

TD - Analyse de la robustesse de contrôleurs

Thao Dang

25 septembre 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 ; θ 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_θ est le coefficient de l'action proportionnelle et kp_θ est le coefficient de l'action intégrale.

Comme nous utilisons la différence entre les vitesses des roues pour contrôler l'orientation θ 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 θ est

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

Le schéma de commande est comme suit. La sortie u_θ 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 θ de $H_\theta(s)$ et l'orientation désirée θ^* . L'évolution de θ contrôlée par la commande u_θ 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 premier 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_θ , kp_θ , 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 η

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

La fonction caractéristique en η est :

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

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

Question 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_θ et ki_θ du PI.

Question 2 : 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 polynôme p , il suffit d'étudier le polynôme suivant :

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

D'abord on écrit le polynôme π comme suit :

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

Puis on le décompose en deux polynômes pour utiliser le **critère Bialas** :

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

où

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

et

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

En considérant $q = (-2\sqrt{C/A} + B/A)$ comme paramètre, utiliser le **critère Bialas** pour trouver la condition de stabilité robuste.