm,2)
H3 = hurwitz(Pmp,2)
H4 = hurwitz(Ppp,2)

```

## 4 Questions 4

Le délai  $\tau$  du moteur peut être introduit dans la fonction de transfert :

$$H_{\theta}(s) = \frac{e^{-\tau s}}{l s}$$

Avec le contrôleur PI

$$C_{\theta}(s) = \frac{ki_{\theta}}{s} + kp_{\theta}.$$

- Calculer le polynôme caractéristique de la boucle fermée
- Etudier sa stabilité par rapport au délai  $\tau$

**Réponse.**

```
l=0.2; %axe inter-roue
alpha=1/l;
omega=0.2;
ksi=0;
ki_teta=omega*omega/alpha;
kp_teta=2*ksi*omega/alpha;
tau=0.001;

[num, denum] = pade(tau, 1)

% p(s) = s^2*denum + (kp*s + ki)*num/l
% p1 = s^2
% p2 = (kp*s + ki)/l

p1 = [1 0 0];
p2 = [kp_teta/l ki_teta/l];

ph = conv(denum,p1)
pl = conv(num,p2)

[sh1, sh2] = size(ph); %sh2 = degre de ph
[sl1, sl2] = size(pl); %sl2 = degre de pl
nbzeroadd = sh2 - sl2;

pll = [zeros(nbzeroadd) pl];
%ajouter des coef 0 pour que les vecteurs aient la meme taille
%afin de faire l'addition de deux polynomes
```

```
p = ph + pl1  
[s1, s2] = size(p);  
hurwitz(p,s2-1)
```