Mini-projet– M2 SAR
Commande non-linéaire

Devoir maison

Stabilisation de trajectoires pour un véhicule de type voiture

Un robot de type voiture est classiquement représenté, d'un point de vue cinématique, comme un système équivalent à deux roues : une roue arrière fixe par rapport au corps du véhicule, et une roue avant orientable. Ces deux roues équivalentes sont représentées en bleu pointillé sur la figure ci-dessous.

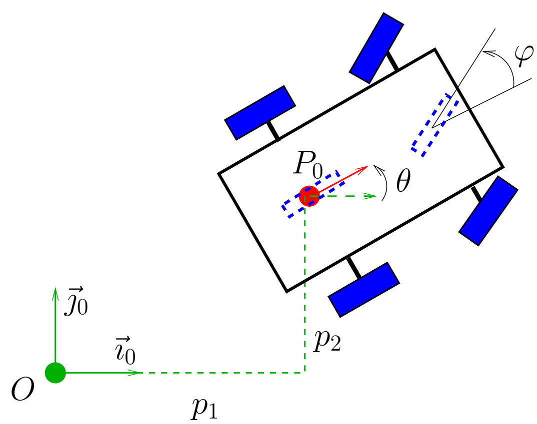

FIGURE 1 – Véhicule de type voiture

Le modèle cinématique d'un tel système est donné par les équations suivantes :

$$\left\{\begin{array}{rcl} \dot{p} & = & u_1 R(\theta) e_1 \\ \dot{\theta} & = & u_1 \dfrac{\tan\varphi}{L} \\ \dot{\varphi} & = & u_2 \end{array}\right. \tag{1}$$

où :

- $p = (p_1, p_2)^T$ est le vecteur des coordonnées du point P_0 par rapport à un repère fixe (voir figure) ;
- θ est l'orientation du robot par rapport à ce même repère ;
- φ est l'angle de braquage, par rapport au corps du robot, de la roue avant équivalente ;
- L est la distance entre l'essieu arrière et l'essieu avant du véhicule ;
- u_1 et u_2 sont les commandes du système, qui représentent respectivement la vitesse de roulement du véhicule et la vitesse de braquage.

Par ailleurs, les notations suivantes sont utilisées dans (1) :

$$R(\theta) = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} , \quad e_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

Le modèle (1) peut se mettre sous la forme classique $\dot{x} = f(x, u)$ avec $x = (p_1, p_2, \theta, \varphi)^T$ et $u = (u_1, u_2)^T$.

L'objectif de commande étudié est de stabiliser la trajectoire du robot à une trajectoire de référence. On suppose que cette trajectoire de référence est un mouvement à vitesse et courbure constante (finie), de sorte que ce mouvement de référence est une solution particulière $x_r = (p_{r,1}, p_{r,2}, \theta_r, \varphi_r)^T$ du système telle que :

$$\begin{cases} \dot{p}_r &= u_{r,1} R(\theta_r) e_1 \\ \dot{\theta}_r &= u_{r,1} \dfrac{\tan\varphi_r}{L} \\ \dot{\varphi}_r &= u_{r,2} = 0 \end{cases} \tag{2}$$

avec $u_{r,1}$ une constante **non nulle**. On notera que $\dot{\theta}_r$ est donc aussi constante.

Question 1 : On définit le vecteur d'erreur suivant :

$$\tilde{x} = \begin{pmatrix} \tilde{p} \\ \tilde{\theta} \\ \tilde{\varphi} \end{pmatrix} \qquad \text{avec } \tilde{p} = R(-\theta_r)(p - p_r)\ ,\ \tilde{\theta} = \theta - \theta_r\ ,\ \tilde{\varphi} = \varphi - \varphi_r$$

Vérifier que $\|\tilde{x}\| = \|x - x_r\|$, et en déduire que la stabilisation asymptotique de x vers x_r est équivalente à la stabilisation asymptotique de $\tilde{x}$ à zéro.

Question 2 : Vérifier les relations suivantes :

$$\begin{cases} \dot{\tilde{p}} &= u_1 R(\tilde{\theta}) e_1 - u_{r,1} e_1 - \dot{\theta}_r S \tilde{p} \\ \dot{\tilde{\theta}} &= u_1 \dfrac{\tan\varphi}{L} - u_{r,1} \dfrac{\tan\varphi_r}{L} \\ \dot{\tilde{\varphi}} &= u_2 \end{cases} \tag{3}$$

où

$$S = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}$$

Question 3 : A partir de la question précédente, mettre la dynamique du système d'erreur sous la forme

$$\dot{\tilde{x}} = \tilde{f}(\tilde{x}, \tilde{u}, u_{r,1}, \varphi_r)$$

avec $\tilde{u} = (\tilde{u}_1, \tilde{u}_2)^T = (u_1 - u_{r,1}, u_2 - u_{r,2})^T = (u_1 - u_{r,1}, u_2)^T$ et $\tilde{f}$ une fonction que l'on spécifiera. On remarquera que puisque $u_{r,1}$ est supposée constante et $u_{r,2} = 0$, φ_r est également constante, de sorte que le système ci-dessus est stationnaire (i.e., il ne dépend pas de la variable exogène t, contrairement au cas général où $u_{r,1}$ et $u_{r,2}$ varieraient en fonction de t).

Question 4 : Déterminer une loi de commande linéaire $\tilde{u}$ qui permette de stabiliser $\tilde{x}$ à zéro, au moins localement. On se basera sur l'étude du système linéarisé tangent. On développera les calculs et l'on proposera un réglage des gains de commande, par exemple en rendant la dynamique longitudinale (i.e., la dynamique de $\tilde{p}_1$) indépendante de la dynamique des autres variables (i.e., la dynamique latérale). Ceci revient donc à séparer la dynamique en deux sous systèmes : un premier ordre, et un troisième ordre. On pourra effectuer le réglage de gains pour la valeur particulière $\varphi_r = 0$ (i.e., pour un mouvement de référence en ligne droite).

Question 5 : On souhaite maintenant synthétiser une loi de commande permettant de garantir un "grand domaine de convergence". Pour ce faire, on commence par effectuer un premier changement de variables d'état et de commande :

$$\tilde{x} \longmapsto \xi \,, \quad u \longmapsto w$$

avec

$$\xi = \begin{pmatrix} \tilde{p} \\ \tan\tilde{\theta} \\ \frac{\tan\varphi}{L(\cos\tilde{\theta})^3} \end{pmatrix} = \begin{pmatrix} \tilde{p} \\ \tan\tilde{\theta} \\ \frac{\tan(\tilde{\varphi}+\varphi_r)}{L(\cos\tilde{\theta})^3} \end{pmatrix} \,, \quad w = \begin{pmatrix} u_1\cos\tilde{\theta} \\ u_2 \end{pmatrix}$$

a) Exprimer la dynamique du système en fonction de ces nouvelles variables d'état et de commande, i.e., $\dot{\xi} = \bar{f}(\xi, w)$.
b) Linéariser exactement la dynamique de $\xi_1 = \tilde{p}_1$ et synthétiser une commande w_1 qui stabilise exponentiellement ξ_1 à zéro.
c) En supposant, grâce à la question précédente, que ξ_1 et w_1 sont négligeables, montrer que le degré relatif de $y = \xi_2 = \tilde{p}_2$ est égal à 3 et en déduire une loi de commande qui stabilise $\tilde{p}_2$ et $\tilde{\theta}$ à zéro.
d) En revenant à la commande de départ u, préciser sous quelle condition cette commande sera bien définie, en particulier en fonction de l'erreur initiale entre l'orientation du robot $\theta(0)$ et la direction initiale $\theta_r(0)$ de la trajectoire à stabiliser. Donner une interprétation géométrique de ce résultat.
e) Quel est le domaine de convergence garanti par ce contrôleur ?

Validations numériques : Vous avez à disposition le modèle Simulink représenté par la figure ci-dessous. Ce modèle est constitué de trois parties pricipales :

1. Le bloc orange est simplement une fonction matlab qui définit les vitesses de la trajectoire de référence. Plus précisément, ce fichier génère des vitesse $(v_r = u_{r,1}, \omega_r)$ telles que

$$\left\{ \begin{array}{rcl} \dot{p}_r & = & v_r R(\theta_r) e_1 \\ \dot{\theta}_r & = & \omega_r \end{array} \right.$$

2. Le bloc bleu intègre les équations ci-dessus ainsi que les équations du modèle (1) de la voiture. Il permet donc d'obtenir en sortie à la fois p_r, θ_r, p, θ, et φ.

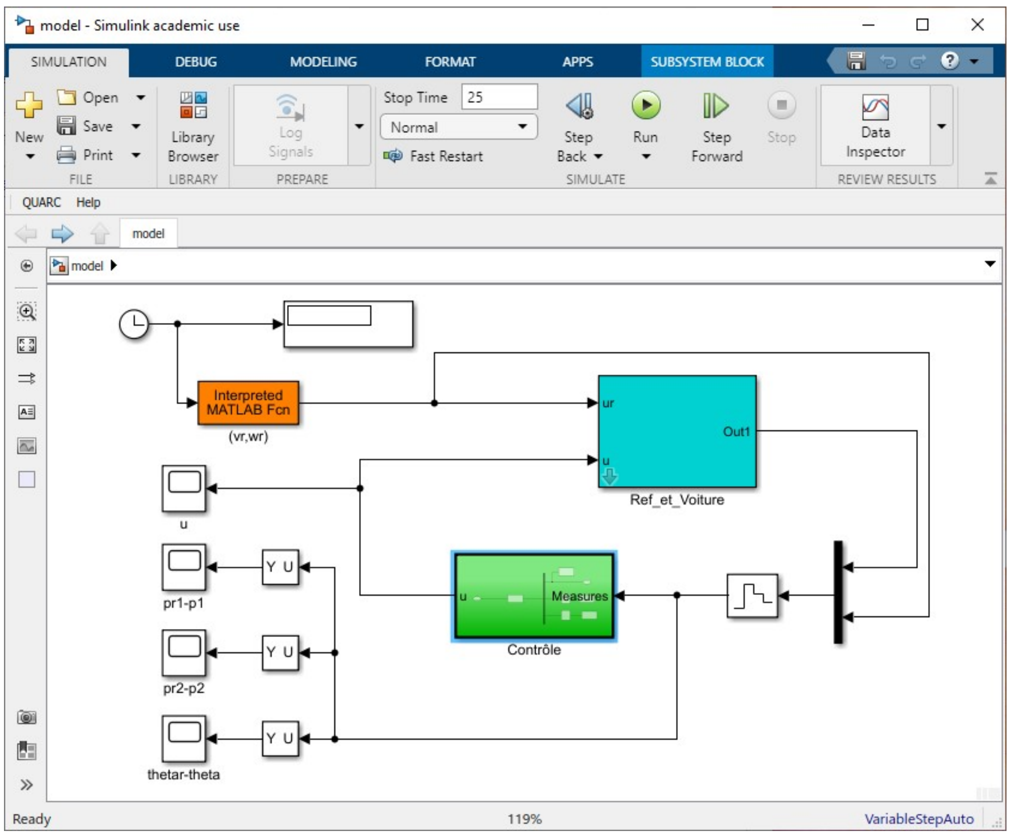

FIGURE 2 – Model simulink

3. Le bloc vert récupère ce vecteur de sortie, ainsi que v_r et ω_r, pour calculer la loi de commande dans le fichier "control.m".

Par ailleurs, vous disposez du script "anim.m" qui permet de faire l'animation du véhicule ainsi que du repère associé à la trajectoire de référence (repère blanc). Idéalement, le point P_0 du véhicule doit donc converger vers le centre du repère blanc, tout en étant aligné sur celui-ci.

Toutes les lois de commande développées dans les questions 4 et 5 doivent être validées sur ce simulateur. Vous devez discuter les performances de ces lois de commande, en fonction du choix des gains de commande. Vous discuterez également leur degré de robustesse vis-à-vis du fait que les vitesses v_r et ω_r ne permettent pas toujours de respecter les conditions $u_{r,1}$ constant et $u_{r,2} = 0$.

Remarque : On notera que le simulateur ne fournit pas directement la valeur de φ_r. On pourra, au choix, implémenter les commandes en prenant la valeur par défaut $\varphi_r = 0$, ou recalculer la valeur de φ_r à partir de la deuxième relation dans (2). Pour éviter la singularité lorsque $u_{r,1} = 0$, on pourra prendre une valeur approchée :

$$\varphi_r \approx \arctan\left(\frac{L u_{r,1} \dot{\theta}_r}{\varepsilon + u_{r,1}^2}\right)$$

avec $\varepsilon > 0$ une valeur "petite" (en quoi est-ce une bonne approximation ? ? ?).