

Département Electronique, Energie Electrique, Automatique  
1ère année de Master ISTR  
Ingénierie des Systèmes Temps Réel

---

## ÉTUDE ET CONTRÔLE DE LA MAQUETTE “BILLE SUR RAIL” DE QUANSER

---

*Etudiants :*  
ABERBOUR Oualid  
SEGHIER Aissa  
RAHMOUN Lokmane

*Encadrant :*  
FERGANI Soheib

Module Initiation à la Recherche et Projet  
Année 2023/2024

# Remerciements

---

Nous souhaitons exprimer nos sincères remerciements à toute l'équipe pédagogique de l'université Paul Sabatier, ainsi qu'aux intervenants professionnels qui ont contribué à notre formation et à notre développement académique. Nous adressons nos remerciements particuliers à notre encadrant Monsieur FERGANI Soheib, pour ses précieux conseils et son accompagnement tout au long de la réalisation de ce projet.

Nous sommes également reconnaissants envers nos parents pour leur soutien inébranlable, leurs encouragements constants et leur confiance en nous durant cette période d'études.

Par ailleurs, nous tenons à remercier tous ceux qui ont contribué de près ou de loin à l'avancement de notre travail de recherche, ainsi que nos amis et collègues pour leurs échanges enrichissants et leur soutien moral.

Enfin, nos plus sincères remerciements vont à toutes les personnes qui ont partagé leur expertise, leur temps et leurs ressources pour rendre ce travail possible. Leurs contributions ont été d'une valeur inestimable pour la réussite de ce projet.

---

# Résumé

Le système bill sur rail constitue un exemple emblématique en théorie de contrôle, caractérisé par sa simplicité et son instabilité en boucle ouverte. Son objectif principal réside dans le contrôle de la position de la bille pour la maintenir à une localisation prédéfinie. Utilisé couramment comme banc d'essai en laboratoire, ce système présente des défis majeurs en raison de sa nature intrinsèquement instable et non linéaire.

L'application de ce système va au-delà de sa simple utilisation expérimentale ; elle s'étend à des domaines tels que l'équilibrage dynamique de la bille sur le rail, offrant ainsi une analogie précieuse avec des situations réelles telles que la stabilisation d'un avion lors de l'atterrissage ou dans des conditions de turbulence aérienne.

Pour relever les défis posés par ce système, une variété d'algorithmes de contrôle sont explorés et testés, allant des approches classiques aux méthodes plus avancées telles que le contrôle non linéaire. Comprendre et maîtriser ce système offre des perspectives significatives dans le développement de nouvelles technologies de contrôle, tout en fournissant des connaissances précieuses pour résoudre des problèmes de stabilité et de contrôle dans des contextes complexes et dynamiques.

L'objectif principal de cette étude est de stabiliser la bille à une position désirée. Pour ce faire, nous passons par plusieurs étapes, allant de l'état de l'art à la modélisation mathématique du système. Cette modélisation est ensuite utilisée pour la commande fréquentielle, la synthèse du retour d'état et la conception d'un observateur d'état.

# Table des matières

<b>1 État de l'art</b>	<b>2</b>
1.1 Introduction . . . . .	2
1.2 Les composants de la maquette . . . . .	3
1.3 La carte d'acquisition . . . . .	3
1.4 Les paramètres de la maquette . . . . .	4
1.5 Conclusion : . . . . .	4
<b>2 Modélisation</b>	<b>5</b>
2.1 Introduction . . . . .	5
2.2 Hypothèses de travail . . . . .	5
2.3 Fonction de transfert du système la bille et le rail . . . . .	5
2.4 Equation du mouvement . . . . .	7
2.5 Validité de l'approximation . . . . .	10
2.6 Modèle non linéaire . . . . .	11
2.7 Point d'équilibre . . . . .	12
2.8 Linéarisation . . . . .	13
2.8.1 Commandabilité . . . . .	14
2.8.2 Observabilité . . . . .	14
2.9 Conclusion . . . . .	14
<b>3 Commande du système et synthèse d'un observateur</b>	<b>15</b>
3.1 Commande fréquentielle : . . . . .	15
3.1.1 Commande du servomoteur : . . . . .	16
3.1.2 Commande du système complet : . . . . .	20
3.2 Commande par retour d'état : . . . . .	25
3.2.1 Calcul de la loi de commande . . . . .	25
3.2.2 Simulation de la commande par retour d'état . . . . .	25
3.2.3 Résultats de l'implémentation sur la maquette . . . . .	26
3.3 Synthèse d'un Observateur . . . . .	27
3.3.1 Modélisation de l'observateur . . . . .	27
3.3.2 Validation par simulation . . . . .	28
3.4 Conclusion : . . . . .	29

# Table des figures

1.1	Maquette du système de bille sur rail de Quanser . . . . .	3
1.2	La carte d'acquisition Q2-USB . . . . .	3
2.1	Boucle ouverte du système $G(p)$ . . . . .	5
2.2	Représentation du système la bille et le rail . . . . .	6
2.3	Moteur à courant continu commandé par armature . . . . .	8
2.4	Évolution de $\sin(x)$ par rapport à $x$ . . . . .	11
3.1	Réponse indicielle du servomoteur en boucle fermée . . . . .	16
3.2	Lieu des racines de $F_{sm}$ . . . . .	16
3.3	Schéma block de l'asservissement du servomoteur . . . . .	17
3.4	Réponse indicielle du servomotor commandé pour différentes valeurs de $K_{c_1}$ . . . . .	19
3.5	Réponse indicielle du servomotor commandé pour différentes valeurs de $z_1$ . . . . .	19
3.6	Réponse de l'angle du servomoteur . . . . .	20
3.7	Lieu des racines de $G_{br}$ . . . . .	21
3.8	Schéma bloc de l'asservissement du système complet . . . . .	21
3.9	Schéma bloc de l'asservissement du système complet avec $\theta_i(t) = \theta_o(t)$ . . . . .	22
3.10	Réponse indicielle du système commandé pour différentes valeurs de $K_c$ . . . . .	23
3.11	Réponse indicielle du système commandé pour différentes valeurs de $z$ . . . . .	23
3.12	Réponse de la position de la bille . . . . .	24
3.13	Agrandissement sur la réponse de la position de la bille . . . . .	24
3.14	Réponse de la position de la bille . . . . .	26
3.15	Réponse de la position de la bille . . . . .	26
3.16	Erreurs de convergence des états estimés . . . . .	29
3.17	Schéma de l'implémentation de la commande fréquentielle du servomoteur . . . . .	I
3.18	Schéma de l'implémentation de la commande fréquentielle du système complet . . . . .	II
3.19	Schéma de l'implémentation de la commande par retour d'état . . . . .	III

# Introduction générale

Avec l'évolution récente de la science et de la technologie, de nombreux problèmes de commande ont émergé dans le monde réel, nécessitant des solutions spécifiques et adaptées. La commande de ces systèmes n'est pas toujours évidente, car ces problèmes réels sont parfois inaccessibles en laboratoire. Cela conduit à la création de systèmes équivalents pour les étudier. Parmi ces systèmes figure la maquette "bille sur rail" de Quanser [2].

Le système "bille sur rail" constitue un modèle utilisé dans l'enseignement de l'ingénierie des systèmes de contrôle, en raison de sa simplicité à explorer la dynamique des systèmes instables. Les théories développées pour résoudre les défis de commande liés à ce système trouvent également des applications concrètes dans d'autres domaines, telles que la stabilisation horizontale d'un avion lors de l'atterrissement et du décollage, le flux d'air turbulent et le contrôle aérospatial[1].

De nombreuses méthodes de contrôle classiques et modernes ont été employées pour stabiliser ce système [12][5]. Ce système comprend un rail inclinable et une bille se déplace le long de ce rail, avec sa position mesurée par un transducteur linéaire qui produit un signal de tension proportionnel. Le contrôleur régule la position de la bille en ajustant l'angle du rail à l'aide du bras de levier relié au servomoteur.

Ce rapport a été rédigé dans le cadre de la formation en Master Ingénierie des Systèmes Temps Réel à l'Université Toulouse III - Paul Sabatier. Il est structuré en trois chapitres distincts.

Le premier chapitre présente en détail la maquette, ses composants, ses paramètres ainsi que la carte d'acquisition utilisée. Une revue de l'état de l'art est également incluse, offrant un panorama synthétique et organisé des travaux existants relatifs à cette maquette.

Le deuxième chapitre, intitulé "Modélisation", établit un modèle mathématique décrivant la dynamique du système. Ce modèle commence par les équations de mouvement, aboutissant à un modèle non linéaire qui est ensuite linéarisé. La fonction de transfert du système ainsi que sa représentation par modèle d'état sont obtenues à cette étape. Cette modélisation est une étape cruciale pour le chapitre suivant, dédié à la "Commande".

Le troisième chapitre est consacré à la commande, structuré en deux parties principales : la commande par retour d'état et la commande fréquentielle, où un correcteur PD est utilisé. Chaque partie expose les étapes de calcul des lois de commande, les résultats de simulation et les résultats d'implémentation sur la maquette. Le chapitre se termine par la présentation d'un observateur d'état, enrichi par des résultats issus de simulations. Malheureusement, le manque de temps n'a pas permis de procéder aux essais pratiques sur la maquette.

Enfin, le rapport se termine par une conclusion générale résumant les résultats du travail ainsi que les perspectives et les travaux futurs.

# Chapitre 1

## État de l'art

### 1.1 Introduction

Le système "Bille sur rail" est constitué d'un rail long inclinable par un servomoteur électrique. Une bille peut rouler le long de ce rail, et sa position est mesurée par un transducteur linéaire qui émet un signal de tension proportionnel à sa position. Un bras de levier, attaché à un côté du rail, est couplé à un servomoteur pour ajuster l'angle du rail et stabiliser la bille à une position désirée. Ce système permet un contrôle précis de la position de la bille en ajustant l'angle du rail grâce au servomoteur.[8]

Ce système permet un contrôle précis de la position de la bille en ajustant l'angle du rail grâce au servomoteur. Il a une propriété très importante : en boucle ouverte, il est instable car la sortie du système (la position de la bille) augmente sans limite pour une entrée fixe (l'angle du rail). Le travail de contrôle consiste à réguler automatiquement la position de la bille en modifiant la position du servomoteur. Cependant, cette tâche de contrôle est difficile car la bille ne reste pas en place sur le rail lorsque ( $\alpha \neq 0$ ), mais se déplace avec une accélération proportionnelle à l'inclinaison du rail.[12]

En raison de la non-linéarité et de la complexité du modèle du système, certains chercheurs ont opté pour des stratégies de commande sans modèle telles que les réseaux de neurones[11], la logique floue [6], et les régulateurs PID [10].

Les techniques de commande basées sur un modèle peuvent être catégorisées en deux catégories. La première considère le couple [12] ou l'accélération [7] comme une entrée, tandis que la deuxième utilise la tension du moteur [10] comme une entrée. Dans [12], l'auteur a conçu un algorithme de contrôle en supposant que la dynamique du système se compose de deux sous-systèmes : le rail et le servomoteur. Il a négligé l'effet du couplage dans les équations dynamiques du rail, du moteur et de la vitesse angulaire du rail. L'auteur a considéré le couple du moteur comme l'entrée du sous-système du rail. Dans [7], les auteurs proposent un Régulateur Linéaire Quadratique (LQR).

Dans [3], une stratégie de commande utilisant une paire de contrôleurs de mode glissant flous découplés (DFSMCs) a été conçue. D'autres techniques de commande existent dans la littérature. Dans ce rapport, nous allons utiliser la commande par retour d'état et la commande par un régulateur Proportionnel Dérivé.

## 1.2 Les composants de la maquette

La figure suivante présente les composants de la maquette. Ce travail ne se concentre pas sur les détails de câblage, ni l'alimentation, tous ces aspects étant décrits en détail dans le catalogue fourni par Quanser.

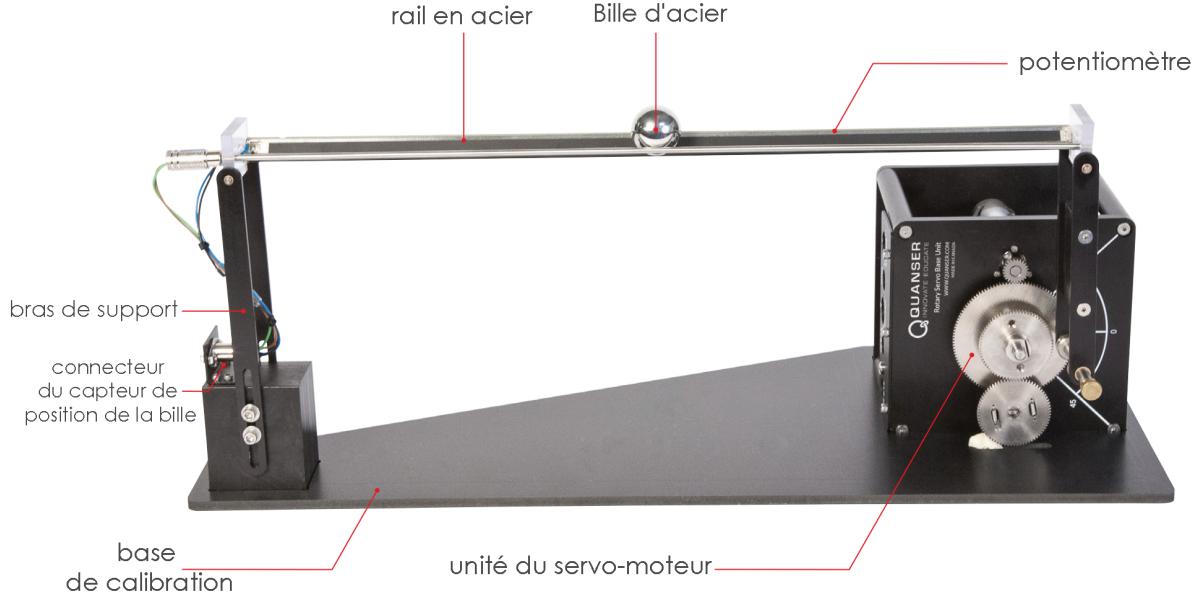


FIGURE 1.1 – Maquette du système de bille sur rail de Quanser

## 1.3 La carte d'acquisition

Pour cette manipulation, nous avons utilisé la carte d'acquisition "Q2-USB" de Quanser. Les informations cruciales pour cette opération concernent les fréquences maximales de cette carte, que nous devons prendre en compte lors de la manipulation. Cette carte a une fréquence maximale de signal PWM généré de 40 MHz, ainsi qu'une fréquence maximale de signal d'entrée dans l'encodeur de 10 MHz.[9]



FIGURE 1.2 – La carte d'acquisition Q2-USB

## 1.4 Les paramètres de la maquette

Le tableau ci-dessous décrit les principaux paramètres associés à la maquette "bille sur rail". Pour une meilleure compréhension de ces paramètres, veuillez vous référer à la Figure 2.2. Certains de ces paramètres sont également présents dans le modèle mathématique du système.

Symbol	Description	Valeur
L	Longueur du rail	42.55 cm
	Longueur du bras de levier	12.00 cm
r	Distance entre l'arbre d'engrenage de sortie servo et le joint couplé	2.54 cm
	Longueur du bras de support	16.0 cm
$r_b$	Rayon de bille	1.27 cm
$m_b$	Masse de bille	0.064 kg

TABLE 1.1 – Les parameters de la maquette[8]

## 1.5 Conclusion :

Une fois que la description de la maquette a été réalisée, ainsi que l'identification des différents composants constituant le système de la bille sur rail, la prochaine étape consiste à établir un modèle mathématique décrivant la dynamique de notre système. Cela nous permettra ensuite d'effectuer une étude et une analyse du modèle en utilisant les différents concepts de l'automatique et du contrôle, dans le but d'asservir le système.

# Chapitre 2

## Modélisation

### 2.1 Introduction

La modélisation mathématique de tout système fournit des moyens pour l'étude scientifique et la compréhension de son fonctionnement et de son comportement. Le système de la bille sur rail peut être divisé en deux sous-systèmes, où le fonctionnement de chaque sous-système contribue de manière intégrée au mouvement global de la bille, apparaissant ainsi comme un mouvement en douceur sur le rail, dépendant de l'inclinaison du rail et vice-versa. D'un point de vue mathématique, il est important de comprendre la contribution et la coordination de chacun des sous-systèmes. En ce qui concerne le contrôle, il est essentiel de développer l'équation du mouvement et la fonction de transfert du système.[4]

### 2.2 Hypothèses de travail

Afin de simplifier les équations de notre modélisation, nous allons négliger le frottement de la bille avec le rail (frottement sec), ainsi que le frottement de la bille avec l'air. Cependant, nous conserverons le frottement visqueux. Dans ce cas, il nous reste deux forces agissant sur notre bille : la force causée par la translation et la force causée par la rotation. Dans notre étude, nous considérerons que la bille roule le long du rail.

### 2.3 Fonction de transfert du système la bille et le rail

La fonction de transfert de la boucle ouverte du système  $G(p)$  peut être obtenue en mettant en cascade la fonction de transfert du "servo moteur"  $F_{sm}(p)$  avec celle de "la bille et le rail"  $G_{br}(p)$ , comme illustré dans la figure 2.1 :

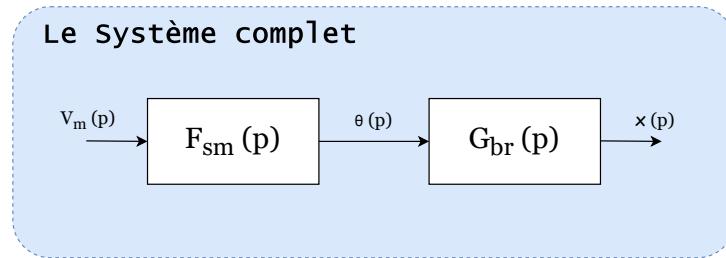


FIGURE 2.1 – Boucle ouverte du système  $G(p)$ .

On peut exprimer la fonction de transfert de notre système complet par l'expression suivante :

$$G(p) = F_{sm}(p) \cdot G_{br}(p) \quad (2.1)$$

La figure 2.2 présente la position de la bille à différents instants, en tenant compte de l'inclinaison du rail le long de l'axe  $\vec{x}$ . Le déplacement de la bille se compose à la fois d'une rotation et d'une translation :

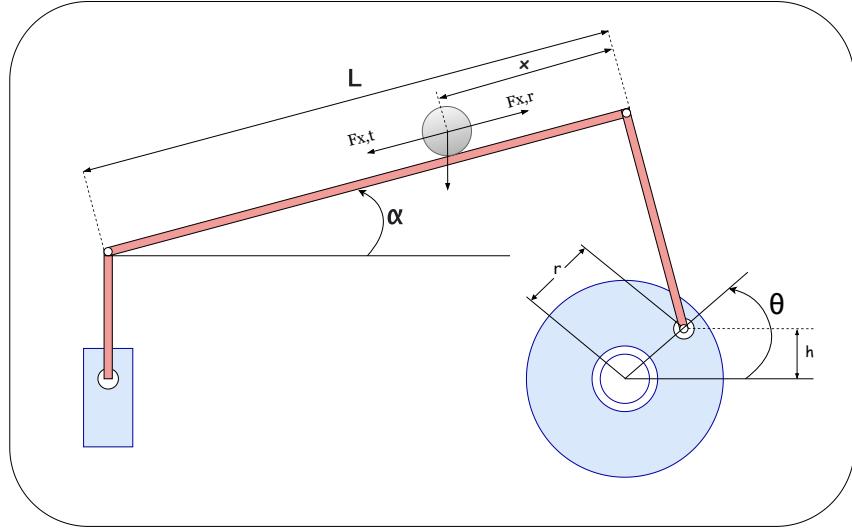


FIGURE 2.2 – Représentation du système la bille et le rail

Avec : 
$$\begin{cases} x & : \text{déplacement de la bille.} \\ \theta & : \text{angle du servo-moteur.} \\ \gamma_b & : \text{angle de la bille.} \\ \alpha & : \text{angle de rail} \end{cases}$$

La force de translation générée par la gravité est donné par :

$$F(x, t) = m_b \cdot g \cdot \sin(\alpha(t)) \quad (2.2)$$

Avec  $m_b$  la masse de la bille et  $g$  la force de la gravité.

La force de l'inertie de la bille causée par la rotation est :

$$F(x, r) = \frac{\tau_b}{r_b} \quad (2.3)$$

Où  $r_b$  représente le rayon de la bille,  $\tau_b$  le couple, et  $J_b$  le moment d'inertie.

Ainsi, le couple peut être exprimé en fonction du moment d'inertie et de l'angle de la bille :

$$\tau_b = J_b \cdot \frac{d^2\gamma_b(t)}{dt^2} \quad (2.4)$$

En utilisant la formule sectorielle qui nous permet de convertir le déplacement angulaire vers un déplacement linéaire nous obtenons :

$$x(t) = \gamma_b(t) \cdot r_b \quad (2.5)$$

En remplaçant donc dans l'équation (2.3), la force de l'inertie peut donc être représenté comme suit :

$$F(x, r) = \frac{1}{r_b} \cdot J_b \cdot \frac{d^2}{dt^2} \left( \frac{x(t)}{r_b} \right) \quad (2.6)$$

Ou bien :

$$F(x, r) = \frac{J_b}{r_b^2} \cdot \ddot{x} \quad (2.7)$$

Le moment d'inertie de la bille  $J_b$  est donné par :

$$J_b = \frac{2}{5} m_b \cdot r_b^2 \quad (2.8)$$

En utilisant la 2ème loi de Newton :

$$\sum \vec{F} = m_b \cdot \vec{\ddot{x}} \quad (2.9)$$

$$\vec{F}(x, r) + \vec{F}(x, t) = m_b \cdot \vec{\ddot{x}} \quad (2.10)$$

En effectuant une projection sur l'axe  $\vec{x}$  nous obtenons :

$$-F(x, r) + F(x, t) = m_b \cdot \ddot{x} \quad (2.11)$$

En remplaçant par les formules trouvées précédemment :

$$m \cdot g \cdot \sin(\alpha(t)) - \frac{J_b}{r_b^2} \cdot \ddot{x} = m_b \ddot{x} \quad (2.12)$$

Ainsi :

$$(m_b + \frac{J_b}{r_b^2}) \ddot{x} - m \cdot g \cdot \sin(\alpha(t)) = 0 \quad (2.13)$$

L'équation physique de notre système est donc :

$$\ddot{x}(t) = \frac{m_b \cdot g \cdot r_b^2}{m_b \cdot r_b^2 + J_b} \cdot \sin(\alpha(t)) \quad (2.14)$$

## 2.4 Equation du mouvement

D'après le schéma de la Figure 2.2, nous pouvons constater que :

$$\sin(\alpha(t)) = \frac{h}{L} \quad (2.15)$$

$$\sin(\theta(t)) = \frac{h}{r} \quad (2.16)$$

Par la suite, nous pouvons donc en déduire la relation statique entre l'angle du servomoteur et l'angle d'inclinaison du rail :

$$\sin(\alpha(t)) = \frac{r}{L} \cdot \sin(\theta(t)) \quad (2.17)$$

En remplaçant l'équation (2.17) dans l'équation (2.14), nous obtenons :

$$\ddot{x}(t) = \frac{r \cdot m_b \cdot g \cdot r_b^2}{L \cdot (m_b \cdot r_b^2 + J_b)} \cdot \sin(\theta(t)) \quad (2.18)$$

Pour des petits angles de  $\theta$  et  $\alpha$ , nous allons supposer les approximations suivantes :

$$\sin(\theta(t)) \approx \theta(t) \quad (2.19)$$

$$\sin(\alpha(t)) \approx \alpha(t) \quad (2.20)$$

Ainsi :

$$\alpha(t) \approx \frac{r}{L} \cdot \theta(t) \quad (2.21)$$

Donc, nous obtenons une équation linéaire qui représente l'accélération de la bille :

$$\ddot{x}(t) = \frac{r \cdot m_b \cdot g \cdot r_b^2}{L \cdot (m_b \cdot r_b^2 + J_b)} \cdot \theta(t) \quad (2.22)$$

$$(2.23)$$

Ensuite, il est possible de remplacer le moment d'inertie de la bille par son équation (2.8) trouvée précédemment, afin d'obtenir la nouvelle équation suivante d'accélération :

$$\ddot{x}(t) = K_{bb} \cdot \theta(t) \quad (2.24)$$

Avec :

$$K_{bb} = \frac{r \cdot m_b \cdot g \cdot r_b^2}{L \cdot (m_b \cdot r_b^2 + J_b)} = \frac{5 \cdot g \cdot r}{7 \cdot L} \quad (2.25)$$

Ensuite, en utilisant la transformation de Laplace :

$$\xrightarrow{\text{TL}} p^2 \cdot x(p) = \frac{r \cdot m_b \cdot g \cdot r_b^2}{L \cdot (m_b \cdot r_b^2 + J_b)} \cdot \theta(p) \quad (2.26)$$

La fonction de transfert entre l'angle du servomoteur et la position de la bille est donc :

$$G_{br}(p) = \boxed{\frac{x(p)}{\theta(p)}} = \frac{r \cdot m_b \cdot g \cdot r_b^2}{L \cdot (m_b \cdot r_b^2 + J_b)} \cdot \frac{1}{p^2} \quad (2.27)$$

Le schéma de la Figure 2.3 représente le schéma d'un moteur à courant continu, sur lequel on va se baser pour trouver le modèle de la fonction de transfert :

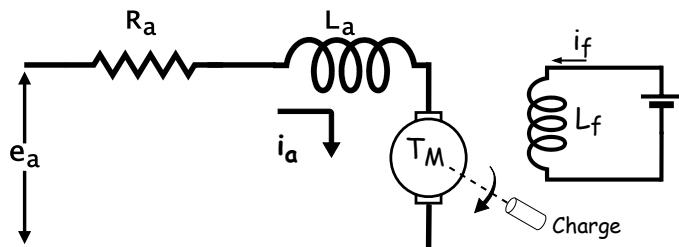


FIGURE 2.3 – Moteur à courant continu commandé par armature

Avec :	$R_a$	résistance induite (rotor-armature)
	$L_a$	inductance induite
	$f$	frottement fluides ( $\text{N} \cdot \text{s} \cdot \text{rad}^{-1}$ )
	$T_m$	couple du moteur
	$\theta$	angle du déplacement du moteur
	$J = J_m + J_L$	moment d'inertie équivalent (moteur+charge)
	$e_a$	la tension appliquée par le rotor
	$i_a$	courant traversant

Le couple du moteur est donnée par :

$$T_m = k_T \cdot \dot{\theta} \quad (2.28)$$

Où  $K_t$  représente la constante électromagnétique du moteur en ( $\frac{vs}{\text{rad}}$ ).

En appliquant la loi de Kirchhoff sur le circuit du moteur, nous obtenons :

$$L_a \cdot \frac{di_a}{dt} + R_a \cdot i_a + e_b = e_a \quad (2.29)$$

Avec  $e_b$  la force électromotrice d'induit, celle-ci est exprimé en fonction de la constante de force électromotrice  $K_b$  et de l'angle du servomoteur :

$$e_b = K_b \cdot \frac{d\theta}{dt} \quad (2.30)$$

Ainsi, l'équation du couple est trouvée :

$$J_a \cdot \frac{d^2\theta}{dt^2} + f_0 \cdot \frac{d\theta}{dt} = T_m = K_T \cdot i_a \quad (2.31)$$

En calculant les transformées de Laplace des équations précédentes, et considérons les conditions initiales nulles :

$$L_a \cdot p \cdot i_a(p) + R_a \cdot i_a(p) + E_b(p) = E_a(p) \quad (2.32)$$

L'équation (2.32) peut être simplifié et représenté comme suit :

$$(L_a \cdot p + R_a) \cdot i_a(p) = E_a(p) - E_b(p) \quad (2.33)$$

L'expression de la force électromotrice en transformée de Laplace est donnée par :

$$E_b(p) = K_b \cdot p \cdot \theta(p) \quad (2.34)$$

En appliquant la transformée de Laplace à l'équation du couple, on obtient :

$$(J \cdot p^2 + f_0) \cdot \theta(p) = K_t \cdot i_a(p) \quad (2.35)$$

A partir de ces équations, le système peut être représenté sous la forme :

$$F_{sm}(p) = \frac{FTchainedirecte}{1 + FTboucleouverte} = \frac{\theta(p)}{E_a(p)} \quad (2.36)$$

Avec  $FT_{chainedirecte}$  représente la fonction de transfert qui relie l'entrée appliquée sur notre système (tension  $E_a$ ) et la sortie du système (mouvement angulaire du servomoteur  $\theta$ ) et lorsque il n'y a pas de mécanisme de régulation ou de boucle de rétroaction dans le système, la fonction de transfert en boucle ouverte  $FT_{boucleouverte}$  est égale à la fonction de transfert de la chaîne directe.

$$F_{sm}(p) = \frac{\frac{K_T}{p \cdot (J_p + f_0) \cdot (L_a \cdot p + R_a)}}{1 + \frac{K_T \cdot K_b}{p \cdot (J_p + f_0) \cdot (L_a \cdot p + R_a)}} \quad (2.37)$$

$$F_{sm}(p) = \frac{K_T}{J \cdot L_a \cdot p^3 + (R_a \cdot J + f_0 \cdot L_a) \cdot p^2 + f_0 \cdot R_a \cdot p + K_T \cdot K_b \cdot p} \quad (2.38)$$

Pour une valeur de  $L_a$  qui est très petite, l'expression de  $F_{sm}$  peut être simplifiée. En supposant que  $L_a$  tend vers zéro, certaines des composantes de l'équation deviennent négligeables :

$$F_{sm}(p) = \frac{K_T}{p \cdot (R_a \cdot p + f_0 \cdot p + K_T \cdot K_b)} \quad (2.39)$$

La fonction de transfert du servo-moteur est donc de la forme suivante :

$$F_{sm}(p) = \frac{\theta(p)}{V_m(p)} = \frac{K_m}{p \cdot (1 + \tau p)} \quad (2.40)$$

Avec :

$$K_m = \frac{K_T}{R_a \cdot f}, \quad \tau = \frac{J}{f}, \quad f = f_0 + \frac{K_t \cdot K_B}{R_a}$$

Finalement, la fonction de transfert du système global, décrivant le rapport entre l'entrée (la tension appliquée) et la sortie (la position de la bille), peut être exprimée comme suit :

$$G(p) = F_{sm}(p) \cdot G_{br}(p) \quad (2.41)$$

Donc :

$$G(p) = \frac{K_m}{p \cdot (1 + \tau p)} \cdot \frac{K_{br}}{p^2} = \frac{K_m \cdot K_{br}}{p^3 \cdot (1 + \tau \cdot p)} \quad (2.42)$$

## 2.5 Validité de l'approximation

Dans la section précédente, nous avons effectué une approximation de la fonction sinus pour de petits angles afin d'obtenir un modèle linéaire respectant les deux propriétés d'homogénéité et de superposition.

Cette approximation va sûrement introduire des incertitudes dans le modèle obtenu, et par conséquent, un léger décalage sera remarqué entre le modèle théorique et le modèle réel (la maquette). Cela peut avoir un effet significatif lors de la commande, où l'objectif est d'asservir le système, pouvant ainsi conduire à des erreurs et à une non-fonctionnalité de la commande sur le dispositif réel.

Ainsi, les deux graphiques de la Figure 2.4 correspondent à une comparaison entre  $x$  et  $\sin(x)$  pour un intervalle de  $[-\pi/4, \pi/4]$ . Puisque l'angle du servomoteur de la maquette est compris entre ces valeurs, nous avons donc choisi cet intervalle.

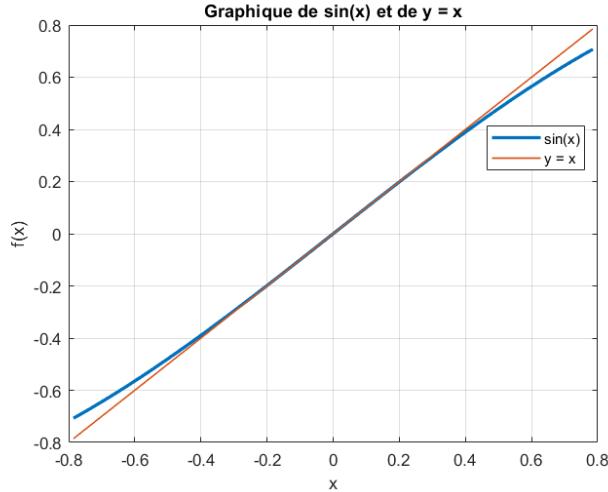


FIGURE 2.4 – Évolution de  $\sin(x)$  par rapport à  $x$

Donc, nous pouvons conclure que l'approximation est bien valide et que la relation est bien linéaire sur l'intervalle donné, avec un léger décalage pas très important vers les extrémités de l'intervalle.

Par conséquent, cette approximation est tolérable et n'aura pas un effet majeur, ce qui nous permet de continuer notre démarche en nous basant sur ce modèle obtenu grâce à cette approximation.

## 2.6 Modèle non linéaire

En se basant sur l'équation d'accélération obtenue précédemment, nous commençons notre démarche pour déterminer notre modèle non linéaire :

$$\ddot{x}(t) = K_{bb} \cdot \sin(\theta(t)) \quad (2.43)$$

Nous prenons donc le vecteur d'état  $z(t)$ , qui est composé des variables  $x$  et  $\dot{x}$  décrivant respectivement la position et la vitesse de déplacement, ainsi que  $\alpha$  et  $\dot{\alpha}$  représentant l'angle de rotation et sa vitesse angulaire :

$$z(t) = \begin{pmatrix} z_1(t) \\ z_2(t) \\ z_3(t) \\ z_4(t) \end{pmatrix} = \begin{pmatrix} x(t) \\ \dot{x}(t) \\ \alpha(t) \\ \dot{\alpha}(t) \end{pmatrix} \quad (2.44)$$

Avec :

$$u(t) = V_m(t) \quad (2.45)$$

La sortie de notre système correspond donc à la position de la bille :

$$y(t) = x(t) = z_1(t) \quad (2.46)$$

Ensuite, la dynamique du système est donnée par le vecteur :

$$\dot{z}(t) = \begin{pmatrix} \dot{z}_1(t) \\ \dot{z}_2(t) \\ \dot{z}_3(t) \\ \dot{z}_4(t) \end{pmatrix} = \begin{pmatrix} \dot{x}(t) \\ \ddot{x}(t) \\ \dot{\alpha}(t) \\ \ddot{\alpha}(t) \end{pmatrix} \quad (2.47)$$

À partir de des équations (2.14) et (2.17), nous trouvons :

$$\ddot{x}(t) = K_{bb} \sin(\theta(t)) = K_{bb} \frac{L}{r} \sin(\alpha(t)) = K_{bb} \frac{L}{r} \sin(z_3(t)) \quad (2.48)$$

Par ailleurs, à partir de (2.21) et (2.40) nous trouvons :

$$\frac{\theta(p)}{V_m(p)} = \frac{\frac{L}{r}\alpha(p)}{V_m(p)} = \frac{K_m}{p(1 + \tau p)} \quad (2.49)$$

Ainsi :

$$\ddot{\alpha}(t) = \frac{-1}{\tau} \dot{\alpha}(t) + \frac{K_m r}{\tau L} V_m(t) = \frac{-1}{\tau} z_4(t) + \frac{K_m r}{\tau L} u(t) \quad (2.50)$$

Pour des raisons de simplification, nous allons poser :

$$K_1 = K_{bb} \frac{L}{r}, \quad K_2 = \frac{K_m r}{L}$$

Finalement, nous obtenons une représentation d'état non linéaire du système :

$$\begin{cases} \dot{z}_1(t) = \dot{x}(t) = z_2(t) \\ \dot{z}_2(t) = \ddot{x}(t) = K_1 \sin(z_3(t)) \\ \dot{z}_3(t) = \dot{\alpha}(t) = z_4(t) \\ \dot{z}_4(t) = \ddot{\alpha}(t) = \frac{-1}{\tau} z_4(t) + K_2 u(t) \end{cases} \quad (2.51)$$

## 2.7 Point d'équilibre

À partir de notre modèle d'état, nous identifions les points d'équilibre de notre système non linéaire  $x = x_{eq}$  pour une entrée nulle  $u(t) = u_{eq} = 0$ .

Nous avons le système d'équations différentielles suivant et nous cherchons les points pour lesquels la dynamique du système est nulle  $\dot{z} = 0$  :

$$\begin{cases} \dot{z}_1(t) = z_2(t) = 0 \\ \dot{z}_2(t) = K_1 \sin(z_3(t)) = 0 \\ \dot{z}_3(t) = z_4(t) = 0 \\ \dot{z}_4(t) = \frac{-1}{\tau} z_4(t) + K_2 u(t) = \frac{-1}{\tau} z_4(t) + K_2 u_{eq} = 0 \end{cases} \iff \begin{cases} z_1 \in \mathbb{R} \\ z_2 = 0 \\ z_3 = 0 \\ z_4 = 0 \end{cases}$$

Donc, le vecteur d'état d'équilibre est donné par :

$$z_{eq} = \begin{pmatrix} z_1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad (2.52)$$

Physiquement, cela signifie que la bille peut être à n'importe quelle position sur le rail, mais que sa vitesse est nulle et que le rail est en position horizontale, avec une vitesse angulaire nulle (le servomoteur ne tourne pas). Par conséquent, elle est en repos (sa dynamique est nulle) et sa position ne change pas, se trouvant ainsi en un point d'équilibre.

## 2.8 Linéarisation

Nous allons linéariser notre modèle non linéaire en calculant les dérivées partielles (jacobienne) pour évaluer la stabilité autour du point d'équilibre choisi.

Nous choisissons le point d'équilibre  $z_{eq} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$  vu qu'il correspond à l'état où la bille se retrouve en repos au milieu du rail. Avec  $V_m = u_{eq} = 0$ , nous avons le système suivant :

$$\begin{cases} \dot{\tilde{z}}(t) = A\tilde{z}(t) + B\tilde{u}(t) \\ \tilde{y}(t) = C\tilde{z}(t) \end{cases} \quad (2.53)$$

Où

$$A = \frac{\partial f(z, u)}{\partial z} \Big|_{z=z_{eq}} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & K_1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & \frac{-1}{\tau} \end{pmatrix} \quad (2.54)$$

représente la matrice jacobienne de  $f(z, u)$  par rapport à  $z$  évaluée à  $z_{eq}$ ,

$$B = \frac{\partial f(z, u)}{\partial u} \Big|_{z=z_{eq}} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ \frac{K_2}{\tau} \end{pmatrix} \quad (2.55)$$

représente la matrice jacobienne de  $f(z, u)$  par rapport à  $u$  évaluée à  $z_{eq}$ , et

$$C = \frac{\partial g(z, u)}{\partial z} = (1 \ 0 \ 0 \ 0) \quad (2.56)$$

représente la matrice jacobienne de  $g(z, u)$  par rapport à  $z$  évaluée à  $z_{eq}$ .

Il est à précisé que :

$$\begin{cases} \dot{z}(t) = f(z, u) \\ y(t) = g(z, u) \end{cases} \quad (2.57)$$

Donc, le modèle linéarisé autour d'un point d'équilibre de notre système est le suivant :

$$\begin{cases} \dot{\tilde{z}}(t) = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & K_1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & \frac{-1}{\tau} \end{pmatrix} \tilde{z}(t) + \begin{pmatrix} 0 \\ 0 \\ 0 \\ \frac{K_2}{\tau} \end{pmatrix} \tilde{u}(t) \\ \tilde{y}(t) = (1 \ 0 \ 0 \ 0) \tilde{z}(t) \end{cases} \quad (2.58)$$

Ainsi, ce modèle obtenu sera utilisé pour la suite, afin d'analyser et d'asservir le système. Notre loi de commande sera donc établie en se basant sur ce modèle linéaire. Toutefois, pour mettre en place une loi de commande par retour d'état et observer les états du système, il est essentiel que ce dernier soit commandable et observable. Par conséquent, la prochaine étape consiste à étudier ces deux propriétés.

### 2.8.1 Commandabilité

La matrice de commandabilité  $Co$  est définie par  $Co = [B \ AB \ A^2B \ A^3B]$ , où :

$$Co = \begin{pmatrix} 0 & 0 & 0 & \frac{K_1 \cdot K_2}{\tau} \\ 0 & 0 & \frac{K_1 \cdot K_2}{\tau} & -\frac{K_1 \cdot K_2}{\tau^2} \\ 0 & \frac{K_2}{\tau} & -\frac{\tilde{K}_2}{\tau^2} & \frac{\tilde{K}_2}{\tau^3} \\ \frac{K_2}{\tau} & -\frac{\tilde{K}_2}{\tau^2} & \frac{\tilde{K}_2}{\tau^3} & -\frac{K_2}{\tau^4} \end{pmatrix} \quad (2.59)$$

La matrice  $Co$  est de rang 4, ce qui indique que toutes les paires d'états sont contrôlables. Ainsi, le système est commandable.

### 2.8.2 Observabilité

L'observabilité d'un système est évaluée à l'aide de la matrice d'observabilité  $O$ , définie par :

$$O = \begin{bmatrix} C \\ CA \\ CA^2 \\ CA^3 \end{bmatrix} \quad (2.60)$$

où  $O$  est la matrice d'observabilité. Pour ce système,  $O$  est donnée par :

$$O = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & K_1 & 0 \\ 0 & 0 & 0 & K_1 \end{pmatrix} \quad (2.61)$$

Avec un rang de 4, la matrice  $O$  est de rang plein, ce qui signifie que toutes les paires d'états sont observables. Donc le système est observable.

## 2.9 Conclusion

avons commencé par établir nos hypothèses de travail, puis dérivé les équations physiques nécessaires. Ensuite, nous avons effectué la modélisation par la fonction de transfert et élaboré un modèle non linéaire. Par la suite, nous avons linéarisé ce modèle autour du point d'équilibre afin d'obtenir un modèle linéaire. Dans le prochain chapitre, nous aborderons l'étape de la commande, où ces modèles serviront de base pour formuler des stratégies de contrôle efficaces pour notre système.

## Chapitre 3

# Commande du système et synthèse d'un observateur

Comme mentionné précédemment, l'objectif principal est d'établir une commande pour asservir le système de la bille sur rail. Ainsi, la finalité de cette commande est de contrôler la position de la bille le long du rail. En d'autres termes, nous avons développé une commande permettant à la position de la bille de suivre une référence donnée. Cette étape intervient après la modélisation du système et l'obtention d'un modèle mathématique de la maquette. Ensuite, nous avons mis en œuvre deux types de commandes différentes : une commande fréquentielle basée sur un contrôleur fréquentiel et une commande par retour d'état.

Pour chaque asservissement, le modèle du système est analysé afin de concevoir la commande selon une démarche spécifique. Cette démarche comprend d'abord l'analyse du système, puis la mise en place de la commande, suivie de la simulation et de l'analyse des résultats. Si nécessaire, la commande est recalibrée avant de passer à l'implémentation sur la maquette. Aussi, ce chapitre comprend la mise en place d'un observateur destiné à estimer les états internes du système. La calibration de l'estimateur ainsi que le test de son fonctionnement en simulation ont été effectués.

Donc, dans les sections suivantes, l'asservissement du système par les différentes commandes est expliqué en détail.

### 3.1 Commande fréquentielle :

Dans cette section de la commande fréquentielle, l'objectif demeure inchangé : garantir que la position de la bille suive une référence prédéfinie. Cependant, le système global se présente comme un système d'ordre quatre, où le modèle du servomoteur constitue un système d'ordre deux, tout comme le procédé de la bille sur rail. De plus, comme illustré dans le schéma bloc de la Figure 2.1, la sortie du servomoteur, qui est l'angle, agit comme une entrée pour le modèle de la bille sur rail.

Ainsi, la commande du servomoteur doit être plus rapide pour garantir la convergence de l'angle du servomoteur, ce qui permettra ensuite de commander la position de la bille. Pour cela, la structure de commande choisie est divisée en deux parties : d'abord, l'asservissement de l'angle du servomoteur pour suivre une référence donnée, puis l'utilisation d'un contrôleur pour asservir le système de la bille sur rail et contrôler sa position.

Toutefois, nous avons opté pour une approche hiérarchique, en commençant par commander uniquement le servomoteur avant de passer à la commande du système complet. Ainsi, nous avons d'abord mis en place l'asservissement du servomoteur et nous nous sommes assurés de son bon fonctionnement avant de passer à l'étape de la commande du système complet. Les détails de cette mise en œuvre sont exposés dans la section suivante.

### 3.1.1 Commande du servomoteur :

Tout d'abord, nous avons étudié le modèle du servomoteur. Ainsi, la fonction de transfert du servomoteur, qui définit la relation entre la tension d'entrée et la sortie (l'angle), a été trouvée précédemment comme suit :

$$F_{sm}(p) = \frac{K_m}{p(1 + \tau p)} \quad (3.1)$$

Le système du servomoteur présente une instabilité en boucle ouverte due à la présence d'un pôle à l'origine. Cependant, il est important de souligner que le système est stable en boucle fermée. Par conséquent, la figure suivante présente la réponse indicielle du système en boucle fermée :

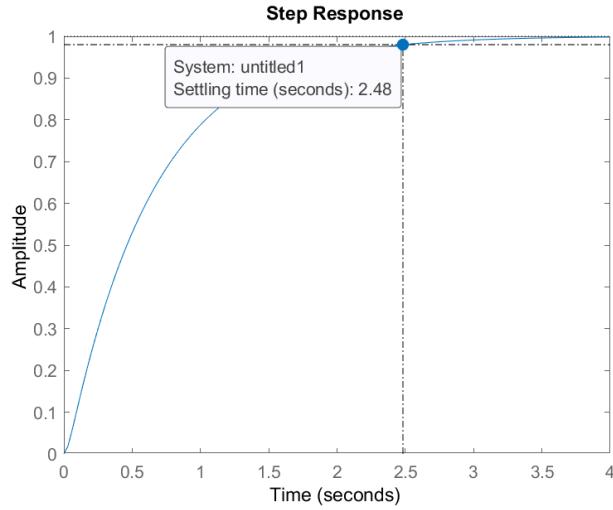


FIGURE 3.1 – Réponse indicielle du servomoteur en boucle fermée

En revanche, nous constatons que le temps de réponse du système est de 2,48 secondes, ce qui est trop lent. Par conséquent, l'objectif est de mettre en place une commande pour rendre le système plus rapide.

Étant donné que la dynamique du système est définie par ses pôles (ou par le pôle dominant), nous avons tracé le lieu des racines de la fonction de transfert du servomoteur, comme illustré sur la figure suivante :

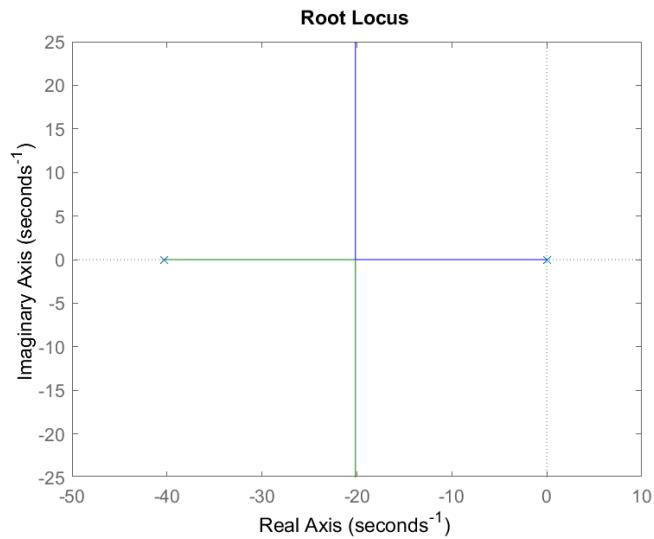


FIGURE 3.2 – Lieu des racines de  $F_{sm}$

Donc, nous pouvons constater qu'il est possible de rendre le système plus rapide avec un simple gain, où nous pouvons modifier les pôles et les éloigner de l'axe imaginaire. Cela signifie que leur dynamique est plus rapide, et par conséquent, la dynamique du système serait également plus rapide. Mais nous pouvons également observer que les pôles peuvent devenir complexes conjugués pour des valeurs de gain un peu importantes. Par conséquent, le système risque de présenter des oscillations.

En revanche, le système comporte deux pôles et aucun zéro. L'ajout d'un gain pourrait entraîner une diminution de la marge de phase, actuellement de 87,8 degrés. Il est donc nécessaire d'injecter de la phase dans notre système pour garantir une marge de phase adéquate. Cette action est cruciale car ce système représente une boucle interne au système complet, et il est impératif de maximiser sa stabilité.

Par conséquent, nous avons opté pour un correcteur de type proportionnel-dérivé afin de rendre le système plus rapide grâce à l'action proportionnelle. De plus, l'action dérivée va ajouter de la phase au système, améliorant ainsi sa stabilité et amortissant les oscillations.

#### *Remarque :*

Il est à noter que la précision du système est déjà parfaite, grâce à l'intégrateur de la chaîne directe qui garantit une erreur de précision nulle. Ainsi, pour une entrée échelon, la précision du système est assurée.

#### Synthèse du correcteur :

Nous avons donc choisi d'utiliser un correcteur proportionnel-dérivé dont la structure est la suivante :  $C_{sm}(p) = K_{c1}(p + z_1)$ . Ainsi, la loi de commande de notre système, représentant la tension d'entrée du servomoteur, prend la forme suivante :  $U(p) = V_m(p) = K_{c1}(p + z_1)[\theta_i(p) - \theta_o(p)]$ .

Avec :

- $K_{c1}$  : gain proportionnel du correcteur
- $z_1$  : zéro du correcteur
- $\theta_i$  : angle désirée du servomoteur (référence)
- $\theta_o$  : angle de sortie du servomoteur

À partir de cela, nous allons donc utiliser et implémenter la commande à partir du schéma d'asservissement suivant :

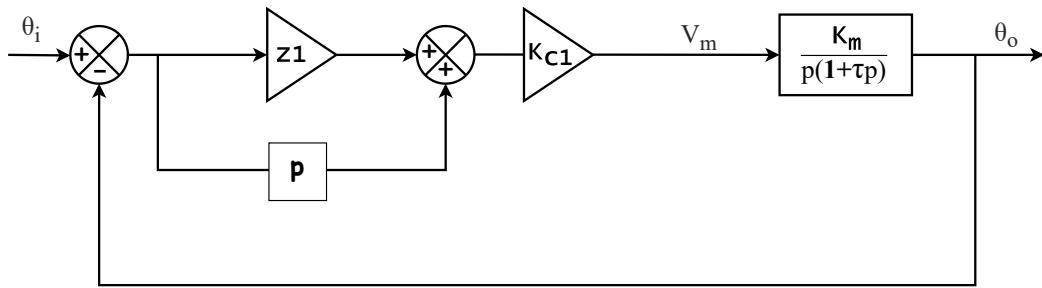


FIGURE 3.3 – Schéma block de l'asservissement du servomoteur

À partir du schéma, la fonction de transfert entre l'entrée et la sortie est déterminée comme suit :

$$\frac{\theta_o(p)}{\theta_i(p)} = \frac{K_m K_{c1}(p + z_1)}{\tau p^2 + (1 + K_m K_{c1})p + K_m K_{c1} z_1} = \frac{\frac{K_m K_{c1}}{\tau}(p + z_1)}{p^2 + \frac{(1 + K_m K_{c1})}{\tau}p + \frac{K_m K_{c1} z_1}{\tau}} \quad (3.2)$$

Le correcteur proportionnel-dérivé conserve ainsi le degré du système, et il est à préciser qu'un système du deuxième ordre s'écrit sous la forme :

$$\frac{\theta_o(p)}{\theta_i(p)} = \frac{\omega_n^2}{p^2 + 2\zeta\omega_np + \omega_n^2} \quad (3.3)$$

Avec :

- $\zeta$  : facteur d'amortissement
- $\omega_n$  : fréquence naturelle du système

Par la suite, en identifiant les dénominateurs des fonctions de transfert des équations précédentes (3.2) et (3.3), nous déterminons les paramètres du correcteur en fonction des paramètres désirés de notre système, à savoir le facteur d'amortissement, la fréquence naturelle et les paramètres du servomoteur :

$$K_{c_1} = \frac{2\zeta\omega_n\tau - 1}{K_m} \quad (3.4)$$

Et le zéro du correcteur est donc :

$$z_1 = \frac{\omega_n^2\tau}{K_m K_{c_1}} \quad (3.5)$$

Il est à préciser que le temps de réponse d'un système du deuxième ordre est donné par la formule suivante :

$$t_r = -\frac{\ln(0.05\sqrt{1-\zeta^2})}{\zeta\omega_n} \quad (3.6)$$

Ainsi, la valeur du dépassement est fournie par la formule suivante :

$$\%d = 100e^{\frac{-\pi\zeta}{\sqrt{1-\zeta^2}}} \quad (3.7)$$

Donc, il reste à choisir les paramètres du système afin de satisfaire les performances désirées. Ensuite, nous déterminerons les valeurs des gains qui vont répondre à ces exigences, tout en veillant à ne pas dépasser une certaine valeur de saturation de la tension d'entrée.

#### **Remarque :**

Il est également important de noter que la fonction de transfert du système asservi, donnée à l'équation (3.2), ne correspond pas strictement au système du deuxième ordre prototype présenté par l'équation (3.3), en raison de la présence d'un zéro au numérateur. Cela entraînera une légère différence dans le comportement attendu. C'est pourquoi il est crucial de simuler le comportement du système avant d'implémenter la commande sur le dispositif réel, afin de s'assurer que les spécifications sont respectées et d'anticiper les écarts potentiels entre le modèle théorique et les performances du système réel.

#### **Choix des paramètres et simulation :**

Ainsi, nous avons analysé la réponse indicielle de notre système asservi avec différentes valeurs des paramètres du correcteur pour observer le comportement du système commandé. Par conséquent, la figure suivante présente la réponse indicielle du système avec la correction où nous avons fixé la valeur de  $z_1 = 5$  et avons fait varier la valeur de  $K_{c_1}$  :

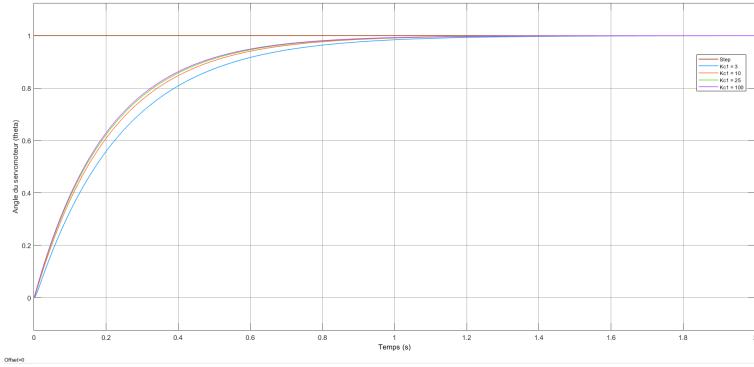


FIGURE 3.4 – Réponse indicielle du servomotor commandé pour différentes valeurs de  $K_{c_1}$

Ensuite, nous avons fixé la valeur du gain proportionnel à 10 et nous avons varié la valeur de  $z_1$  pour obtenir les réponses indicielles correspondantes, illustrées sur la figure suivante :

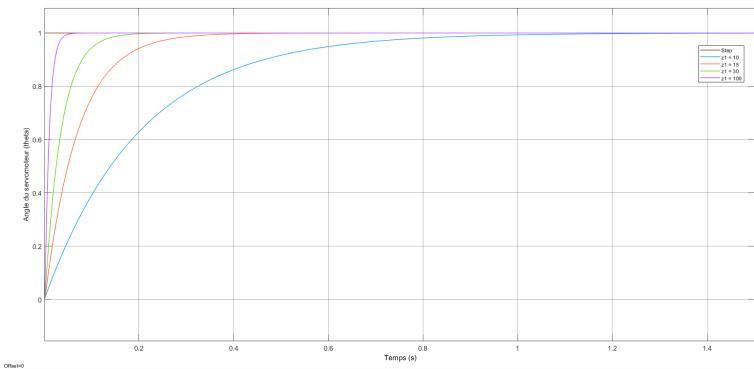


FIGURE 3.5 – Réponse indicielle du servomotor commandé pour différentes valeurs de  $z_1$

Donc, nous pouvons constater qu'il est possible de commander le servomoteur et de réduire le temps de réponse du système. De plus, l'effet de la variation des valeurs des paramètres du correcteur sur la réponse du système commandé nous permet de visualiser leurs limites afin de ne pas saturer la commande (la tension d'entrée). Cela nous guide dans le choix des caractéristiques de notre système asservi. Par ailleurs, la prochaine étape consiste à tester cette commande sur le dispositif réel afin de valider la solution et d'ajuster les paramètres pour sélectionner la solution la plus adéquate.

### Implémentation du correcteur sur la maquette :

La commande du servomoteur et du système en général se fait donc à travers Simulink. Pour cela, nous nous appuyons sur le modèle "SRV02 QUARC Integration" fourni par Quanser, qui est implanté pour lire la position de la bille ainsi que l'angle du servomoteur. Ce modèle permet également de générer la tension  $V_m$  pour commander la maquette, le tout à travers la carte d'E/S. Ainsi, nous avons mis en œuvre la loi de commande illustrée dans la Figure 3.3 en suivant le schéma d'implémentation présenté dans la Figure (3.17) de l'annexe.

Étant donné que la vitesse angulaire est nécessaire pour la commande, cette dernière n'est évidemment pas accessible via la mesure directe, et il n'est pas possible de l'obtenir en utilisant un bloc déivateur, car l'action derivée n'est pas causal. Par conséquent, nous avons eu recours à l'utilisation d'un pseudo-déivateur, qui lui est causal, afin d'obtenir la vitesse angulaire et de l'utiliser par la suite dans la loi de commande.

## Tests et résultats :

Vu que nous voulons que ce sous-système soit une boucle interne au système global, sa réponse doit donc converger rapidement vers la consigne. C'est pourquoi nous avons décidé que la réponse désirée ait un temps de réponse de  $t_s = 0.1s$ , et par la suite, nous autorisons un dépassement de 7%. Il est possible de choisir une autre limite pour le dépassement. Cependant, si nous décidons de réduire les dépassements, nous aurons besoin de gains de correcteur plus élevés. Vu que notre modèle est basé sur une approximation et que la dérivée est obtenue avec un pseudo-déivateur, toute petite erreur de calcul dans le procédé réel sera ensuite amplifiée et injectée dans la commande. Ainsi, la commande risque de ne pas fonctionner correctement, ce qui peut entraîner l'instabilité du système par la suite.

Ainsi, nous utilisons les équations (3.4), (3.5), (3.6) et (3.7) pour trouver les valeurs des paramètres  $K_{cl} = 0.41$  et  $z_1 = 100.92$ .

Nous avons donc calibré les valeurs des gains et testé la commande sur le servomoteur, et nous avons obtenu la réponse suivante :

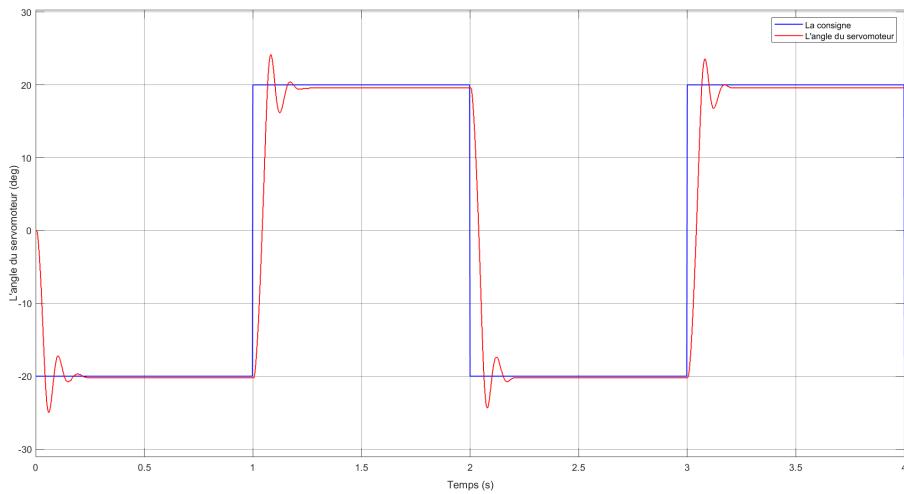


FIGURE 3.6 – Réponse de l'angle du servomoteur

Nous constatons ainsi que l'angle du servomoteur converge vers la référence de 20 degrés en moins de 0.3 secondes, cependant, il présente également des oscillations ainsi qu'un dépassement de 15% (une réponse maximale de la réponse de  $\pm 23$  degrés). Ces résultats diffèrent quelque peu des prévisions théoriques, principalement en raison des approximations faites et de la présence du zéro stable dans la fonction de transfert (3.2). Un zéro stable tend à ralentir la réponse du système et à induire des oscillations, ce qui explique les écarts observés par rapport aux attentes théoriques. De plus, les valeurs des gains du correcteur sont calculées sur la base des données fournies par la documentation, mais dans notre modèle, ces valeurs peuvent varier légèrement. Cependant, la Figure 3.6 révèle également une différence entre la valeur finale de l'angle et la référence, indiquant une erreur statique mesurée à 0,5 degré. Bien que présente, cette erreur n'est pas significativement importante. Elle peut être due à l'effet du poids du rail sur le servomoteur, qui pourrait légèrement perturber l'angle de ce dernier.

Par conséquent, nous pouvons nous appuyer sur ce modèle pour nos travaux ultérieurs, car notre objectif initial est de contrôler la position de la bille le long du rail.

### 3.1.2 Commande du système complet :

L'objectif de cette partie est donc de commander le système complet de la bille sur rail, où nous cherchons à contrôler la position de la bille à travers la mise en place d'un correcteur fréquentiel. Ainsi, comme expliqué précédemment, l'asservissement du système complet sera réalisé

en se basant sur une architecture de commande répartie en une boucle interne qui se chargera du contrôle de l'angle du servomoteur, établi lors de la section précédente, et une boucle externe qui contrôlera la position de la bille le long du rail.

D'abord, nous devons étudier la fonction de transfert du sous-système de la bille sur rail, qui est donnée comme suit :

$$G_{br}(p) = \frac{K_{br}}{p^2} \quad (3.8)$$

Ce système est donc instable du fait qu'il comporte deux pôles à l'origine. Ensuite, nous traçons le lieu des racines, comme illustré dans la figure suivante :

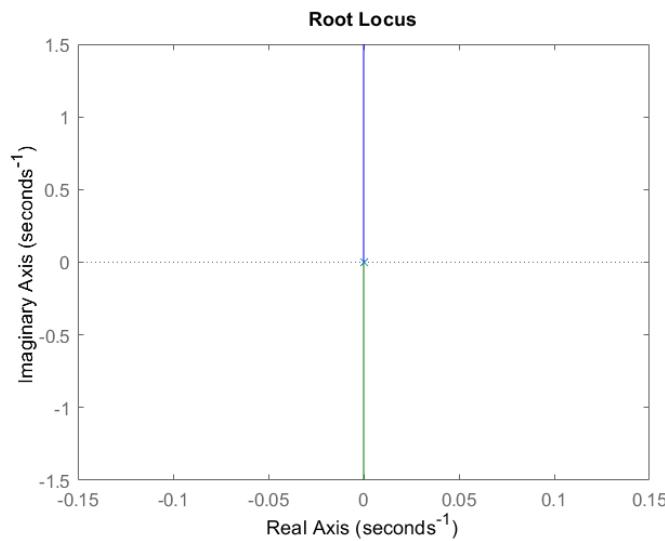


FIGURE 3.7 – Lieu des racines de  $G_{br}$

Ainsi, nous pouvons constater qu'il est impossible de ramener les pôles du système dans le demi-plan gauche avec un correcteur proportionnel, c'est-à-dire en ajoutant simplement un gain.

Par la suite, le système est caractérisé par une marge de phase nulle. Nous cherchons donc à ajouter de la phase au système. Étant donné que la chaîne directe comporte deux intégrateurs, une erreur statique nulle est déjà garantie, ce qui signifie que notre système est précis et qu'il n'est donc pas nécessaire d'ajouter un effet intégral. Par conséquent, nous avons choisi d'utiliser un correcteur de type proportionnel-dérivé afin d'asservir le système et de le rendre stable.

### Synthèse du correcteur :

La structure du correcteur est donc :  $C_{br}(p) = K_{c2}(p + z_2)$ . Par la suite, la loi de commande représentant la position désirée du servomoteur est la suivante :  $U(p) = \theta_i(p) = K_{c2}(p + z_2)(X_i(p) - X_o(p))$ . Par ailleurs, le schéma bloc de l'asservissement est représenté dans la figure suivante :

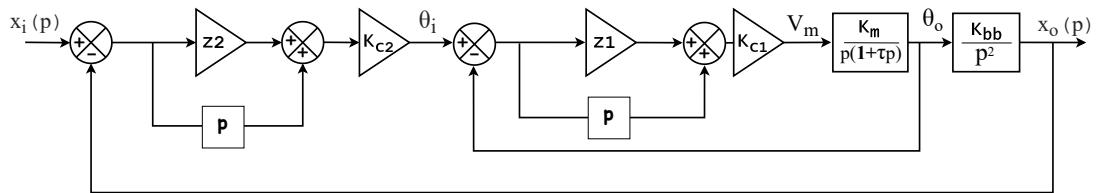


FIGURE 3.8 – Schéma bloc de l'asservissement du système complet

Avec :

- $K_{c2}$  : gain du correcteur
- $z_2$  : zéro du correcteur

C'est là que réside l'intérêt d'avoir la boucle interne plus rapide, où nous pouvons supposer que l'angle du servomoteur converge et donc  $\theta_i(t) = \theta_o(t)$ . Par conséquent, nous obtenons le schéma suivant :

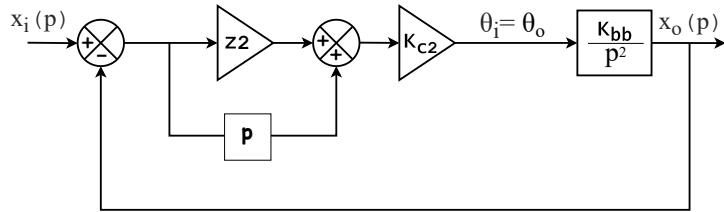


FIGURE 3.9 – Schéma bloc de l'asservissement du système complet avec  $\theta_i(t) = \theta_o(t)$

À partir du schéma bloc de la Figure 3.9, nous établirons la fonction de transfert entre l'entrée et la sortie, qui est obtenue comme suit :

$$\frac{X_o(p)}{X_i(p)} = \frac{K_{br}K_{c2}(p + z_2)}{p^2 + K_{br}K_{c2}p + K_{bb}K_{c2}z_2} \quad (3.9)$$

Par identification avec l'équation 3.3, nous trouvons l'expression du gain :

$$K_{c2} = \frac{2\zeta\omega_n}{K_{br}} \quad (3.10)$$

Ainsi que l'expression du zéro :

$$z_2 = \frac{\omega_n^2}{K_{br}K_{c2}} \quad (3.11)$$

Il reste donc à établir les caractéristiques désirées du système asservi afin de trouver la valeur du gain et du zéro qui vont réaliser ces caractéristiques.

**Remarque :**

Il est également important de noter que la fonction de transfert contient un zéro, ce qui ne correspond pas exactement à un système prototype de deuxième ordre donné par la fonction de transfert de l'équation (3.3), comme c'est le cas pour l'asservissement du servomoteur. Par conséquent, nous allons d'abord effectuer des simulations avant de passer à l'implémentation. De plus, il est évident que des différences apparaîtront entre le modèle théorique et le modèle réel.

**Choix des paramètres et simulation :**

L'objectif de cette partie consiste à établir un modèle Simulink afin de tester la commande, dans le but d'avoir un aperçu de la réponse et de s'assurer que la commande satisfait les exigences avant de l'implémenter. Il est aussi nécessaire de veiller à ce que la commande ne dépasse pas une saturation, où l'angle du servomoteur ne doit pas dépasser  $\pm 45$  degrés.

Tout d'abord, nous fixons la valeur de  $z_2$  à 1,5 et nous faisons varier la valeur du gain pour observer l'effet de cette variation. Cela nous permet de vérifier que la commande ne se sature pas pour ces valeurs. Ainsi, nous obtenons les graphes illustrés dans la Figure 3.10 :

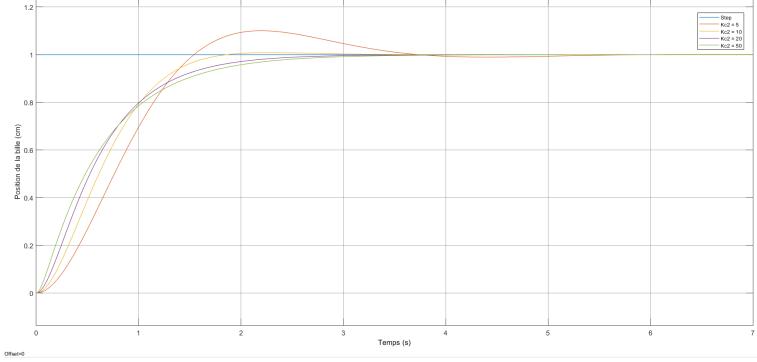


FIGURE 3.10 – Réponse indicielle du système commandé pour différentes valeurs de  $K_c$

Par la suite, nous avons fixé la valeur du gain  $K_{c_2}$  à 10 et avons testé plusieurs valeurs de  $z_2$  afin de constater l'effet du zéro du correcteur et de vérifier que la valeur de l'angle  $\theta$  ne dépasse pas la saturation exigée. Voici les résultats obtenus :

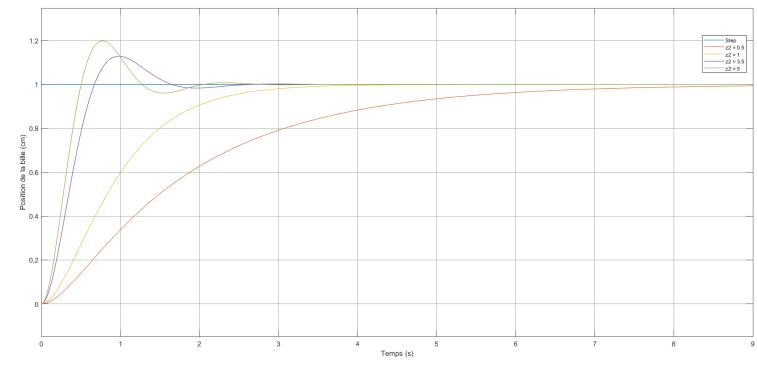


FIGURE 3.11 – Réponse indicielle du système commandé pour différentes valeurs de  $z$

Donc, on peut constater qu'il est possible de rendre le système plus rapide et d'avoir une réponse amortie ou bien contenante des oscillations. Comme nous pouvons le constater, l'effet d'augmentation du gain ainsi que l'effet de changement de la position du zéro du correcteur sont évidents.

Ainsi, en augmentant la valeur du gain, nous constatons une amélioration du temps de réponse du système. Cependant, la valeur du zéro affecte le régime transitoire. Nous observons que des oscillations apparaissent lorsque la valeur de  $z_2$  augmente. Ce zéro contribue à amortir les oscillations, mais cela rend également le système plus lent.

Dans les sections suivantes, l'implémentation de la commande sera réalisée en choisissant soigneusement les caractéristiques pour qu'elles soient adéquates avec le modèle réel.

### Implémentation du correcteur sur la maquette :

La commande du système complet est réalisée en suivant la même méthodologie, en utilisant un modèle Simulink. Ainsi, la commande est basée sur le schéma illustré dans la Figure 3.18 de l'annexe.

Aussi nous avons également utilisé un filtre (pseudo-déivateur) afin d'obtenir la dérivée, et c'est pour la même raison (la non-causalité du déivateur).

Comme nous ajoutons également des blocs de saturation pour des raisons de sécurité, dans le cas où une erreur de calcul se produirait, afin d'éviter que la tension injectée en entrée soit trop grande, ce qui pourrait causer des comportements indésirables.

## Tests et résultats :

Pour l'implémentation sur la maquette, nous avons choisi que les performances désirées soient un temps de réponse de 3 secondes et un dépassement de 10 %. Il est à préciser que ces valeurs ont été choisies car elles ne font pas saturer la commande en simulation et qu'elles nous permettent de faire fonctionner la commande sur la maquette. Si, par exemple, nous choisissons que le système soit plus rapide et avec moins d'oscillation, cela ne peut pas être fonctionnel sur le dispositif réel en raison des approximations ou de la saturation.

La prochaine étape consiste donc à tester la commande avec les valeurs du gain et du zéro trouvées, qui sont  $K_{c2} = 5.14$  et  $z_2 = 1.49$ . Nous avons donc obtenu la réponse montrée sur la figure ci-dessous :

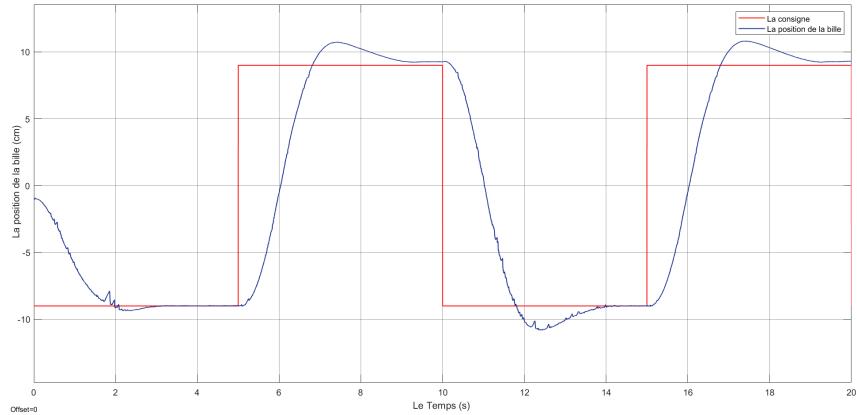


FIGURE 3.12 – Réponse de la position de la bille

Les résultats de la figure montrent donc que la bille atteint une position stable de  $\pm 9$  cm après environ 3,5 secondes, avec une erreur de position d'environ 0,3 cm et un dépassement de 19 %. Ainsi, la figure suivante montre un agrandissement de la réponse maximale ainsi que du régime permanent, permettant de comparer la réponse avec la référence et de vérifier les résultats obtenus :

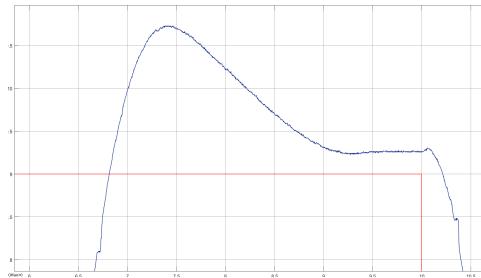


FIGURE 3.13 – Agrandissement sur la réponse de la position de la bille

Ce décalage entre la simulation et la partie expérimentale est dû aux approximations du modèle de la maquette. La synthèse du correcteur est basée sur un modèle idéal du servomoteur, où les deux pôles électriques du moteur ont été négligés (car ils sont trop rapides et n'imposent pas leur dynamique), seuls les deux pôles mécaniques sont pris en compte, ce qui peut causer un décalage. De plus, les différentes valeurs des gains ont été trouvées en se basant sur un modèle linéaire alors que le modèle réel comporte des non-linéarités. En outre, pour obtenir un modèle mathématique de la maquette, nous avons utilisé directement les valeurs des paramètres fournies par la documentation de Quanser, sans procéder à une étape d'identification des valeurs exactes de notre maquette.

Il est important de prendre en compte d'autres facteurs qui peuvent perturber le système de commande. Par exemple, la maquette est placée sur une table, ce qui peut entraîner une légère inclinaison de la maquette par rapport à une position horizontale parfaite. De plus, d'autres perturbations extérieures peuvent également influencer la commande, entraînant ainsi des incertitudes sur notre modèle.

Par conséquent, il est impératif d'améliorer notre système et de développer une modélisation plus précise de la maquette afin de pouvoir mettre en place des commandes plus précises et garantir de meilleures performances. Pour ce faire, il est nécessaire de vérifier les valeurs réelles des paramètres de notre maquette. Ensuite, afin de réduire les oscillations, il est envisageable de modifier la valeur du zéro de l'action dérivée et d'augmenter le gain pour améliorer la précision. De plus, l'ajout d'un terme intégral et l'utilisation d'un correcteur PID pour contrôler la position de la bille et améliorer le rejet de perturbation, bien que cela puisse complexifier le système.

## 3.2 Commande par retour d'état :

Dans cette section, nous allons synthétiser une loi de commande en utilisant le retour d'état, en nous appuyant sur la représentation d'état linéaire établie dans le chapitre 2.

### 3.2.1 Calcul de la loi de commande

Étant donné que nous avons déjà vérifié la commandabilité de notre modèle, il nous reste à calculer le gain du retour d'état. C'est pourquoi nous avons défini le cahier des charges suivant : " Nous désirons obtenir une réponse sans oscillation ni dépassement, avec un temps de réponse inférieur à 5 secondes."

Pour répondre au cahier des charges précédent, nous avons décidé de choisir des valeurs propres désirées avec une partie réelle négative afin d'éviter toute oscillation indésirable. Étant donné que nous souhaitons que les pôles du servomoteur convergent avant ceux du rail, nous avons choisi le vecteur de valeurs propres désirées  $P_{des} = [-2 \ -3 \ -20 \ -21]$ . Ainsi, les deux derniers pôles convergent rapidement et avant les deux premiers.

En utilisant Matlab "le code correspondant se trouve en annexe du rapport" Nous avons calculé le gain de retour d'état et obtenu  $K = [97.7527 \ 91.0031 \ 171.3387 \ 1.5416]$ .

Les pôles du système déterminent la dynamique du système, c'est-à-dire son comportement transitoire. En revanche, le gain K seul ne permet pas de régler le problème de la précision ni de choisir la valeur du régime permanent. Par conséquent, nous allons calculer un gain précompensateur N qui nous permettra de régler le gain statique du système en boucle fermée.

Nous avons  $G(p) = C(pI - A)^{-1}B$ , la fonction de transfert associée à notre modèle d'état. La fonction de transfert du système en boucle fermée s'écrit  $G_{bf}(p) = C(pI - A + BK)^{-1}BN$ . Supposons que  $y_{ref}(t) = y_{ref}$  est une constante. Pour que  $\lim_{t \rightarrow \infty} y(t) = y_{ref}$ , il est donc nécessaire que  $G_{bf}(0) = 1$ , c'est-à-dire  $N = \frac{1}{C(-A+BK)^{-1}B}$ .  
AN :  $N = 97.7527$

### 3.2.2 Simulation de la commande par retour d'état

Après la synthèse de la loi de commande par retour d'état et avant les tests sur la maquette, nous avons testé notre commande sur une simulation. Pour réduire les approximations, nous avons également testé cette commande sur le modèle non linéaire en utilisant un pseudo-déivateur (un filtre) que nous avons employé pour l'implémentation. La Figure 3.19 de l'annexe représente le schéma de simulation.

Nous avons testé notre système avec une consigne de signal carré de fréquence  $f = 0.075$  Hz et d'amplitude  $A = 9$  cm :

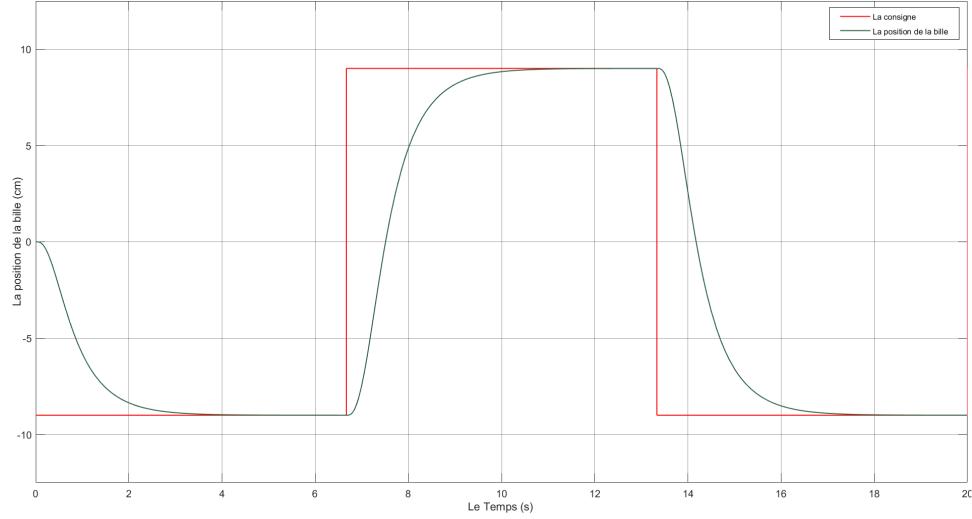


FIGURE 3.14 – Réponse de la position de la bille

Dans la figure précédente, il est évident que nous avons atteint les exigences spécifiées dans le cahier des charges, avec une réponse sans oscillation ni dépassement, un temps de réponse inférieur à 5 secondes, et une erreur statique nulle.

### 3.2.3 Résultats de l'implémentation sur la maquette

Dans cette étape, nous avons testé notre loi de commande par retour d'état sur la maquette en utilisant le même schéma que celui utilisé pour la simulation, en remplaçant simplement le bloc bille sur rail. Nous avons conservé les mêmes valeurs de  $K$  et  $N$ . Malheureusement, nous avons observé une erreur statique notable. Pour améliorer cette erreur, nous avons utilisé une méthode expérimentale pour calculer le gain statique de notre système, ce qui nous a donné une nouvelle valeur du gain précompensateur  $N_{\text{exp}} = 105.1$ .

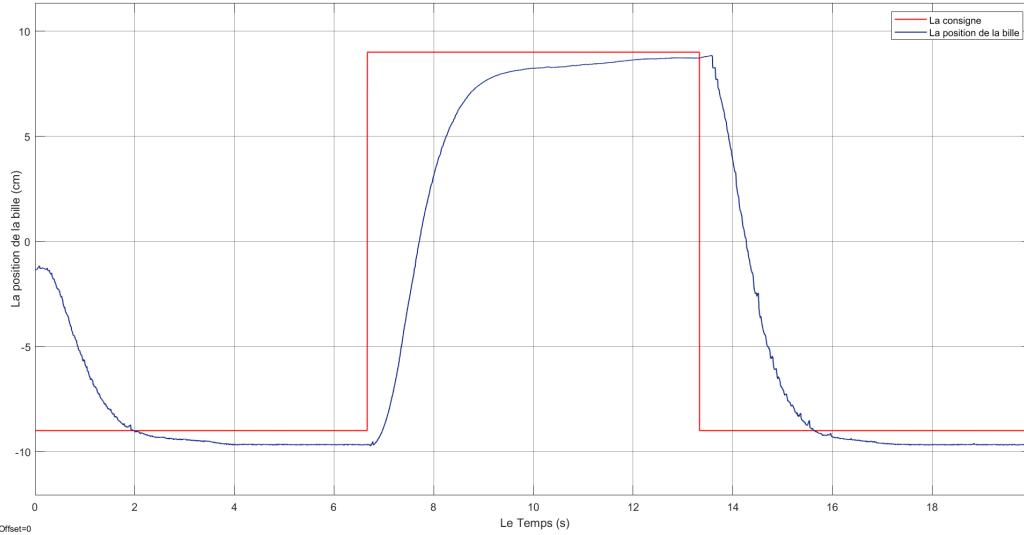


FIGURE 3.15 – Réponse de la position de la bille

Nous observons une réponse relativement acceptable sans oscillation ni dépassement, avec un temps de réponse inférieur à 5 secondes, mais une erreur statique non nulle. Cette erreur est due au recalculation du gain précompensateur, basé sur une valeur expérimentale du gain statique mesurée. En raison des approximations dans notre modèle, qui n'est pas entièrement fiable, nous constatons une différence entre les valeurs théorique et expérimentale du gain statique.

### 3.3 Synthèse d'un Observateur

Dans les chapitres précédents, nous avons constaté qu'il est nécessaire d'obtenir les états du système afin d'effectuer la commande. Cependant, seuls deux de ces états sont mesurables directement via la sortie. Pour estimer les deux états manquants, nous avons utilisé un pseudo-déivateur. Il est crucial de choisir correctement la fréquence de coupure de ce pseudo-déivateur, en tenant compte de la rapidité des états, afin de minimiser les erreurs de calcul.

Ainsi, l'objectif de cette partie est de mettre en place un observateur afin d'estimer les états internes du système. Cet observateur sera ensuite utilisé dans les lois de commande pour améliorer la performance globale du système.

#### 3.3.1 Modélisation de l'observateur

Un observateur est un système dynamique qui possède deux entrées : l'entrée du système à observer et la sortie de ce même système. La sortie de l'observateur est une estimation des états internes du système. Le modèle d'état du système peut être représenté comme suit :

$$\begin{cases} \dot{w}(t) = Fw(t) + Gy(t) + Hu(t) \\ \tilde{z}(t) = Mw(t) + Ny(t) \end{cases} \quad (3.12)$$

Avec :

$$w(t) = Tz(t) + \epsilon(t) \quad (3.13)$$

Et :

$$\epsilon(t) = z(t) - \tilde{z}(t) \quad (3.14)$$

Où :

- $w$  : le vecteur d'état de l'observateur
- $\tilde{z}$  : le vecteur d'état estimé
- $z$  : le vecteur d'état
- $y$  : le vecteur de sortie
- $u$  : le vecteur d'entrée
- $\epsilon$  : l'erreur entre  $z$  et  $\tilde{z}$

L'objectif est de faire en sorte que les états estimés convergent vers les états réels. Pour atteindre cet objectif, les équations suivantes doivent être satisfaites :

$$\begin{cases} MT + NC = I_n \\ FT - TA + GC = 0 \\ H = TB \\ Re(VP(F)) < 0 \end{cases} \quad (3.15)$$

Nous avons décidé de mettre en place un observateur de type identité afin de reconstituer tous les états du système. Ainsi les équations suivantes doivent être satisfaites :

$$\begin{cases} T = I_n = I_4 \\ M = I_n = I_4 \\ N = \phi_{n \times m} = \phi_{4 \times 1} \end{cases} \quad (3.16)$$

Il reste donc à choisir les valeurs propres de la matrice F de manière à ce que les états estimés convergent rapidement vers les états réels. Étant donné que la loi de commande se basera sur ces états estimés, il est primordial que l'erreur  $\epsilon(t)$  converge vers zéro plus rapidement que le mode le plus rapide de notre système asservi.

Ainsi, nous avons choisi comme valeurs propre  $VP(F) = \{-25, -26, -27, -28\}$ . Ensuite, pour déterminer la matrice de gain G, nous utilisons des techniques de placement des pôles qui consiste à résoudre le système d'équations suivantes :

$$F = A - GC \quad (3.17)$$

Donc, les valeurs des matrices de l'obsevateur sont obtenues à partie des équations (3.15) et (3.16), ce qui nous permet d'obtenir la représentation d'état suivante de l'observateur :

$$\begin{cases} \dot{w}(t) = \begin{bmatrix} -0.0617 & 0.0010 & 0 & 0 \\ -1.4120 & 0 & 0.0070 & 0 \\ -1.3232 & 0 & 0 & 0.0010 \\ -6.8176 & 0 & 0 & -0.0403 \end{bmatrix} w(t) + \begin{bmatrix} 061.7 \\ 1412 \\ 1323.2 \\ 6817.6 \end{bmatrix} y(t) + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 3.6828 \end{bmatrix} u(t) \\ \tilde{z}(t) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} w(t) \end{cases} \quad (3.18)$$

### 3.3.2 Validation par simulation

Une fois la valeur de G déterminée, il est crucial de valider l'observateur par des simulations. Cela permet de vérifier la convergence des états estimés vers les états réels du système. Les simulations doivent inclure des scénarios variés pour évaluer la robustesse de l'observateur face aux incertitudes et aux perturbations.

Ainsi, la Figure 3.16 suivante présente les résultats de la simulation. Où les graphes représentent l'écart entre chaque état réel et son état estimé correspondant. La figure montre que les états estimés convergent vers les états réels. En comparant les états réels et estimés, nous constatons que l'erreur est faible et converge rapidement vers zéro pour tous les états.

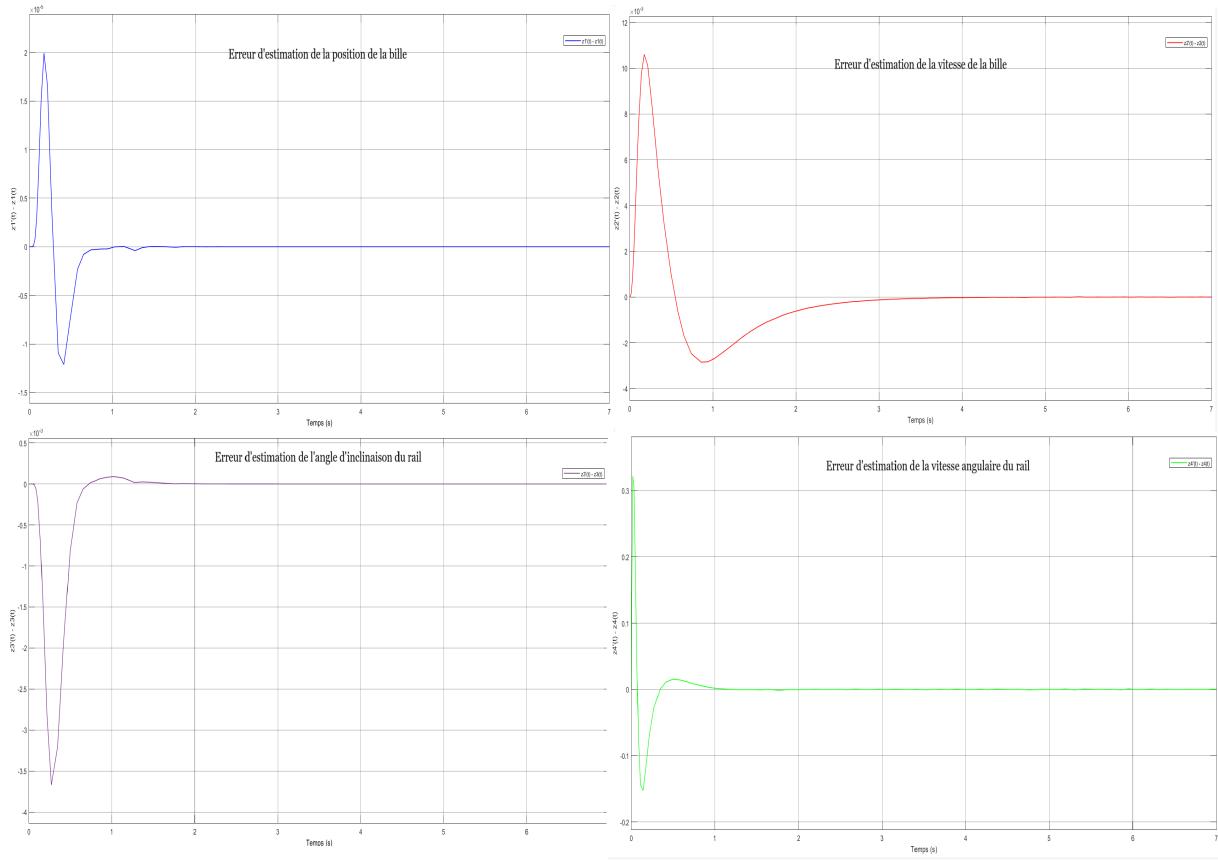


FIGURE 3.16 – Erreurs de convergence des états estimés

Les résultats obtenus précédemment restent théoriques. L'implémentation de l'observateur sur le dispositif réel n'a pas encore été réalisée pour vérifier sa validité sur la maquette. Cette étape est cruciale pour voir si nous parvenons à reconstruire les états du système. Il est également important de comparer ces résultats avec ceux obtenus en utilisant le pseudo-dérivateur et de trouver des moyens d'améliorer l'estimation des états avec l'observateur. Cela nous permettra finalement de synthétiser une loi de commande basée sur l'observateur.

### 3.4 Conclusion :

Après avoir obtenu un modèle mathématique décrivant le système, le but de cette partie était de commander le système en utilisant deux différentes méthodes d'asservissement. La première méthode consiste en une commande par retour d'état, où l'objectif était de modifier le placement des pôles du système commandable en se basant sur le modèle espace d'état, puis de tester cette commande sur la maquette pour analyser son fonctionnement. Cela inclut également l'explication des différences entre le modèle théorique en simulation et la réponse du système réel.

Ensuite, une commande fréquentielle a été implémentée sur le système, suivant une démarche structurée pour la synthèse du correcteur, afin d'asservir le système et de faire en sorte que la bille suive une référence donnée.

L'objectif principal était de valider la commande sur la maquette, analyser la réponse du système réel, constater les différences avec le modèle théorique, et trouver des explications pour ces différences afin de développer un modèle plus précis et plus performant.

Finalement, nous avons mis en place un observateur identité afin de reconstruire les états manquants et de pouvoir baser nos différentes commandes sur les états estimés par l'observateur. Cependant, nous n'avons pas pu tester la validité de l'observateur, et les résultats demeurent donc théoriques.

# Conclusion générale et perspectives

Notre étude de la maquette bille sur rail de Quanser a été menée de manière méthodique, en commençant par une analyse approfondie de la documentation disponible pour comprendre les caractéristiques et le fonctionnement de la maquette. Ensuite, nous avons passé en revue les travaux existants sur ce thème, cherchant à approfondir notre compréhension et à identifier les avancées ainsi que les lacunes dans les recherches précédentes. Puis, nous avons entamé la modélisation, une étape cruciale pour déterminer notre modèle mathématique. Sur la base de cette modélisation, nous avons exploré plusieurs approches dans la partie commande, telles que l'utilisation de la commande proportionnelle-dérivée (PD), le retour d'état et l'observateur. Ces techniques ont été essentielles pour concevoir des stratégies efficaces afin de stabiliser le système et d'atteindre les objectifs de performance fixés. Avant d'implémenter ces techniques dans notre maquette, nous avons utilisé la simulation sous Matlab pour valider et vérifier la faisabilité du modèle et de la technique de commande.

La modélisation précise de notre système s'est révélée être un défi majeur, en raison des forces et des interactions physiques complexes, incluant les frottements et les pertes d'énergie. L'implémentation sur la maquette et la détection des problèmes rencontrés n'étaient pas évidentes, mettant en avant la stabilisation de la bille et la précision comme nos principaux enjeux.

Pour les futurs travaux, nous proposerons de définir un modèle plus précis en prenant en compte l'influence du poids de la bille sur le servomoteur, le frottement sec de la bille avec le rail, et les deux points de contact de la bille avec le rail. Nous recommanderons également l'utilisation de modèles non linéaires pour capturer plus précisément les comportements réels du système. Intégrer des incertitudes, telles que des variations de masse, permettra d'évaluer la robustesse et la fiabilité du système dans des conditions imprévues. Ajouter une commande avec un effet intégrateur au retour d'état améliorera les résultats obtenus, tels que le rejet de perturbation. Ces approches amélioreront notre compréhension du système et renforceront sa capacité à fonctionner efficacement dans des environnements réels.

# Bibliographie

- [1] Awadalla Ali, Osama Taha, A Naseraldeen, and Taifour Ali. Design and implementation of ball and beam system using pid controller. 3 :1–4, 04 2017.
- [2] Santosh Anand and Rajkishore Prasad. Dynamics and control of ball and beam system. *Int. J. Recent Innov. Trends Comput. Commun.*, 5(5) :1332–1339, 2017.
- [3] Yeong-Hwa Chang, Chia-Wen Chang, Chin-Wang Tao, Hung-Wei Lin, and Jin-Shiu Taur. Fuzzy sliding-mode control for ball and beam system with fuzzy ant colony optimization. *Expert Systems with Applications*, 39(3) :3624–3633, 2012.
- [4] Bipin Krishna, Sagnik Gangopadhyay, and Jim George. Design and simulation of gain scheduling pid controller for ball and beam system. In *International Conference on Systems, Signal Processing and Electronics Engineering (ICSSEE'2012)*, volume 26, pages 199–203, 2012.
- [5] Xiaou Li and Wen Yu. Synchronization of ball and beam systems with neural compensation. *International Journal of Control, Automation and Systems*, 8 :491–496, 2010.
- [6] Sung-Kwun Oh, Han-Jong Jang, and Witold Pedrycz. The design of a fuzzy cascade controller for ball and beam system : A study in optimization with the use of parallel genetic algorithms. *Engineering Applications of Artificial Intelligence*, 22(2) :261–271, 2009.
- [7] Zhong-Hua Pang, Geng Zheng, and Chun-Xiang Luo. Augmented state estimation and lqr control for a ball and beam system. In *2011 6th IEEE Conference on Industrial Electronics and Applications*, pages 1328–1332. IEEE, 2011.
- [8] Quanser. *Quanser Ball and Beam user manual*. Quanser Inc, Markham, Ontario Canada, 2011.
- [9] Quanser. *Q2-USB Data Acquisition Board user manual*. Quanser Inc, Markham, Ontario Canada, 2012.
- [10] Mohd Fuaad Rahmat, Herman Wahid, and Norhaliza Abdul Wahab. Application of intelligent controller in a ball and beam control system. *International journal on smart sensing and intelligent systems*, 3(1) :45–60, 2010.
- [11] H Verreist, Kristel Van Acker, Johan Suykens, Bart Motmans, Bart De Moor, and Joos Vandewalle. NI q neural control theory : Case study for a ball and beam system. In *1997 European Control Conference (ECC)*, pages 1303–1308. IEEE, 1997.
- [12] Wen Yu. Nonlinear pd regulation for ball and beam system. *International Journal of Electrical Engineering Education*, 46(1) :59–73, 2009.

# Annexes

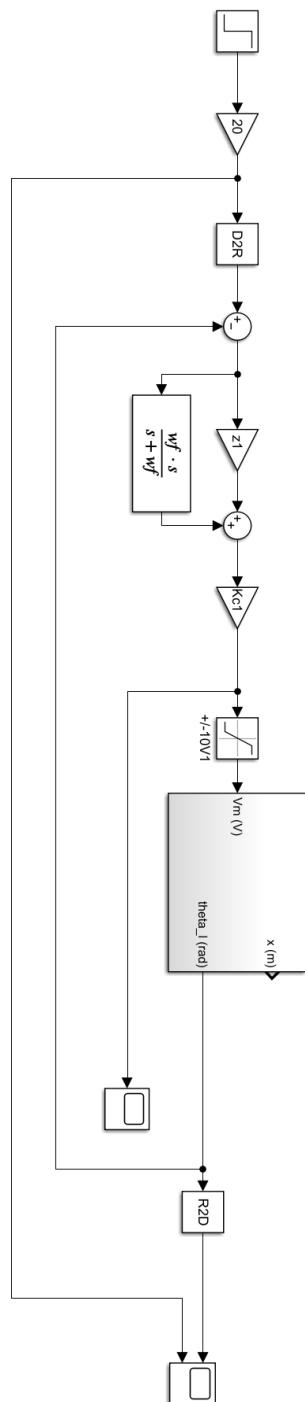


FIGURE 3.17 – Schéma de l'implémentation de la commande fréquentielle du servomoteur

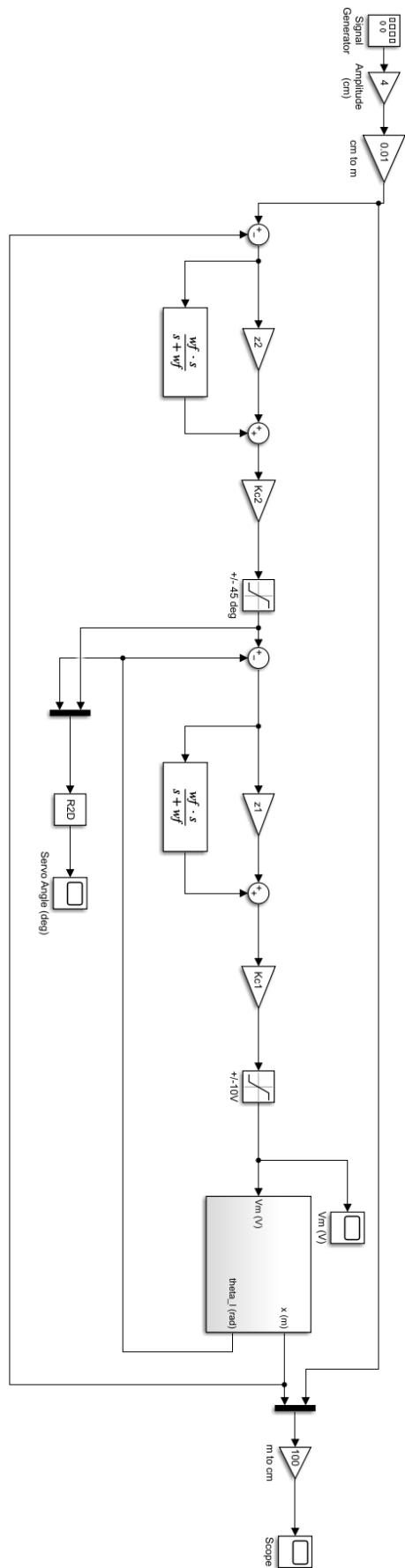


FIGURE 3.18 – Schéma de l'implémentation de la commande fréquentielle du système complet

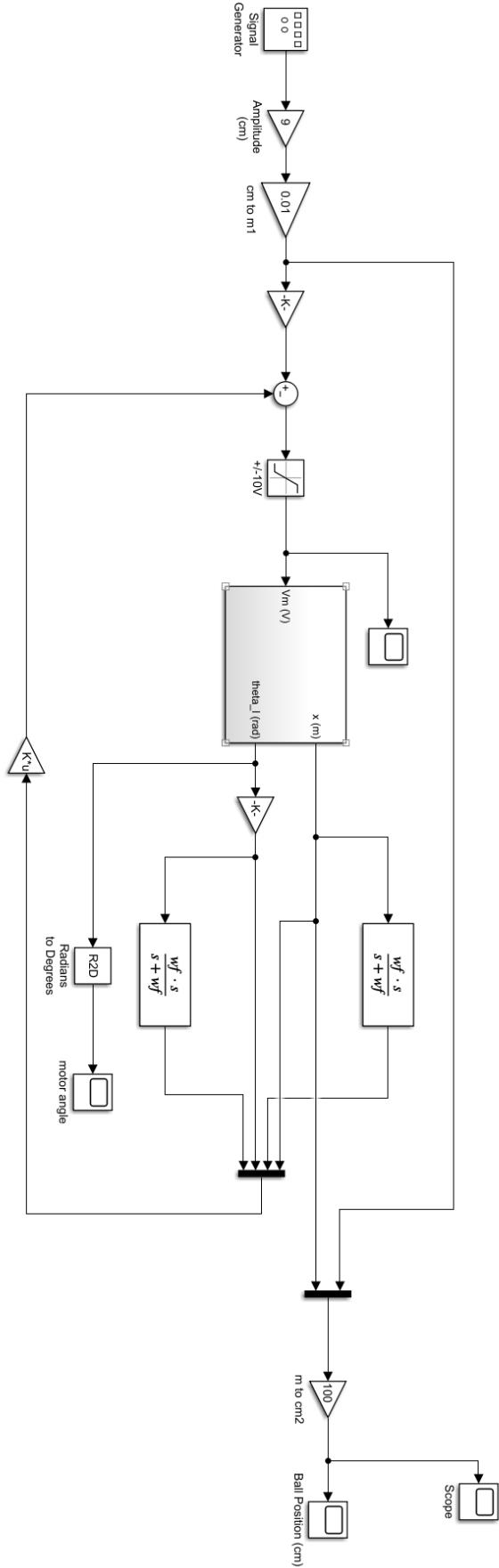


FIGURE 3.19 – Schéma de l'implémentation de la commande par retour d'état

```

1 %% SETUP_BALL_BEAM
2 % Sets the necessary parameters to run the Ball and Beam
3 % experiment.
4 %
5 clear;
6 %
7 %% Servo Configuration
8 % External Gear Configuration: set to 'HIGH' or 'LOW'
9 EXT_GEAR_CONFIG = 'HIGH';
10 % Type of Load: set to 'NONE', 'DISC', or 'BAR'
11 LOAD_TYPE = 'NONE';
12 % Amplifier gain used.
13 % Note: VoltPAQ users: Set K_AMP to 1 and Gain switch on amplifier
14 % to 1
15 K_AMP = 1;
16 % Amplifier Type
17 AMP_TYPE = 'VoltPAQ';
18 % Is servo equipped with tachometer? Set to 'YES' or 'NO'
19 TACH_OPTION = 'NO';
20 %
21 %% Lab Configuration
22 % Type of controller: set it to 'AUTO', 'MANUAL',
23 % CONTROL_TYPE = 'AUTO';
24 CONTROL_TYPE = 'MANUAL';
25 %
26 %% Control specifications
27 % Settling time percentage
28 c_ts = 0.04;
29 % Settling time (s)
30 ts = 3;
31 % Percentage overshoot (%)
32 PO = 5.0;
33 % Ball and beam filter cutoff frequency (rad/s)
34 wf = 2 * pi * 5.0;
35 %
36 %% System Parameters
37 % Sets model variables according to the user-defined system
38 % configuration
39 [ Rm, kt, km, Kg, eta_g, Beq, Jm, Jeq, eta_m, K_POT, K_ENC,
40   VMAX_AMP, IMAX_AMP ] = config_servo( EXT_GEAR_CONFIG,
41   TACH_OPTION, AMP_TYPE, LOAD_TYPE );
42 % Load Ball and Beam model parameters.
43 [ L, r_arm, r_b, m_b, J_b, g, THETA_OFF, THETA_MIN, THETA_MAX,
44   K_BS] = config_bb( );
45 %
46 %% Calculate Control Parameters
47 if strcmp ( CONTROL_TYPE , 'MANUAL' )
48   % Rotary servo position control proportional gain (V/rad)
49   kp = 12;
50   % Calculate Ball and Beam model gain.
51   K_bb = 1;

```

```

46 % Calculate natural frequency and damping ratio given
47 % specifications.
48 wn = 0;
49 zeta = 0;
50 % Design Ball and Beam compensator
51 z = 1;
52 Kc = 0.5;
53 %
54 elseif strcmp ( CONTROL_TYPE , 'AUTO' )
55 % Rotary servo position control proportional gain (V/rad)
56 kp = 12;
57 % Calculate Ball and Beam model gain.
58 [ K_bb ] = d_bb_model_param(r_arm, L);
59 % Design Ball and Beam compensator
60 [ Kc, z ] = d_bb_pd(K_bb, P0, ts, c_ts);
61 end
62 %
63 %% Display
64 disp( ' ' );
65 %
66 disp( 'Ball and Beam model parameter: ' );
67 disp( [ ' K_bb = ' num2str( K_bb, 3 ) ' m/s^2/rad' ] );
68 %
69 disp( 'Ball and Beam Control Specifications: ' );
70 disp( [ ' ts = ' num2str( ts, 3 ) ' s' ] );
71 disp( [ ' P0 = ' num2str( P0, 3 ) ' %' ] );
72 %
73 disp( 'Ball and Beam control: ' );
74 disp( [ ' Kc = ' num2str( Kc, 3 ) ' s/m' ] );
75 disp( [ ' z = ' num2str( z, 3 ) ' rad/s' ] );
76 disp( [ ' wf = ' num2str( wf, 3 ) ' rad/s' ] );
77 %
78
79 %% Parametres et calculs
80 %% Parametres du systemes
81 L_beam = 0.01*42.55; % longueur du rail (m)
82 r_arm = 0.01*2.54; % distance entre l'arbre de sortie du
83 % servo-moteur et l'articulation couplee (m)
84 r_b = 0.01*1.27; % rayon de la bille (m)
85 m_b = 0.064; % masse de la bille (kg)
86 K_bs = -0.01*4.25; % gain du capteur de la position de la bille
87 % (m/V)
88 g = 9.8; % acceleration gravitationnelle (m/s^2)
89 J_b = 2 / 5 * m_b * r_b^2; % moment d'inertie de la bille
90 K_bb = (r_b^2 * r_arm * m_b * g) / (L_beam * (m_b * r_b^2 + J_b));
91 % gain du modele de la bille sur rail
92 % K_bb = (5*r_arm*g)/(7*L_beam)
93 %
94 %% Parametres du modele nominal du servo-moteur
95 K_sr = 1.53; % gain statique (rad/(V.s))

```

```

93 tau = 0.0248; %constante de temps (s)
94
95 F_sr = zpk(tf(K_sr,[tau 1 0])); % fonction de transfer du
   servo-moteur
96
97 %% Fonction de transfert du modele de la bille sur rail
98 F_bb = tf(K_bb,[1 0 0]);
99
100 %% Fonction de transfert du systeme complet en boucle ouverte
101 F_sys = F_sr * F_bb;
102
103 %% Representation d'etat du systeme linearise
104 k1 = 5 * g / 7;
105 k2 = K_sr * r_arm / L_beam;
106
107 A = [0 1 0 0 ;
108      0 0 k1 0 ;
109      0 0 0 1 ;
110      0 0 0 -1/tau];
111 B = [0 ; 0 ; 0 ; k2/tau];
112 C = [1 0 0 0];
113 D = 0;
114
115 sys = ss(A,B,C,D); % modele espace d'etat
116 sys_aug = augstate(sys); % modele espace d'etat augumentee
117
118 P_des = [-2 -3 -20 -21]; % poles desires
119
120 K = acker(A,B,P_des); % gain du retour d'etat
121
122 N = inv(C*inv(-A+B*K)*B);
123
124 sys_BF = ss(A-B*K,B*N,C,D); % modele espace d'etat du systeme en
   boucle fermee
125 F_BF = tf(sys_BF); % fonction de transfer du systeme en boucle
   fermee
126
127 %% Mise en place d'un observateur identite
128 P_obs = [-24 -25 -26 -27]; % poles de l'observateur identite
129
130 G = acker(A', C', P_obs');
131 F = A - G*C;
132 H = B;
133 M = eye(4);
134 N_o = zeros(4, 1);
135
136 A_obs = F;
137 B_obs = [G H];
138 C_obs = M;
139 D_obs = [N_o zeros(4, 1)];
140

```

```

141 sys_obs = ss(A_obs, B_obs, C_obs, D_obs);
142
143 %% Correcteur frequentielle de type proportionnel derivative
144 % (servo-moteur)
144 zeta_sr = 0.65; % facteur d'amortissement desire
145 tr_sr = 0.1; % temps de reponse desire (s)
146
147 wn_sr = -log(0.05*sqrt(1-zeta_sr^2))/(zeta_sr*tr_sr);
148
149 Kc1 = (2*zeta_sr*wn_sr*tau-1)/K_sr;
150 z1 = (wn_sr^2*tau)/(K_sr*Kc1);
151
152 %% Correcteur frequentielle de type prclc
153 zeta_bb = 0.6; % facteur d'amortissement desire
154 tr_bb = 3; % temps de reponse desire(s)
155
156 wn_bb = -log(0.05*sqrt(1-zeta_bb^2))/(zeta_bb*tr_bb);
157
158 Kc2 = (2*zeta_bb*wn_bb)/K_bb;
159 z2 = wn_bb^2/(K_bb*Kc2);
160
161 %% Pulsation de coupure du pseudo-derivateur
162 wf = 2*pi*5;

```