TP LEGO Segway

Placement de pôles (PID numérique vs retour d'état)

La durée de ce TP est de 8 heures.
Ce TP est construit sur la base du Lego Segway proposé par Yorihisa Yamamoto
`[http://www.mathworks.com/matlabcentral/fileexchange/19147]`.

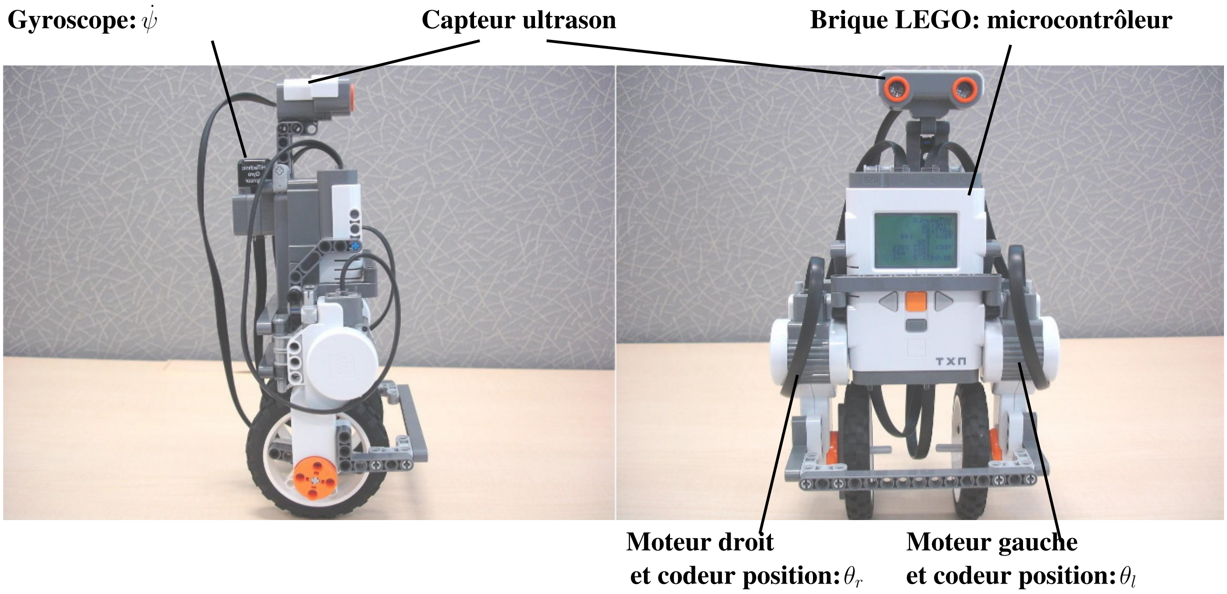

FIGURE 1 – Photo du lego Segway étudié

1 Introduction au TP

1.1 Objectif

L'objectif est de maintenir à l'équilibre le LEGO Segway (figure 1), qui est assimilable à un pendule inversé. Ce système a la particularité d'être naturellement instable en boucle ouverte.

On définit ψ, l'angle entre le corps du Segway et la verticale (figure 2) ; et θ, l'angle de rotation des roues par rapport au sol tel que la position au sol du Lego soit $x = R\theta$. L'asservissement consiste alors à commander U, la tension commune aux bornes des 2 moteurs afin de :

- maintenir le corps du Segway à l'équilibre (réguler à 0 l'angle ψ),
- asservir θ (ou $\dot{\theta}$) pour asservir la position au sol du Segway dans la direction $\vec{x}$.

On peut utiliser dans ce but :

- des correcteurs PID numériques avec une synthèse directe dans le domaine échantillonné, ou une synthèse à temps continu avant d'obtenir la version échantillonnée/à temps discret via la transformée bilinéaire.
- ou un retour d'état (cf. cours 1A) calculé sur le modèle échantillonné du système.

Dans ce TP, on utilise le logiciel Matlab pour la synthèse et Simulink pour la simulation des correcteurs et la génération du code C qui sera implanté dans la brique Lego.

1.2 Descriptif des entrées et sorties du système

L'entrée du système est la tension d'induit U commune aux 2 moteurs ($U = U_r = U_l$). Dans un premier temps, notre segway n'est pas prévu pour savoir tourner.

Entrée	Commande	Résolution	Plage
Tension d'induit du moteur U (Volt)	Vitesse du rotor	1% de la tension de la batterie ($\approx$ 8 Volts à pleine charge)	-100% $<<$ +100%

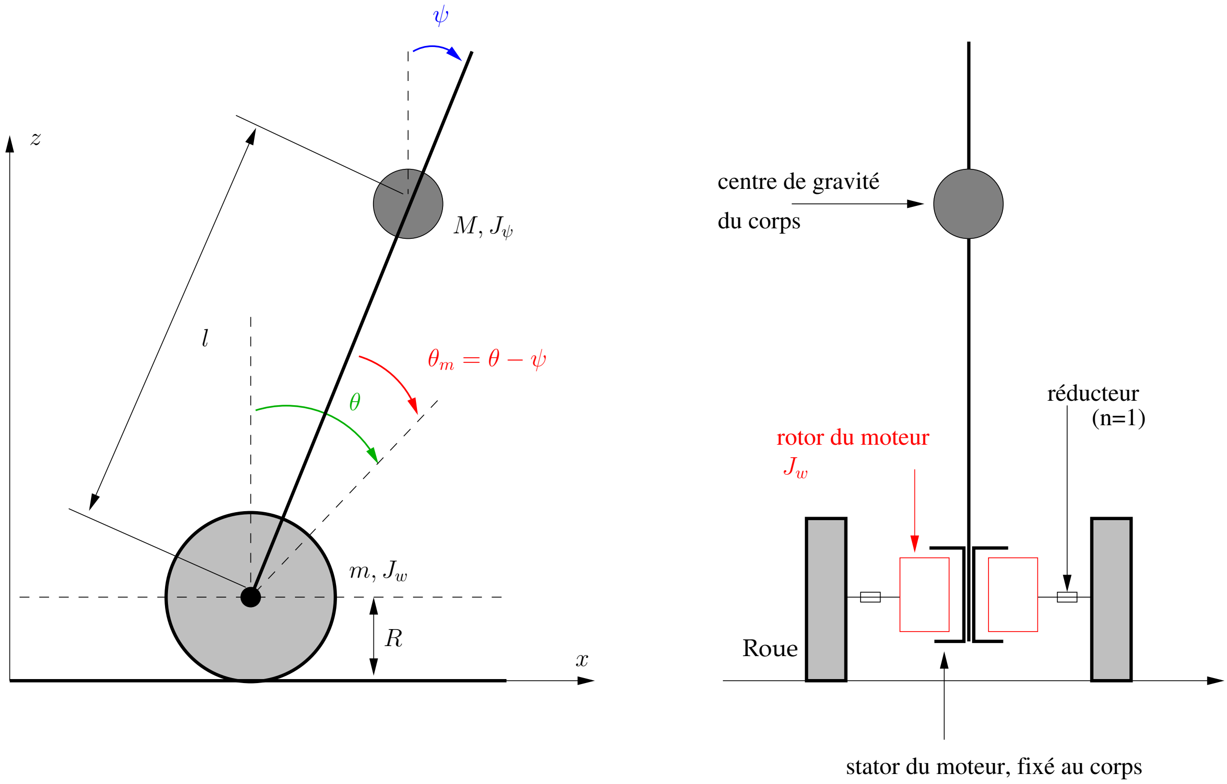

FIGURE 2 – Paramétrisation du Lego Segway

Capteur	Mesure	Résolution
Codeur incrémental moteur	$\theta_{m_{\{l,r\}}}$: angle entre roues {right/left} et corps	1 deg
Gyroscope	$\dot{\psi}$: vitesse angulaire du corps	1 deg/s, hors biais

Les 2 grandeurs ψ, angle entre le corps du robot et la vertical, et θ, angle moyen des roues par rapport au sol ne sont pas directement accessibles à la mesure via les capteurs disponibles. On peut cependant estimer la valeur de ces 2 grandeurs ($\hat{\psi}$ et $\hat{\theta}$) par les procédés suivants :

— intégration au cours du temps de $\dot{\psi}$ et suppression de l'offset par un filtrage passe-haut [1].

— $\hat{\theta} = \theta_m + \hat{\psi}$, avec $\theta_m = \frac{\theta_{m_r} + \theta_{m_l}}{2}$: angle moyen entre les roues et le corps, donné par les codeurs contenus dans les moteurs.

2 Préparation : Modèle dynamique du système

Les équations d'Euler-Lagrange appliquées au Segway donnent les équations dynamiques non linéaires suivantes pour son mouvement, en fonction des couples moteurs :

$$\left[(2m+M)R^2 + 2(J_w + J_m)\right]\ddot{\theta} + \left[MRl\cos(\psi) - 2J_m\right]\ddot{\psi} - MRl\sin(\psi)\dot{\psi}^2 = C_\theta \quad (1)$$

$$\left[Ml^2 + J_\psi + 2J_m\right]\ddot{\psi} + \left[MRl\cos(\psi) - 2J_m\right]\ddot{\theta} - Mgl\sin(\psi) = C_\psi \quad (2)$$

1. L'intégration sans précautions d'une vitesse n'est pas une bonne idée, car tout capteur analogique a un biais de mesure non nul (même si infiniment petit) qui a pour conséquence une divergence plus ou moins rapide de la mesure : $\int_0^t (\dot{x}(t) + \text{petit\_offset})dt = x(t) + \text{petit\_offset} \times t$.

avec

l : distance axe de la roue-centre de gravité du corps

m : masse de la roue
M : masse du corps (brique)
J_w : moment d'inertie d'une roue
J_ψ : moment d'inertie du corps
J_m : moment d'inertie du rotor d'un moteur
R : rayon de la roue
g : accélération de la gravité
C_θ : Couples appliqués aux roues
C_ψ : Couple appliqué au corps

Note : Ce modèle physique n'est valide que si les angles sont exprimés en radians et tension en volt.

2.1 Modèle linéaire du système en fonction de la tension moteur

Le calcul d'une commande pour stabiliser le Segway nécessite un modèle linéaire (le cours n'a abordé que ce type de système) et ayant pour entrée la tension moteur U.

1. Linéariser le système d'équations (1) et (2) autour de la position d'équilibre du système ($\psi = 0$) en fonction des grandeurs ψ, θ et leurs dérivées : utiliser un développement limité à l'ordre 1 des fonctions trigonométriques autour de 0 et négliger les derniers termes non linéaires.
2. On néglige le frottement des roues sur le sol (hypothèse de mouvement sans glissement). Les 2 moteurs sont situés entre les roues et le corps, on a alors l'égalité suivante ("principe d'action-réaction") :

$$C_\theta = -C_\psi \tag{3}$$

et

$$\begin{aligned} C_\theta &= 2.(C_m - f_m(\dot{\theta} - \dot{\psi})) & (4)\\ &= 2.(k_t i_m - f_m(\dot{\theta} - \dot{\psi})) & (5) \end{aligned}$$

avec
— C_m : le couple d'un moteur
— f_m : un coefficient de friction proportionnel à la vitesse de rotation du rotor
— k_t : la constante de couple du moteur
— i_m : l'intensité du courant dans l'induit du moteur

D'après le modèle du moteur à courant continu (voir section 2.1 du cours 1A), exprimer i_m en fonction de :
— k_b, la constante de la force contre-électromotrice
— U, la tension aux bornes de l'induit
— R_m, la résistance de l'induit (on néglige l'inductance du moteur)

Montrer alors que les équations (1) et (2), après leur linéarisation et l'expression des efforts en fonction de la tension d'induit U, peuvent s'écrire sous la forme :

$$\begin{aligned} E_{11}\ddot{\theta} + E_{12}\ddot{\psi} + 2\beta(\dot{\theta} - \dot{\psi}) &= 2\alpha U & (6)\\ E_{12}\ddot{\theta} + E_{22}\ddot{\psi} - 2\beta(\dot{\theta} - \dot{\psi}) - Mgl\psi &= -2\alpha U & (7) \end{aligned}$$

avec

$\alpha = \frac{k_t}{R_m}$ et $\beta = \frac{k_t k_b}{R_m} + f_m$
$E_{11} = (2m + M)R^2 + 2(J_w + J_m)$
$E_{12} = MRl - 2J_m$
$E_{22} = Ml^2 + J_\psi + 2J_m$

2.2 Modèle sous forme de représentation d'état

4. Écrire les équations (6) et (7) sous la forme matricielle suivante en précisant l'expression des matrices E, F, G et H. On rappelle que U est la tension commune des moteurs, soit un scalaire.

$$E \begin{bmatrix} \ddot{\theta} \\ \ddot{\psi} \end{bmatrix} + F \begin{bmatrix} \dot{\theta} \\ \dot{\psi} \end{bmatrix} + G \begin{bmatrix} \theta \\ \psi \end{bmatrix} = HU \tag{8}$$

5. Donner alors l'expression des matrices A_1 et B_1 du modèle d'état $\dot{X} = A_1X + B_1U$ du Segway (figure 3) en fonction de E, F,G, H, la matrice identité et la matrice nulle .

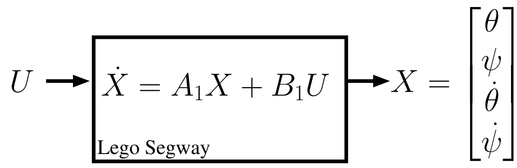

FIGURE 3 – Représentation d'état du système

Consignes pour le TP :

— Allumer le PC sous windows. Utiliser le compte local `groupeX-Y@psi` avec `X` votre numéro de groupe et `Y` le numéro de demi-groupe (`1` pour binômes 1 à 6, `2` pour binômes 6 à 13).
Le mot de passe est `Telxxxx!` où `xxxx` est l'année courante (Ex : `Tel2022!` pour l'année universitaire 2022-2023).

— Créer un répertoire sur le bureau portant vos noms. Tous vos fichiers Matlab devront être installés ou créés dans celui-ci.

— Répondre de manière concise et précise aux questions dans le fichier Matlab `.m`. Le cours 1A est fourni avec le reste des fichiers et peut quelquefois être utile.

3 TP, partie 0 : Construction du Lego Segway

3.1 Assemblage

1. Dans la page Moodle du module Automatisme et Robotique vous trouverez les fichiers du TP dont le plan de montage du lego nommé Balanc3r. Assemblez le Lego.
2. Une fois assemblé, **mettre la batterie du Lego EV3 en charge avec son chargeur**. Allumer la brique (bouton central-validation), tester le bon fonctionnement des codeurs incrémentaux, du gyroscope, et des moteurs avec les utilitaires `Port view` et `Motor control` disponibles dans les menus.
3. Éteindre la brique avec le bouton "retour-annulation" situé en haut à gauche.

4 TP, partie 1 : Du modèle à temps continu vers le temps discret

4.1 Analyse du modèle continu

1. Copier dans votre répertoire sur le bureau l'archive du TP disponible dans Moodle. Ouvrir le fichier `model.m` qu'elle contient. Complétez les paramètres physiques manquants : une balance est notamment à votre disposition.
2. Créer ensuite le modèle d'état du système :
   — Définir les matrices A_1 et B_1 du modèle d'état. Pour cela, il est recommandé de définir d'abord les matrices E,F,G et H en accord avec votre préparation.
   — Créer un système sous forme d'état (state-space) avec la commande `ss`, où l'on choisira pour le moment C_1 de manière à avoir pour sortie du système tous les états.
   Note : on peut associer des noms textuels aux états, entrées, sorties du système via les propriétés du modèle `ss` (voir commentaires du fichier .m) pour mieux s'y retrouver.
3. Quels sont les états d'équilibre de ce système[1] ?
4. Les équilibres sont-ils stables[2] ? Justifier.
   (a) en décrivant une expérience que vous pouvez réaliser avec le Lego Segway devant vous qui montre que l'équilibre est instable ;
   (b) en utilisant la fonction `pzmap` pour une justification mathématique. Quelles sont les valeurs des pôles retournées par `pzmap` (`help pzmap`) ?
   Autovalidation : On donne ici les valeurs approximatives des pôles obtenus (-370, -7, 0, 7).

4.2 Modèle à temps échantillonné du Segway

Le système à temps continu

$$\dot{X}(t) = A_1\,X(t) + B_1\,U(t) \tag{9}$$
$$y(t) = C_1 X(t) + D_1 U(t) \tag{10}$$

1. on considère X stationnaire et $U = 0$ dans la représentation d'état pour les calculer.
2. dépend de la stabilité des pôles.

a pour forme à temps discret avec une période d'échantillonnage T_s (en seconde !) :

$$X(k+1) = \Phi(T_s)\, X(k) + \Gamma(T_s)\, U(k) \tag{11}$$
$$y(k) = C_1\, X(k) + D_1\, U(k) \tag{12}$$

où $X(k) = X(t = k\, T_s)$ et $\{\Phi, \Gamma\}$ sont les matrices d'état et d'entrée du système à temps discret. Les coefficients de ces 2 matrices dépendent de la période d'échantillonnage choisie. Dans le cas d'une entrée $U(k)$ mise à jour à chaque pas d'échantillonnage et dont la valeur est bloquée entre 2 pas, on a la formule $\Phi(T_s) = e^{A_1\, T_s}$.

1. Utiliser la fonction `c2d(SYSC,TS,METHOD)` pour obtenir le modèle échantillonné à la période $T_s = 4\,\text{ms}$ du Segway en choisissant la méthode adéquate. Pour choisir la méthode adéquate (`doc c2d`), on sera conscient du fait que le convertisseur numérique-analogique des moteurs dans la brique conserve la tension de sortie demandée (comme tout CNA) tant qu'il ne reçoit pas une nouvelle consigne de tension.
2. Vérifier que la matrice d'état du modèle discret obtenue est en accord avec la formule donnée plus haut pour Φ (fonction Matlab `expm`). On accède aux matrices d'un modèle `my_model` sous forme `ss` avec `my_model.a` pour la matrice d'état A par exemple.
3. Si on analyse les pôles du système discret (avec `eig` ou `pzmap`), on constate qu'il existe une correspondance avec les pôles du système continu donnée par la relation :

$$z_p = e^{s_p\, T_s}$$

   où z_p représente le pôle discret et s_p le pôle continu.

   La limite entre le demi-plan stable et le demi-plan des pôles instables en temps continu est l'axe des imaginaires purs dans le plan complexe. D'après la relation précédente et l'équation paramétrique de l'axe des imaginaires purs ($s = jw$ avec $w : -\infty \to +\infty$), quelle est la limite de stabilité dans le plan complexe en temps discret ?
4. Avant de se lancer dans une correction/stabilisation par retour d'état, ce système discret qu'est le Segway à temps échantillonné est-il commandable (fonction `rank`) ?
   Le test de commandabilité est similaire en discret au cas continu en substituant les matrices $\{\Phi, \Gamma\}$ aux matrices $\{A, B\}$. Pourquoi est-il important de vérifier a priori cette propriété du système ?

5 TP, partie 2 : Stabilisation par retour d'état

Pour rappel, un retour d'état statique (figure 4) consiste à imposer une commande stabilisatrice au système de la forme :

$$U = -K\, x(k) \tag{13}$$

alors l'expression en boucle fermée du système est :

$$X(k+1) = \Phi\, X(k) + \Gamma\, U(k) \tag{14}$$
$$= \Phi\, X(k) - \Gamma\, K X(k) \tag{15}$$
$$X(k+1) = (\Phi - \Gamma\, K) X(k) \tag{16}$$

Il suffit alors pour stabiliser le système de choisir le vecteur de gain K tel que la matrice d'état $(\Phi - \Gamma\, K)$ a des valeurs propres (pôles z_p) stables.

1. Dans le cas où K vérifie la condition ci-dessus pour le système corrigé (i.e $|z_{p_i}| < 1$), vers quelle valeur $X_{eq}(k)$ converge $X(k)$ quand $k \to \infty$? La valeur finale X_{eq} est-elle effectivement un équilibre du Segway ?

   On peut s'inspirer du cas diagonal simple ci-dessous avec $X_{eq}(k) = [x_1, x_2]^T$ pour le raisonnement :

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \end{bmatrix} = \begin{bmatrix} z_{p1} & 0 \\ 0 & z_{p2} \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} \tag{17}$$

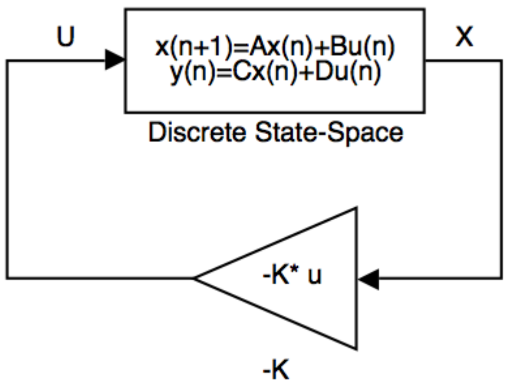

FIGURE 4 – Schéma bloc de principe du retour d'état : $u = -Kx$

Pour déterminer K dans le retour d'état, il faut choisir les pôles souhaités du système corrigé : c'est un placement de pôles.

On propose ici pour le choix des pôles du système bouclé :

— de conserver les 2 pôles stables et rapides présents dans le système discret non corrigé [2] ;

— et de choisir un pôle dominant réel ≥ 0 de multiplicité 2 pour les pôles restants : l'approche simple et classique.

2. Quel pôle appelle-t-on "dominant" parmi un ensemble de pôles : le plus lent ou le plus rapide ? Le dominant est celui que l'on ne peut pas négliger pour rendre compte de la dynamique du système.
   Avec la relation précédente $z_p = e^{s_p T_s}$ et pour se faire une idée des constantes de temps en jeu avec une période d'échantillonnage $T_s = 4$ms, quelles constantes de temps sont associées aux 2 pôles stables du Segway à temps discret ? En discret, un pôle lent est-il proche de 0 ou du cercle unité ?
3. Utiliser alors la formule d'Ackermann (voir aide de la fonction `acker`) pour calculer le gain K du retour d'état permettant d'imposer les pôles souhaités au système bouclé. Pour le choix du pôle dominant multiple, on prendra soin à trouver un compromis entre un pôle trop rapide (qui demande une tension de commande trop grande [3]) et un pôle trop lent (l'état du système s'éloigne trop du point d'équilibre et l'approximation modèle linéaire n'est plus valide).

   Pour chercher une valeur optimale de ce pôle, il vous faut implémenter et simuler le retour d'état Fig. 4 via Simulink avec un gain et un bloc state-space discret de période T_s. Pour des vitesses nulles et une position angulaire ψ initiale limite de 10 degrés (à définir dans les bonnes unités dans le vecteur d'état initial du bloc state-space), la commande doit rester compatible avec la charge maximale de la batterie et ne doit donc pas dépasser les 10 volts.
   Vérifier sur les courbes que le système est bien stabilisé à l'équilibre.

   Rappel : On peut utiliser toute variable Matlab définie dans l'espace de travail comme paramètre d'un bloc Simulink.

[Point de validation : Faire valider par un encadrant.]
[Déposer vos fichiers .m et simulink (.slx) dans le dépot Moodle séance 1]

2. Une alternative consiste à conserver uniquement le pôle très rapide et choisir un pôle triple pour les pôles restants.
3. En effet, rien en simulation n'empêche de choisir les pôles très rapides pour une réponse instantanée du système (1 femto seconde) mais ce temps de réponse est alors associé à une commande du système physiquement irréaliste (10^{13} volts).