

UNIVERSITÉ DE MONTRÉAL

UTILISATION DE LA DÉCOMPOSITION VIRTUELLE POUR LA
COMMANDE D'UNE MEMBRURE D'UN BRAS ROBOTIQUE FLEXIBLE

MATHILDE CALLOT

DÉPARTEMENT DE GÉNIE ÉLECTRIQUE
ÉCOLE POLYTECHNIQUE DE MONTRÉAL

MÉMOIRE PRÉSENTÉ EN VUE DE L'OBTENTION
DU DIPLOME DE MAÎTRISE ÈS SCIENCES APPLIQUÉES
(MSCA)

DÉCEMBRE 2006



Library and
Archives Canada

Bibliothèque et
Archives Canada

Published Heritage
Branch

Direction du
Patrimoine de l'édition

395 Wellington Street
Ottawa ON K1A 0N4
Canada

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence

ISBN: 978-0-494-29218-1

Our file Notre référence

ISBN: 978-0-494-29218-1

NOTICE:

The author has granted a non-exclusive license allowing Library and Archives Canada to reproduce, publish, archive, preserve, conserve, communicate to the public by telecommunication or on the Internet, loan, distribute and sell theses worldwide, for commercial or non-commercial purposes, in microform, paper, electronic and/or any other formats.

The author retains copyright ownership and moral rights in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

AVIS:

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque et Archives Canada de reproduire, publier, archiver, sauvegarder, conserver, transmettre au public par télécommunication ou par l'Internet, prêter, distribuer et vendre des thèses partout dans le monde, à des fins commerciales ou autres, sur support microforme, papier, électronique et/ou autres formats.

L'auteur conserve la propriété du droit d'auteur et des droits moraux qui protègent cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

In compliance with the Canadian Privacy Act some supporting forms may have been removed from this thesis.

Conformément à la loi canadienne sur la protection de la vie privée, quelques formulaires secondaires ont été enlevés de cette thèse.

While these forms may be included in the document page count, their removal does not represent any loss of content from the thesis.

Bien que ces formulaires aient inclus dans la pagination, il n'y aura aucun contenu manquant.


Canada

UNIVERSITÉ DE MONTRÉAL

ÉCOLE POLYTECHNIQUE DE MONTRÉAL

Ce mémoire intitulé:

UTILISATION DE LA DÉCOMPOSITION VIRTUELLE POUR LA
COMMANDE D'UNE MEMBRURE D'UN BRAS ROBOTIQUE FLEXIBLE

présenté par: CALLOT Mathilde

en vue de l'obtention du diplôme de: Maîtrise ès sciences appliquées

a été dûment accepté par le jury d'examen constitué de:

M. HURTEAU Richard, D.Ing., président

M. GOURDEAU Richard, Ph.D., membre et directeur de recherche

M. ZHU Wen-Hong, Ph.D., membre et codirecteur de recherche

M. PIEDBOEUF Jean-Claude, Ph.D., membre

à tout un tas de monde

REMERCIEMENTS

Je remercie tout d'abord mon directeur de recherche, Richard Gourdeau, de m'avoir aidé à réaliser un rêve : mettre un pied dans le domaine spatial avant la fin de mes études. Avec son aide j'ai pu effectuer ma recherche à l'Agence Spatiale Canadienne et rencontrer ainsi des gens passionnés par leur métier. Malgré la distance il a toujours été à l'écoute de mes problèmes, ce qui m'a permis de mener à bien ce travail.

Je remercie également mon co-directeur de recherche, Wen-Hong Zhu, pour son sujet qui m'a permis de travailler à l'Agence Spatiale. Ses différents conseils m'ont aidé à bien orienter ma recherche. Travailler dans le milieu industriel pendant ma recherche, m'a fait entrer en douceur dans le monde du travail et découvrir ces bons et ces mauvais côtés. Merci aussi à Christian Lange de m'avoir fourni les explications et les programmes qui ont alimenté mes simulations. Merci également à Erik Dupuis pour l'aide qu'il m'a apportée pour passer au travers des méandres administratifs.

Je remercie aussi tous les autres, qui ont croisé mon chemin pendant ces deux ans pour m'avoir aidée autant qu'ils pouvaient. Grâce à eux, j'ai pu sortir le nez de ma maîtrise et profiter de la vie parascolaire pour mieux me remettre au travail ensuite.

RÉSUMÉ

L'objectif de cette recherche est de concevoir un contrôleur pour un bras robotique flexible libre de tous mouvements plans. L'étude d'un simple bras flexible plutôt que de l'ensemble d'un robot complexe est rendue possible par la décomposition virtuelle. En effet un tel concept permet de découper un robot complexe en plusieurs *objets* et *chaînes ouvertes* et d'en faire le contrôle individuellement. On suppose que le couplage dynamique entre deux objets est entièrement représenté par le *flux virtuel de puissance* ce qui rend les objets indépendants entre eux. C'est pourquoi notre étude se limite à un seul bras flexible.

Le principe de décomposition virtuelle nous dit également que la meilleure méthode pour contrôler un *objet* est un pseudo contrôle en mode de glissement. C'est ce que nous avons appliqué lors du développement de nos contrôleurs. Dans le but de contrôler un large éventail de bras (flexibles à rigides) nous avons développé trois contrôleurs.

Les performances de ces différents contrôleurs sont énoncées dans la dernière partie de ce mémoire. Afin de mieux comprendre leurs avantages et inconvénients nous les avons comparés deux à deux. De plus pour bien visualiser le mouvement de la barre dans chacun des cas, nous avons effectué une animation de la barre en déformation.

ABSTRACT

The goal of this study is to design a controller for a flexible free beam in plane movement. We study just a simple flexible beam instead of a complex robot with the help of virtual decomposition. Such a concept states that a complex robot can be decomposed in several *objects* and *open chains*. And each item can be controlled individually. Note that the dynamic coupling between every two objects is completely represented by the *virtual power flow* in the sense of L_2 stability. It makes the objects independent one of each others. That's why we limit this study to only one flexible beam.

The principle of the virtual decomposition implies that a suitable method to control an *object* is pseudo sliding mode control. We have developed our controllers with this method. In order to control a large selection of beams with various flexibility, we have developed three controllers.

The last part of this project deals with the performances of these controllers. In order to understand well their advantages and their disadvantages we have compared them in pairs. Moreover, in order to visualise the movement of the beam in each case we have carried out an animation of the beam's deformation.

TABLE DES MATIÈRES

DÉDICACE	iv
REMERCIEMENTS	v
RÉSUMÉ	vi
ABSTRACT	vii
TABLE DES MATIÈRES	viii
LISTE DES FIGURES	xiii
LISTE DES NOTATIONS ET DES SYMBOLES	xvi
LISTE DES TABLEAUX	xvii
INTRODUCTION	1
CHAPITRE 1 REVUE DE LITTÉRATURE	3
1.1 Introduction	3
1.2 Manipulateur virtuel	4
1.2.1 Décomposition virtuelle	5
1.3 Mouvement et commande	6
1.3.1 Mouvement	6
1.3.2 Commande	7
1.3.2.1 Commande à l'aide des quaternions	7
1.3.2.2 Commande par apprentissage	7
1.3.2.3 Commande par la technique des perturbations sin- gulières	8
1.3.2.4 Commande aux deux extrémités	9

1.3.2.5	Contrôleur par commande adaptative	10
1.3.2.6	Adaptation de la commande au contexte de notre problème	10
1.4	Conclusion	11
CHAPITRE 2 DÉCOMPOSITION VIRTUELLE		12
2.1	Introduction	12
2.2	Les grandes lignes de la décomposition virtuelle	13
2.2.1	Objets	14
2.2.2	Chaînes ouvertes	14
2.2.3	Lien entre objets et chaînes ouvertes	15
2.3	Préliminaire pour le contrôle	16
2.3.1	Force et vitesse	16
2.3.2	Variables intermédiaires	17
2.4	Flux virtuel de puissance	18
2.4.1	Définition	18
2.4.2	Spécifications du flux virtuel de puissance	19
2.5	Stabilité	20
2.6	Conclusion	21
CHAPITRE 3 ENONCÉ ET SOLUTION STABLE DU PROBLÈME . .		22
3.1	Situation du problème	22
3.1.1	Système	22
3.1.1.1	Equation de la dynamique	24
3.1.1.2	Conditions aux limites	25
3.1.2	Contrôleur	25
3.2	Contrôleur choisi	26
3.2.1	Première partie du contrôleur	27
3.2.1.1	Contrôleur par mode de glissement	27

3.2.2	Notre contrôleur	27
3.2.3	Deuxième partie du contrôleur : propagation de la commande	29
3.2.3.1	Discrétisation de la poutre	29
3.2.3.2	Obtention des valeurs de commande	29
3.3	Stabilité du contrôleur	34
3.3.1	Equation de la dynamique de référence	34
3.3.2	Fonction de Lyapunov	35
3.3.3	Dérivée de la fonction de Lyapunov	35
3.4	Conclusion	38
CHAPITRE 4	PREMIERS CODES DU CONTRÔLEUR	40
4.1	Premier code de commande simple	40
4.1.1	Premier choix d'équation pour le contrôleur	40
4.1.1.1	Contrôleur en pseudo mode de glissement	40
4.1.1.2	Boucle de propagation	41
4.1.2	Stabilité du système avec ce contrôleur en simulation	42
4.1.2.1	Constantes du contrôleur	42
4.1.2.2	Simulation	42
4.1.3	Commande de la position angulaire	45
4.1.3.1	Modification de notre contrôleur	45
4.1.3.2	Contrôleur en pseudo mode de glissement	46
4.1.3.3	Boucle de propagation	47
4.1.4	Résultats et limites de ce contrôleur	48
4.1.4.1	Variables et leurs effets	48
4.1.4.1.1		48
4.1.4.1.2		48
4.1.4.1.3		49
4.1.4.2	Limites de ce modèle	51

4.1.5	Conclusion	55
4.2	Deuxième code du contrôleur	55
4.2.1	De nouvelles équations de commandes	56
4.2.1.1	Contrôleur en pseudo mode de glissement	56
4.2.1.2	Boucle de propagation	56
4.2.1.3	Changement des équations liées à la force	57
4.2.1.4	Changement du moment de commande	58
4.2.2	Résultats et limites de ce modèle	59
4.2.3	Conclusion	64
CHAPITRE 5	TROISIÈME CODE DU CONTRÔLEUR	65
5.1	Equations d'état d'un système réel	66
5.2	Rappel sur la méthode de Kalman	67
5.2.1	Méthode de prédiction-correction	68
5.2.2	Gains du filtre	69
5.2.3	Matrices de covariance des bruits	70
5.2.4	Procédure du filtre de Kalman	70
5.3	Application de cette méthode à notre contrôleur	71
5.3.1	Nouvelles équations	71
5.3.1.1	Equations du système	71
5.3.1.2	Boucle de propagation	73
5.4	Résultats et limites de ce contrôleur	77
5.4.1	Modèle très flexible	77
5.4.2	Modèle très rigide	80
5.5	Conclusion	82
CHAPITRE 6	COMPARAISONS	83
6.1	Trois exemples	83
6.1.1	Barre très flexible	83

6.1.2	Antenne de satellite	83
6.1.3	Robot spatial typique	84
6.2	Comparaisons sur des simulations idéales	84
6.2.1	Robot spatial typique	85
6.2.2	Antenne typique	88
6.2.3	Barre très flexible	91
6.3	Comparaisons sur la simulation d'une situation réelle	93
6.3.1	Reconstitution de la vitesse	94
6.3.1.1	Premier code du contrôleur	94
6.3.1.2	Deuxième code du contrôleur	95
6.3.1.3	Troisième code du contrôleur	95
6.3.2	Ajout de bruit sur la position	96
6.3.2.1	Premier code du contrôleur	98
6.3.2.2	Deuxième code du contrôleur	98
6.3.2.3	Troisième code du contrôleur	100
6.4	Conclusion	100
CONCLUSION		102
RÉFÉRENCES		103

LISTE DES FIGURES

Figure 2.1	Exemple de robots de grandes dimensions	13
Figure 2.2	Schéma illustrant la décomposition des robots de la figure 2.1	14
Figure 2.3	Exemple de <i>chaîne ouverte</i>	15
Figure 2.4	Schéma des <i>flux virtuels de puissance</i> pour un robot décom- posé	20
Figure 3.1	Schéma du mouvement de la barre	23
Figure 3.2	Schéma du système complet	23
Figure 3.3	Détails du contrôleur	26
Figure 3.4	Discrétisation de la barre	29
Figure 3.5	Propagation de l'information de commande de la fin vers de le début de la barre discrétisée	31
Figure 4.1	Profil d'accélération désirée pour la première loi de com- mande	43
Figure 4.2	Exemple de mouvement d'une barre en simulation	43
Figure 4.3	Erreur de position et position angulaire initiale avec le pre- mier contrôleur sans contrôleur d'orientation	44
Figure 4.4	Théorie des petits angles	46
Figure 4.5	Erreur de position linéaire avec le premier contrôleur et une barre très rigide ($EI = 10000N.m^2$)	50
Figure 4.6	Erreur de position angulaire avec le premier contrôleur et une barre très rigide ($EI = 10000N.m^2$)	50
Figure 4.7	Erreur de position linéaire avec le premier contrôleur et dif- férentes barres	52
Figure 4.8	Erreur de position angulaire avec le premier contrôleur et différentes barres	52

Figure 4.9	Erreur de position linéaire avec le premier contrôleur et sa limite	54
Figure 4.10	Erreur de position angulaire avec le premier contrôleur et sa limite	54
Figure 4.11	Profil d'accélération suivi par les deux dernières lois de commande	59
Figure 4.12	Erreur de position linéaire avec le deuxième contrôleur et une barre très rigide	60
Figure 4.13	Erreur de position angulaire avec le deuxième contrôleur et une barre très rigide	60
Figure 4.14	Erreur de position linéaire avec le deuxième contrôleur pour différentes barres	61
Figure 4.15	Erreur de position angulaire avec le deuxième contrôleur pour différentes barres	62
Figure 4.16	Erreur de position linéaire avec le deuxième contrôleur pour différentes barres	63
Figure 4.17	Erreur de position angulaire avec le deuxième contrôleur pour différentes barres	63
Figure 5.1	Schéma d'un système réel	66
Figure 5.2	Erreur de position linéaire d'une barre très flexible ($EI = 1N.m^2$) avec le troisième contrôleur	78
Figure 5.3	Erreur de position angulaire d'une barre très flexible avec le troisième contrôleur ($EI = 1N.m^2$)	79
Figure 5.4	Erreur de position linéaire d'un robot spatial typique avec le troisième contrôleur	81
Figure 5.5	Erreur de position angulaire d'un robot spatial typique avec le troisième contrôleur	81

Figure 6.1	Erreur de position linéaire du robot spatial typique avec les trois contrôleurs	86
Figure 6.2	Erreur de position angulaire du robot spatial typique avec les trois contrôleurs	86
Figure 6.3	Détails de l'erreur de position angulaire du robot spatial typique avec les trois contrôleurs	87
Figure 6.4	Erreur de position linéaire de l'antenne avec les deux derniers modèles	89
Figure 6.5	Erreur de position angulaire de l'antenne avec les deux derniers modèles	90
Figure 6.6	Erreur de position linéaire de la barre très flexible	92
Figure 6.7	Erreur de position angulaire de la barre très flexible	92
Figure 6.8	Erreur de position linéaire du robot spatial typique avec reconstitution de la vitesse et le troisième code du contrôleur	95
Figure 6.9	Erreur de position angulaire du robot spatial typique avec reconstitution de la vitesse et le troisième code du contrôleur	96
Figure 6.10	Erreur de position linéaire d'un bras très rigide avec bruitage de la mesure et le premier code du contrôleur	97
Figure 6.11	Erreur de position angulaire d'un bras très rigide avec bruitage de la mesure et le premier code du contrôleur	97
Figure 6.12	Erreur de position linéaire de l'antenne avec bruitage de la mesure et le troisième code du contrôleur	99
Figure 6.13	Erreur de position angulaire de l'antenne avec bruitage de la mesure et le troisième code du contrôleur	99

LISTE DES NOTATIONS ET DES SYMBOLES

$(\cdot)_r$:	variable/vecteur requis
$(\cdot)_d$:	variable/vecteur désirée ou de référence
$(\cdot)_B$:	variable/vecteur situé au début d'un corps
$(\cdot)_T$:	variable/vecteur situé à la fin d'un corps
F :	vecteur force
τ :	vecteur moment
$\omega(x, t)$:	variable de déformation de la poutre au point x et au temps t
$\omega_0(x, t)$:	déformation de la poutre au point 1
$\theta_0(x, t)$:	angle entre la tangente au premier point de la poutre et l'horizontale
W :	<i>flux virtuel de puissance</i>
FVP :	Flux Virtuel de Puissance
q_{jk} :	déplacement en position du joint k de la j^{ieme} chaîne ouverte
X :	vitesse linéaire/angulaire généralisée
χ_j :	vecteur vitesse de la j^{ieme} chaîne ouverte $(= [\dot{q}_j^T \quad X^T]^T)$
n_o :	nombre d' <i>objets</i>
n_c :	nombre de <i>chaînes ouvertes</i>
ddl :	degré(s) de liberté

LISTE DES TABLEAUX

Tableau 4.1	Paramètres du premier contrôleur pour une barre de rigidité $EI = 10000N.m^2$	49
Tableau 4.2	Paramètres du premier contrôleur pour différentes barres .	51
Tableau 4.3	Paramètres du deuxième contrôleur pour différentes barres	61
Tableau 5.1	Paramètres du troisième contrôleur pour une barre très flexible	78
Tableau 5.2	Paramètres du troisième contrôleur pour un robot spatial typique	80
Tableau 6.1	Paramètres des contrôleurs pour le robot spatial typique . .	85
Tableau 6.2	Performances des deux derniers contrôleurs pour le robot spatial typique	87
Tableau 6.3	Paramètres des deux derniers contrôleurs pour l'antenne . .	89
Tableau 6.4	Performances des différentes commandes sur le modèle antenne	89
Tableau 6.5	Paramètres des deux derniers contrôleurs pour la barre très flexible	91
Tableau 6.6	Caractéristiques des deux derniers contrôleurs pour la barre très flexible	91

INTRODUCTION

En robotique spatiale, les structures légères ou constituées de long bras présentent des modes flexibles qui peuvent entraîner des problèmes. En limitant le design de la commande du robot à des modèles rigides, les vitesses d'opération sont sévèrement limitées. Souvent l'opérateur du robot doit attendre que les modes flexibles du robot s'amortissent avant d'exécuter le mouvement suivant.

Plus le robot a de segments flexibles, plus la complexité du couplage dynamique entre les segments est grande. Jusqu'à maintenant les contrôleurs qui ont eu le plus de succès sont basés sur des modèles simples, par exemple un robot avec un ou deux bras flexibles dans un mouvement planaire. Afin de commander un robot avec de multiples segments flexibles, on peut utiliser la décomposition virtuelle, permettant de considérer chaque segment du robot séparément et d'en étudier la dynamique indépendamment. Dans notre travail, nous allons étudier un segment flexible. Celui-ci sera un sous-ensemble dans le problème de commande d'un robot plus complexe contenant des bras flexibles.

Dans un premier temps nous allons étudier la dynamique de déformation de la poutre. Puis nous allons l'analyser et en effectuer la commande, c'est-à-dire trouver un contrôleur adapter à cette dynamique. Nous allons vérifier notre analyse par voie de simulation.

Le but de cette recherche est de contrôler la trajectoire d'un bras robotique flexible et la force qu'il applique sur l'environnement. La déformation de la poutre à son extrémité est la variable de contrôle. La trajectoire suivie est une trajectoire de déformation.

Cette déformation nous sert également à calculer la force et le moment à injecter

en début de poutre pour que la fin de la poutre suive la trajectoire désirée.

Après une revue de littérature qui mettra en avant les différents contrôleurs utilisés pour le contrôle de bras flexible, nous allons présenter la décomposition virtuelle. Puis dans les chapitres 3, 4 et 5 différentes solutions de contrôle seront développées. Une étude en simulation des différents contrôleurs permettra de dégager leurs avantages et inconvénients.

CHAPITRE 1

REVUE DE LITTÉRATURE

1.1 Introduction

Depuis une vingtaine d'année, la modélisation et la commande de manipulateurs robotiques légers ont inspiré de nombreux chercheurs. Dès 1986, Simo et Vu-quoc (Simo et Vu-Quoc, 1986) s'intéressent à la dynamique de déformation de poutres flexibles. Ils analysent notamment le mouvement d'une barre dont aucune de ses extrémités n'est fixée, soumise à une force et un moment à un de ces bouts. En 1987, Kane et al. étudient dans (Kane et al., 1987) la dynamique d'une poutre en porte à faux accrochée à une base mobile. Ils montrent que la barre possède une certaine flexibilité dont il faut tenir compte dans la dynamique du mouvement. En 1991, Morgül (Morgül, 1991) établit un contrôleur pour le type de barre étudié dans (Kane et al., 1987) en mouvement plan de rotation. Au fur et à mesure des années, la diversité des approches pour la commande de robots flexible se multiplie : en 1993, la méthode des perturbations singulières est utilisée par Bouaziz, Dochain, Piedboeuf et Hurteau (Bouaziz et al., 1993), en 1998 dans (Sun et Mills, 1998), Sun et Mills se servent de capteurs piézo-électriques permettant d'avoir accès à la déformation réelle de la barre, en 2000 Garcia et Feliu (Garcia et Feliu, 2000) utilisent le caractère flexible du manipulateur pour mieux commander les tâches en contact avec l'environnement. Ceci ne constitue que quelques exemples des nombreuses recherches qui ont été effectuées sur la commande d'un ou plusieurs bras flexible.

Actuellement un des problèmes d'intérêt est de pouvoir commander un robot léger

donc possédant plusieurs segments flexibles. Cependant la tâche est rendue ardue par la déformation élastique des segments ainsi que par les vibrations en bout de segment causées par la flexibilité. Yang, Xu et Tso (Yang et al., 2001) se sont servis de capteurs à laser optique pour mesurer la déformation de chaque segment flexible. Puis ils ont attaqué le problème en utilisant une approche de compensation d'erreur. Celle-ci permet d'améliorer le positionnement et le suivi de trajectoire de l'extrémité du robot flexible. Sun (Sun, 2002) également étudie la commande de manipulateurs possédant de multiples segments flexibles. Comme lui, de nombreux auteurs (Senda et Murotsu, 2000), (Bouaziz et al., 1993), (Jonkers et Aarts, 2001), ont décidé de partir d'une commande existante pour des segments rigides et de l'adapter aux segments flexibles. Pour cela, il faut ajouter à la commande une deuxième composante tenant compte de la flexibilité du segment. L'étude de la commande de robots manipulateurs possédant de multiples segments flexibles reste tout de même complexe. Ceci a entraîné les chercheurs à développer de nouvelles techniques pour mieux représenter le robots, tel le manipulateur virtuel (*Virtual Manipulator - VM*)

1.2 Manipulateur virtuel

En même temps qu'on a cherché à commander des robots plus légers et donc plus flexibles, on a commencé à développer des techniques pour mieux commander un robot complexe sans attaches particulières. Une des ces techniques est le concept de manipulateur virtuel repris par de nombreux chercheurs. Il s'agit d'une chaîne cinétique idéale connectée à la base du robot et reliée à certains points choisis du manipulateur qui coïncideront toujours ensemble. Dubowsky et Papadopoulos (Dubowsky et Papadopoulos, 1993) montrent l'utilité du manipulateur virtuel dans la commande d'un robot spatial. Walker, dans (Walker, 1990), utilise ce concept

pour hiérarchiser une structure complexe. Afin de simplifier l'étude, Walker crée des séquences uniques de segments en introduisant le concept de segment connecteur. Ce segment fictif permet de créer des séquences uniques lorsque plusieurs segments sont connectés au même segment. A l'aide de ce concept, Walker met en place un arbre binaire qui contient toutes les séquences. Ceci facilite l'implantation de l'algorithme de commande.

L'idée d'un manipulateur virtuel a stimulé d'autres recherches, notamment Zhu et al. qui ont développé le concept de décomposition virtuelle (Zhu et al., 1997).

1.2.1 Décomposition virtuelle

Afin de simplifier l'étude d'un robot multi-bras, donc complexe, Zhu, Xi, Zhang et De Schutter ont développé un concept qui permet de décomposer ce robot en plusieurs objets et chaînes ouvertes. Chaque item peut être étudié séparément et surtout la commande peut être effectuée en rapport avec chaque objet séparément. En effet, il existe un *flux virtuel de puissance* entre deux objets qui prend en compte le couplage entre ceux-ci. Ce concept sera expliqué plus loin. Cela permet de découpler dynamiquement deux sous-systèmes connectés physiquement et donc d'isoler un objet. Ce dernier peut alors être commandé indépendamment des autres ce qui rend l'analyse du système plus simple. Ce principe sera présenté plus en détail au chapitre 2. L'intérêt d'une telle approche est que l'algorithme de commande pourra être implanté sous forme de module de données pour chaque objet et chaîne ouverte. Ceci permettra de former un contrôleur adapté à une application spécifique.

Dans ce cas, on peut ne considérer qu'un seul objet simple, tel qu'un segment flexible ; étudier sa dynamique et effectuer sa commande. Puis le replacer dans le

système plus complexe. Nous nous intéresserons donc qu'à un seul bras flexible.

1.3 Mouvement et commande

1.3.1 Mouvement

Comme énoncé précédemment, les bras flexibles ont été étudiés depuis une vingtaine d'année. En particulier pour des raisons de simplification, la plupart des études se sont portées sur des structures simples, c'est-à-dire un seul segment flexible à un degré de liberté (ddl) dans un mouvement planaire. Très souvent, lorsqu'il s'agit d'un segment flexible, on étudie son mouvement en rotation autour d'un axe, tel est le cas dans (Queiroz et al., 1999), (Bouaziz et al., 1993), (Kane et al., 1987), (Morgül, 1991), (Morgül, 1992), (Martins et al., 2002), (Sun et Mills, 1998). La plupart de ces articles implique une équation de la dynamique et des conditions aux limites spécifiques. Par exemple dans (Sun et Mills, 1998), les auteurs simplifient l'équation de la dynamique en négligeant les déformations axiales ainsi que les effets de la gravité et de la dynamique du moyeu qui entraîne la barre. De plus les conditions aux limites sont spécifiques aux problèmes à savoir pas de mouvement au bout accroché au moyeu et des force/moment en fin de poutre mesurés par un capteur piézo-électrique placé en fin de poutre. Ceci implique des conditions aux limites non nulles. En revanche dans (Martins et al., 2002), bien que l'équation de la dynamique ait la même expression que dans (Sun et Mills, 1998), car le mouvement de la barre est le même, les conditions aux limites sont différentes car le problème est posé différemment. En effet, dans ce cas, il n'y a aucun capteur accroché à la fin de la barre, les conditions aux limites sont donc nulles.

Afin d'étendre le domaine d'application de notre recherche et de faciliter la tâche, nous allons considérer un problème plus général : la commande du mouvement

d'une poutre libre, c'est-à-dire à 6ddl. Ceci entraîne une équation de la dynamique spécifique et des conditions aux limites particulières à ce problème. En effet elle est beaucoup plus simple que dans tous les exemples énoncés ci-dessus puisqu'elle ne comporte que deux termes. On étudiera le mouvement de la barre dans le plan.

L'ouvrage de Léonard Meirovitch (Meirovitch, 1997) nous fournit l'équation de la dynamique ainsi que les conditions aux limites dans notre cas. Ceci sera détaillé plus loin dans le mémoire.

1.3.2 Commande

A l'inverse de la configuration quasiment unique étudiée, beaucoup de stratégies ont été développées pour commander le mouvement d'une barre flexible.

1.3.2.1 Commande à l'aide des quaternions

Dans (Wen et Kreutz, 1991), Wen et Delgado développent différents contrôleurs à l'aide des quaternions unitaires. Ces contrôleurs sont tous de la même forme : Proportionnel Dérivé par rétroaction avec une compensation par rétroaction positive. La stabilité de tous leurs contrôleurs est démontrée par une méthode d'énergie de Lyapunov. La bonne performance des différents contrôleurs est illustrée par simulation.

1.3.2.2 Commande par apprentissage

Une autre méthode possible pour la commande de trajectoire est la méthode par apprentissage. Celle-ci est présentée dans (Rhim et al., 2001). Ici l'objectif est

de commander un manipulateur flexible à l'aide de la commande en forme. Cette dernière nécessite un système linéaire ou linéarisé. Comme les non linéarités engendrent une détérioration des performances dans la commande en forme, les auteurs utilisent cette méthode en conjonction avec un contrôleur par mode d'apprentissage. Ce contrôleur permet de compenser les erreurs répétitives causées par les non linéarités lors du suivi de trajectoire. Cependant cette commande n'est pertinente que lors de l'exécution de tâches répétitives.

1.3.2.3 Commande par la technique des perturbations singulières

Une autre méthode développée dans (Bouaziz et al., 1993) consiste à dégager une commande basée sur un modèle rigide ne prenant en compte les effets de la flexibilité qu'en fin de bras. Les auteurs considèrent une barre basée sur le modèle d'Euler-Bernouilli accrochée à un axe tournant. Afin de modéliser correctement le système à l'aide de la méthode des perturbations singulières, Bouaziz, Dochain, Piedboeuf et Hurteau choisissent le paramètre de perturbation comme étant inversement proportionnel au carré du premier mode de fréquence des oscillations de la barre. Ce paramètre mesure l'importance relative de la rigidité de la barre. Ainsi Bouaziz et al. montrent que plus ce paramètre est petit, plus le robot flexible tend à devenir un robot rigide. Donc si on suppose ce paramètre suffisamment petit, la dynamique de la barre est correctement représentée par un modèle rigide accompagné d'une estimation de la flexibilité au bout de la poutre. Dans ce cas il est aisé de diminuer les effets de la flexibilité en bout de barre par une correction par rétroaction.

Avec cette méthode, il est donc possible de commander un segment flexible à partir de commandes de segments rigides déjà existantes. Il suffit d'ajouter à la commande une deuxième composante tenant compte de la flexibilité du segment. Cette ap-

proche a été reprise par d'autres chercheurs, notamment dans (Jonkers et Aarts, 2001). Jonkers et Aarts découpent la commande en deux parties : un système rigide décrit le mouvement nominal du segment rigide et un système linéaire, linéarisé autour de la trajectoire nominale, décrit le mouvement vibratoire. L'algorithme de commande employé contient alors deux boucles : une ouverte et une fermée. La commande en boucle ouverte produit le mouvement désiré du segment rigide, alors que la boucle fermée doit stabiliser et amortir les vibrations élastiques qui sont naturellement créées par le mouvement du segment rigide. Cependant cette méthode nécessite l'accès à des équations linéarisées de façon précise ainsi qu'un nombre de points de linéarisation assez conséquent.

1.3.2.4 Commande aux deux extrémités

De la même façon que la méthode des perturbations singulières suppose une division de la commande en deux parties, certains auteurs, ont choisi d'effectuer la commande aux deux bouts de la barre afin de commander l'orientation ou la position tout en assurant la stabilisation d'un système. Dans les articles (Queiroz et al., 1999) et (Morgül, 1991) la stratégie de commande est du même type. Par exemple dans (Queiroz et al., 1999), on souhaite commander en position ainsi que la rapidité d'amortissement de la déformation de l'extrémité libre, la commande est donc composée d'un couple limite de commande appliqué à l'actionneur et d'une force limite de commande appliquée à l'extrémité libre de la poutre. Une telle commande nécessite la connaissance du déplacement de l'extrémité libre, sa déformation, la position angulaire de l'actionneur et la dérivée de ces quantités. A partir de ces données Queiroz et al. retiennent deux stratégies pour la commande : une première simple pour commander asymptotiquement la position désirée et une deuxième plus élaborée, un contrôleur adaptatif pour prendre en compte les incertitudes du modèle. Après démonstration, leurs deux contrôleurs sont stables

théoriquement. Par expérimentation, les auteurs mettent en évidence que le contrôleur par commande adaptative donne de meilleurs résultats pour des modèles aux paramètres peu ou mal connus.

1.3.2.5 Contrôleur par commande adaptative

Dans (Queiroz et al., 1999) Quiroz et al. montrent qu'une commande adaptative est efficace pour contrer la déformation de l'extrémité libre de la barre tout en amenant l'actionneur à une position désirée. En plus de stabiliser efficacement le système, cette commande peut être plus avantageuse en présence de paramètres peu ou pas connus. Une telle commande est basée sur la commande adaptative de Slotine-Li développée en 1987 ((Slotine et Li, 1987), (Slotine et Li, 1988)). Cette commande présente l'avantage d'être simple et robuste face aux incertitudes du modèle. On constate également que de nombreux chercheurs ont étudié cette commande (par exemple (Sadegh et Horowitz, 1990), (Walker, 1990)).

Le fait d'être robuste face aux incertitudes du modèle est une caractéristique intéressante dans notre cas, car les seules données accessibles sont en bout de bras et tous les paramètres de la barre sont connus de façon approximative. Une commande adaptative apparaît donc comme un choix judicieux pour notre problème.

1.3.2.6 Adaptation de la commande au contexte de notre problème

D'après l'article (Zhu et al., 1997), il faut adapter notre commande à l'existence de *flux virtuel de puissance*. Zhu et al. suggèrent d'effectuer une commande introduisant des variables intermédiaires qui prendraient en compte ce flux. Cette variable intermédiaire a une expression similaire à celle proposée par Slotine et Li. De même que pour la commande adaptative, Zhu et al. suggèrent l'utilisation

d'une commande Proportionnelle Dérivée par rétroaction à laquelle on ajoute une compensation par rétroaction positive avec une adaptation des paramètres inconnus du système. Les auteurs montrent par une analyse de l'énergie de Lyapunov qu'une telle commande est stable.

Cependant pour l'application à notre modèle et sa simplicité de commande, on ne retiendra que la partie faisant intervenir les variables intermédiaires pour bien prendre en compte le *flux virtuel de puissance*. De façon similaire cette commande sera stable à cause du flux virtuel de puissance et ceci sera prouvé ultérieurement.

1.4 Conclusion

Comme nous avons décidé d'appliquer le concept de décomposition virtuelle, nous limiterons notre étude à la commande d'un seul segment flexible. Pour que l'étude soit la plus générale possible, nous considérons que la barre possède 6 ddl et qu'elle n'est soutenue à aucune des ses extrémités.

Le cadre de travail étant la décomposition virtuelle, nous tiendrons compte du principe de *flux virtuel de puissance* lors du développement de notre contrôleur. Ceci nous amènera à en développer plusieurs afin de trouver le contrôleur le mieux adapté au problème.

CHAPITRE 2

DÉCOMPOSITION VIRTUELLE

2.1 Introduction

L'objectif de la décomposition virtuelle telle qu'exposée dans l'article (Zhu et al., 1997) est de développer un cadre de contrôle général pour les systèmes robotiques généralisés de grandes dimensions. L'idée est de décomposer virtuellement le système robotique à grandes dimensions en plusieurs *objets* et *chaînes ouvertes*. Ainsi le problème de contrôle du système originel est converti en un problème de contrôle de chaque *objet* et *chaîne ouverte* respectivement. Seule la dynamique du sous-système est requise pour le développement du contrôleur.

Le principal avantage d'une telle approche est qu'on peut implanter l'algorithme de contrôle de chaque sous système de façon modulaire dans un processeur.

En utilisant le concept de décomposition virtuelle, le développement de la commande ainsi que l'analyse de stabilité sont grandement simplifiées. Un résultat important est qu'au sens de Lyapunov, le couplage dynamique entre deux sous-systèmes physiquement connectés est entièrement représenté par le *flux virtuel de puissance* aux points de découpage entre eux. Le concept de *flux virtuel de puissance* évoque le principe de "travail virtuel" utilisé par Walker dans (Walker, 1990). Il permet également une interprétation plus intrinsèque du couplage dynamique entre les *objets* et les *chaînes ouvertes*. C'est un outil mieux adapté.

Nous allons tout d'abord présenter les grandes lignes de la décomposition virtuelle et introduire quelques préliminaires pour l'utiliser dans un contrôleur. Ensuite nous

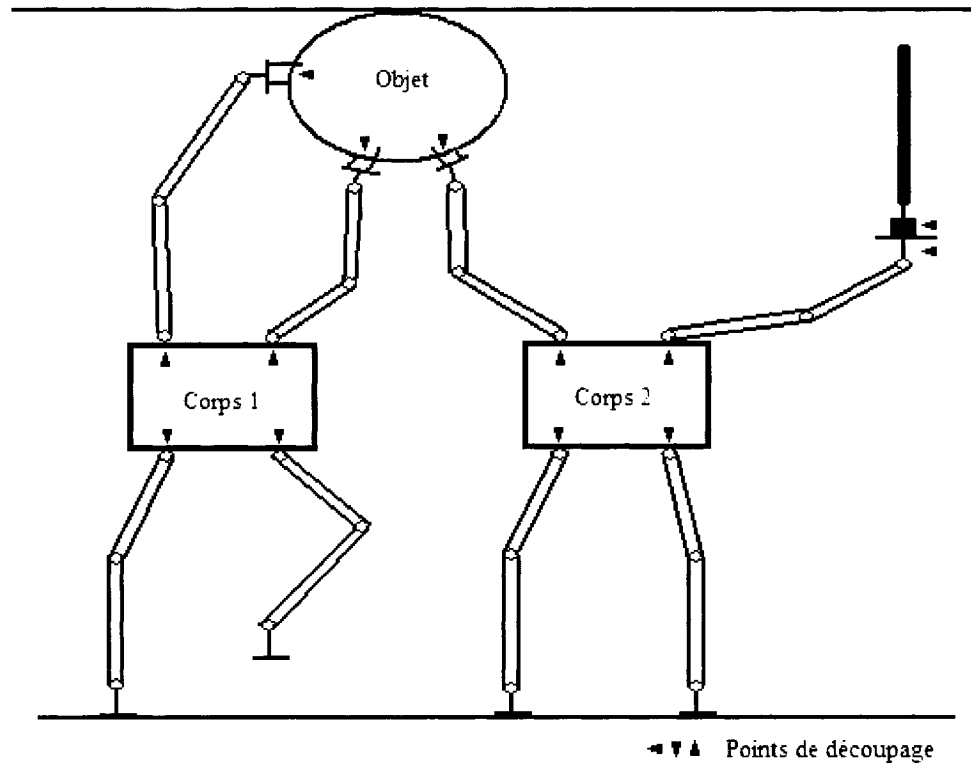


Figure 2.1 Exemple de robots de grandes dimensions

allons expliquer le principe de *flux virtuel de puissance*. Enfin, nous montrerons la stabilité et la convergence des systèmes utilisant cette approche par une analyse de Lyapunov.

2.2 Les grandes lignes de la décomposition virtuelle

Dans la décomposition virtuelle, on découpe un robot complexe en n_o *objets* et n_c *chaînes ouvertes*. Un exemple de découpage pour un robot de grandes dimensions généralisées est donné aux figures 2.1 et 2.2

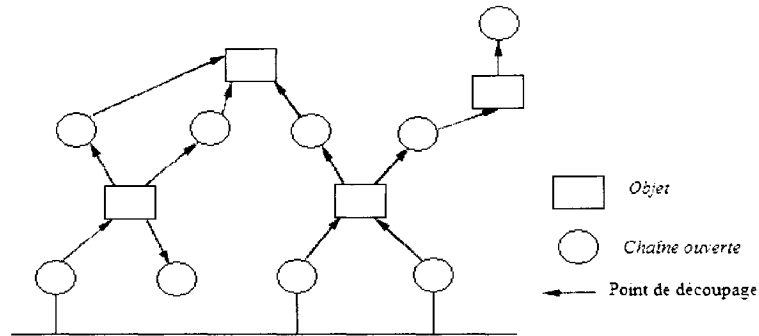


Figure 2.2 Schéma illustrant la décomposition des robots de la figure 2.1

2.2.1 Objets

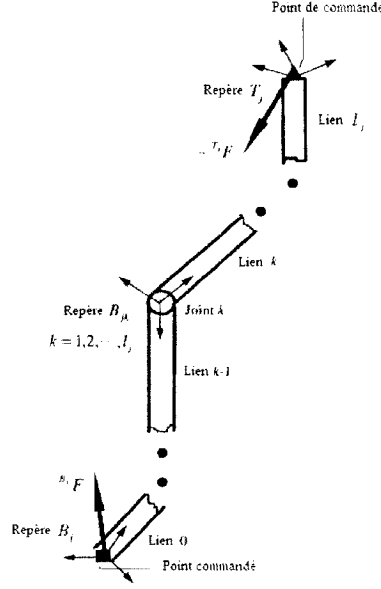
Un *objet* est un corps rigide. Tous les corps rigides en contact avec l'environnement doivent être considérés comme des *objets*.

2.2.2 Chaînes ouvertes

Une *chaîne ouverte*, représentée à la figure 2.3, est une chaîne unique à base mobile en mouvement libre, constituée d'une série de segments rigides connectés un à un par une articulation à un degré de liberté. Ainsi une chaîne ouverte à l articulations est composée de $l + 1$ segments.

Remarque : D'après la définition de *chaîne ouverte*, un segment rigide avec trois articulations ou plus doit être considéré comme un *objet*.

Sur notre exemple à la figure 2.1, on constate qu'une patte accrochée au corps 1 ou 2 correspond à une *chaîne ouverte*. En effet il s'agit d'une chaîne composée de deux segments rigides liés entre eux par une articulation en rotation, d'axe de rotation perpendiculaire à la feuille. Nous avons donc huit *chaînes ouvertes* de ce type. Le

Figure 2.3 Exemple de *chaîne ouverte*

baton tenu par le corps 2 est en translation par rapport à sa base, il s'agit donc d'une neuvième *chaîne ouverte*. Les trois corps rigides restant seront considérés comme des *objets*. On retrouve bien cette décomposition sur la figure 2.2.

2.2.3 Lien entre objets et chaînes ouvertes

Les points où sont effectués le découpage du robot en *objets* et *chaînes ouvertes* sont les *points de découpage*. Chaque point de découpage doit être situé sur un corps rigide afin que toutes les composantes de la force et du moment soient complètement transférées à travers ce point de découpage. En effet à travers un point de découpage passe la commande du sous-système. Il est donc à la fois un point de commande et un point commandé, en fonction du sous-système sur lequel on se place.

Sur l'exemple de la figure 2.3, la flèche représente un point de découpage. Elle pointe vers un point commandé ou point d'entrée et part d'un point de commande ou point de sortie.

Ainsi à un bout d'une *chaîne ouverte* se trouve un point commandé et à l'autre se trouve un point de commande.

Un objet est composé de plusieurs points commandés et points de commande.

Ainsi un point de découpage peut être à la fois un point qui effectue la commande pour un *objet* et un point où est effectué la commande pour une *chaîne ouverte* et inversement.

Il n'existe pas de points de commande pour un *objet* ou une *chaîne ouverte* à une extrémité de la partie décomposée. La force/moment, qui s'exerce à un point de découpage donné, est défini comme la force/moment exercé du point de commande d'un *objet* (resp. *chaîne ouverte*) vers le point commandé de la *chaîne ouverte* (resp. *objet*) associée.

A travers un point de découpage passe le *flux virtuel de puissance*. Cette notion sera expliquée ultérieurement.

2.3 Préliminaire pour le contrôle

2.3.1 Force et vitesse

Soient α et β deux repères attachés à un corps rigide.

Soit ${}^{\alpha}F$ un vecteur de force/moment généralisé donné, mesuré dans le repère α . Il peut être exprimé dans le repère β à l'aide de la matrice de transformation ${}^{\beta}U_{\alpha}$ selon l'expression suivante :

$${}^{\beta}F_{\alpha} = {}^{\beta}U_{\alpha} \cdot {}^{\alpha}F$$

De même pour une vitesse linéaire/angularisée généralisée donnée on peut écrire la relation suivante :

$${}^{\alpha}X = {}^{\beta}U_{\alpha}^T.{}^{\beta}X$$

On obtient alors :

$${}^{\alpha}X^T.{}^{\alpha}F = {}^{\beta}X^T.{}^{\beta}F_{\alpha}$$

Ceci signifie que le flux de puissance à l'intérieur d'un corps rigide est invariant. Ceci est la base pour introduire le concept de *flux virtuel de puissance* qui sera fréquemment utilisé pour l'analyse de stabilité.

2.3.2 Variables intermédiaires

Afin de prendre en compte le *flux virtuel de puissance* et d'effectuer le contrôle, on introduit des variables intermédiaires de la forme :

- pour toutes les articulations k d'une chaîne ouverte j :

$$\dot{q}_{jkr} = \dot{q}_{jkd} + \lambda(q_{jkd} - q_{jk}) \quad \lambda > 0 \quad (2.1)$$

- pour tous les objets du robots :

$$\chi_{ir} = \chi_{id} + \lambda\epsilon_i \quad (2.2)$$

Avec ϵ_i , vecteur d'erreur, défini tel que pour une matrice K_i définie positive donnée, il existe deux réels $\gamma_i > 0$ et $k_i > 0$ tels que :

$$k_i \epsilon_i^T \epsilon_i \geq \int_0^t \left[(\chi_{id} - \chi_i)^T K_i \epsilon_i \right] dt \geq -\gamma_i \quad (2.3)$$

Ainsi de façon similaire à la méthode de Slotine-Li, dans les équations 2.1 et 2.2 on introduit la vitesse requise comme étant la vitesse désirée ou de référence plus un terme correctif relié aux erreurs de position ou orientation.

Cependant contrairement à Slotine-Li, la stabilité en position n'est pas considérée pour construire la vitesse requise. Dans ce cas l'équation 2.3 tient lieu de condition qui élargit le champs de conception.

2.4 Flux virtuel de puissance

2.4.1 Définition

Le *flux virtuel de puissance* (FVP) à un point de découpage, où est fixé le repère α , est défini comme le produit vectoriel entre l'erreur de vitesse généralisée et l'erreur de force généralisée en ce point :

$$W_\alpha = ({}^\alpha X_r - {}^\alpha X)^T \cdot ({}^\alpha F_r - {}^\alpha F) \quad (2.4)$$

où ${}^\alpha X$ et ${}^\alpha F$ sont la vitesse actuelle et la force actuelle respectivement et ${}^\alpha X_r$ et ${}^\alpha F_r$ sont la vitesse requise et la force requise respectivement.

D'après la relation qui existe entre la force généralisée et la vitesse généralisée, le

FVP est invariant par rapport au repère de mesure.

2.4.2 Spécifications du flux virtuel de puissance

En plus de donner une vue d'ensemble sur le système complet (comme le fait le concept de *travail virtuel* donné par Walker dans (Walker, 1990)), le FVP donne une vue détaillée du couplage dynamique entre les sous-systèmes. Pour un système robotique de grande dimension, le couplage dynamique entre les sous-systèmes peut être entièrement représenté par le FVP au point de découpage.

En effet, on suppose que la force de friction, prise comme une part de la dynamique de l'*objet*, est séparée de la force de contrainte de cet *objet*. Ceci assure que le *flux virtuel de puissance* entre un *objet* et son environnement est nul. On peut alors affirmer que si le FVP entre deux sous-systèmes est nul alors ils peuvent se voir l'un l'autre comme un environnement. Cette caractéristique assure la décomposition complète.

De même le couplage dynamique entre une *chaîne ouverte* et le reste des sous-systèmes du système robotique de grande dimension est complètement représenté par le FVP aux deux points de découpage de la *chaîne ouverte*. Le signe du FVP est positif à un point commandé et négatif à un point de commande. Il a la même valeur absolue, pour un point de découpage donné, au point de commande d'un *objet* qu'au point commandé d'une *chaîne ouverte* et inversement.

Le *flux virtuel de puissance* sera également utilisé pour démontrer la stabilité du système.

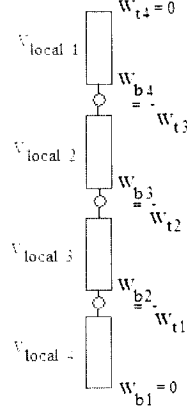


Figure 2.4 Schéma des *flux virtuels de puissance* pour un robot décomposé

2.5 Stabilité

On choisit pour fonction de Lyapunov du système la somme de toutes les fonctions non négatives associées aux sous-systèmes :

$$E_L = \sum_{i=0}^{n_o} V_{O_i} + \sum_{j=0}^{n_c} V_j \geq 0$$

Chaque fonction locale est telle que $V_{locale} \geq 0$ et $\dot{V}_{locale} \leq -Ks^2 + W_t - W_b$ où $K > 0$ et W_t et W_b sont les *flux virtuels de puissance* à chaque extrémité de l'*objet* ou la *chaîne ouverte*, comme représenté sur la figure 2.4.

Or un FVP représente le couplage dynamique entre deux sous-systèmes et à un point de découpage deux FVP sont égaux et de signe opposé. Donc ils s'annulent deux à deux d'un bout à l'autre du robot lorsqu'on les additionne. Il reste alors les FVP qui représentent le couplage dynamique entre le robot et son environnement :

$$\dot{E}_L = - \sum_{i=0}^{n_o} K_i s_i^2 - \sum_{j=0}^{n_c} K_j s_j^2$$

$$+ \sum_{i=0}^{n_o} W_{Ti} + \sum_{j=0}^{n_c} W_{Tj} + \sum_{i=0}^{n_o} W_{Bi} + \sum_{j=0}^{n_c} W_{Bj}$$

Or comme les forces de friction ont été modélisées dans la dynamique de l'*objet*, le FVP, représentant le couplage dynamique entre le robot et son environnement, est nul. Les gains K_i et K_j étant tous positifs, on a donc $\dot{E}_L \leq 0$. La stabilité du système est bien assurée.

2.6 Conclusion

Dans l'article (Zhu et al., 1997) Zhu et al. appliquent la décomposition virtuelle à divers manipulateurs complexes, effectuant des tâches précises. Ils montrent qu'elle permet une approche beaucoup plus aisée de la commande. En effet à l'aide de ce principe, le problème de commande du mouvement du système originel est transformé en un problème lié à chaque sous-système, à savoir les *objets* et les *chaînes ouvertes*. Alors que le problème de commande de forces de contraintes et internes n'est lié qu'à chaque *objet* seulement.

L'avantage d'un tel modèle est que le contrôleur a une structure programmable hiérarchique. Chaque sous-système possède un module de commande indépendant des autres.

Enfin la stabilité et la robustesse du système ainsi décomposé sont garanties au sens de Lyapunov. En effet la stabilité asymptotique au sens de Lyapunov est assurée par le fait que tous les *flux virtuels de puissance* entre le robot et l'environnement sont nuls et que les *flux virtuels de puissance* entre deux sous-systèmes physiquement connectés sont égaux et de signes opposés. Ainsi le couplage dynamique entre deux sous-systèmes physiquement connectés est complètement représenté par le FVP.

CHAPITRE 3

ENONCÉ ET SOLUTION STABLE DU PROBLÈME

3.1 Situation du problème

3.1.1 Système

On considère le système bras flexible à 6ddl (donc il n'est attaché à aucune base) en libre mouvement plan, comme montré à la figure 3.1. Il s'agit d'un tube de section circulaire, de longueur L , de masse linéique ρ et de rigidité EI . Ces trois derniers paramètres sont des constantes. Elles changent de valeurs en fonction de l'objet considéré.

Nous ne disposons que de deux capteurs placés au début et à la fin de la barre. Les variables auxquelles nous avons accès sont donc la vitesse et la position de déformation aux deux extrémités.

Nous choisissons comme variables de commande la force/moment qui agissent de l'environnement sur le début de la barre.

Comme montré sur le schéma à la figure 3.2 les entrées du système sont la force et le moment en début de bras. La sortie du système est la déformation et la vitesse de déformation de la poutre.

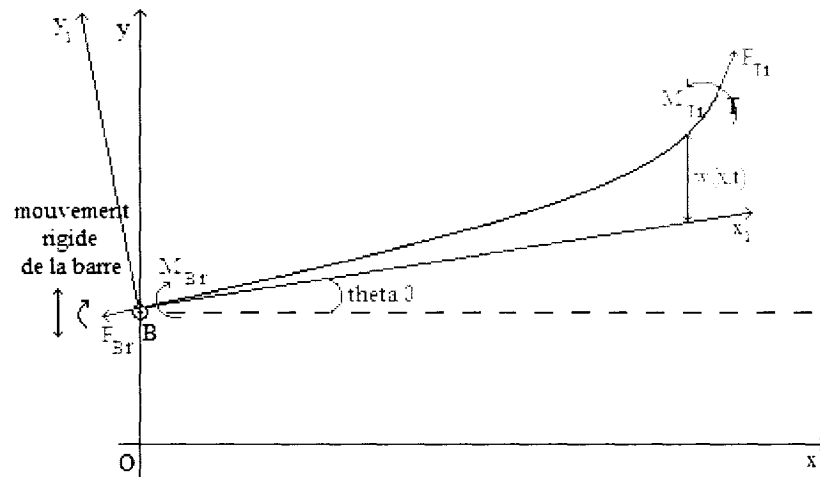


Figure 3.1 Schéma du mouvement de la barre

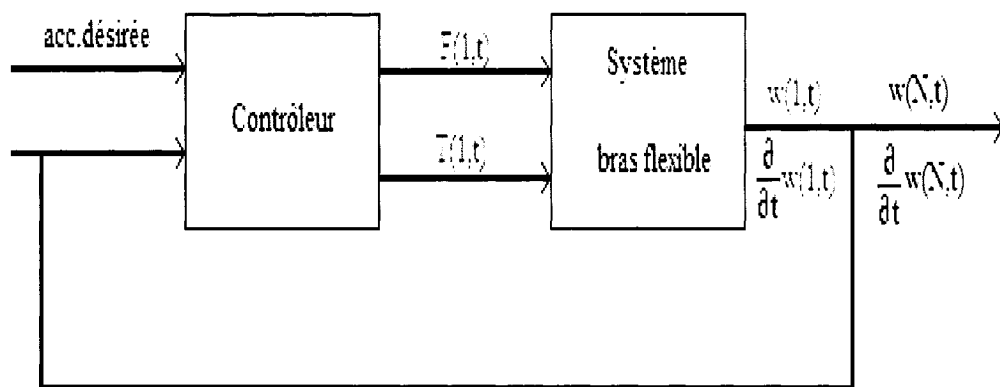


Figure 3.2 Schéma du système complet

3.1.1.1 Equation de la dynamique

L'équation générale de la dynamique pour une poutre est donnée par L.Meirovitch (Meirovitch, 1997).

Équation générale

Soit une poutre de masse m et de longueur L et de rigidité de flexion EI , soumise à une force axiale $P(x)$ et une distribution de charges transversales $f(x, t)$.

Soit $\omega(x, t)$ le déplacement transverse de la poutre, tel qu'illustré à la figure 3.1.

L'équation de la dynamique de cette poutre est donnée par :

$$m \frac{\partial^2 \omega}{\partial t^2} + \frac{\partial^2}{\partial x^2} \left(EI \frac{\partial^2 \omega}{\partial x^2} \right) - \frac{\partial}{\partial x} \left(P \frac{\partial \omega}{\partial x^2} \right) - f = 0 \quad \text{pour } 0 < x < L$$

avec des conditions aux limites spécifiques à chaque cas.

Notre équation de la dynamique

Comme dit précédemment, on suppose la barre de longueur L , de masse volumique ρ , de rigidité de flexion EI constantes.

On suppose que le mouvement s'effectue dans un plan et que la poutre n'est soumise à aucune force transverse distribuée ($f(x, t) = 0$), ni aucune force axiale ($P(x) = 0$), ni force de gravité. L'équation de la dynamique s'écrit alors :

$$\rho \frac{\partial^2 \omega(x, t)}{\partial t^2} + EI \frac{\partial^4 \omega(x, t)}{\partial x^4} = 0 \quad 0 < x < L \quad (3.1)$$

3.1.1.2 Conditions aux limites

Toujours d'après le livre de Meirovitch (Meirovitch, 1997), dans le cas d'une poutre prismatique, c'est-à-dire avec $EI = \text{constante}$, les relations suivantes sont toujours vraies :

$$\begin{aligned} -EI \frac{\partial^2 \omega(x, t)}{\partial x^2} &= \tau_{total}(x, t) & 0 < x < L \\ EI \frac{\partial^3 \omega(x, t)}{\partial x^3} &= F_{totale}(x, t) & 0 < x < L \end{aligned}$$

On suppose de plus que la barre est en libre mouvement et que la force de gravité est négligée. Les seules forces et moments qui s'appliquent aux extrémités de la barre sont donc ceux que la barre applique sur son environnement. A l'extrémité B, sur la figure 3.1 nous avons F_B et τ_B et à l'extrémité T, nous avons F_T et τ_T . Nous en déduisons alors les conditions aux limites qui s'appliquent à notre problème :

1.

$$F_{Br} - EI \frac{\partial^3 \omega}{\partial x^3} \Big|_{x=0} = 0 \quad \text{et} \quad F_{Tr} - EI \frac{\partial^3 \omega}{\partial x^3} \Big|_{x=L} = 0 \quad (3.2)$$

2.

$$\tau_{Br} + EI \frac{\partial^2 \omega}{\partial x^2} \Big|_{x=0} = 0 \quad \text{et} \quad \tau_{Tr} + EI \frac{\partial^2 \omega}{\partial x^2} \Big|_{x=L} = 0 \quad (3.3)$$

3.1.2 Contrôleur

Pour une trajectoire désirée donnée de la fin de la poutre, l'objectif est de trouver une loi de commande qui, à partir des mesures des variables accessibles, gère les actions de la base telles que le bout du robot suive la trajectoire désirée. Le

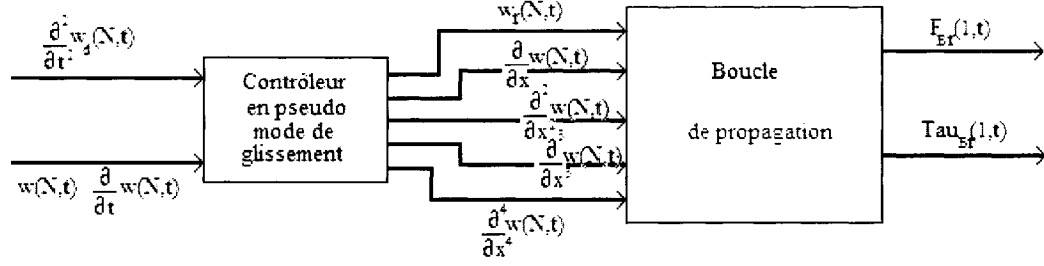


Figure 3.3 Détails du contrôleur

schéma de la figure 3.2 montre que le contrôleur prend en entrée l'accélération de déformation désirée, la déformation actuelle ainsi que celle en vitesse, en début de poutre. En sortie on obtient la force et le moment, à injecter en début de poutre, calculés pour suivre l'accélération désirée avec le moins d'oscillations possible.

3.2 Contrôleur choisi

Notre contrôleur est composé de deux parties. Comme le montre la figure 3.3, la première partie correspond à un contrôleur en pseudo mode de glissement. Cette première partie permet de respecter la théorie de la décomposition virtuelle, comme énoncé dans l'article (Zhu et al., 1997). A partir des données du système, le contrôleur calcule la force et le moment requis en fin de poutre. Cependant la commande doit être injectée en début de poutre. La deuxième partie du contrôleur permettra donc de déplacer les données de commande de la fin vers le début de la poutre.

3.2.1 Première partie du contrôleur

3.2.1.1 Contrôleur par mode de glissement

Le contrôleur par mode de glissement est celui qui semble le plus approprié à la résolution de notre problème. Un tel contrôleur se comporte correctement même si les paramètres du manipulateur sont entachés d'incertitude. En effet, bien que nous connaissions parfaitement les caractéristiques des barres que nous utilisons, le modèle utilisé pour recréer leur dynamique n'est pas très précis. Il ne comporte que deux modes de déformation, ce qui donne une reproduction approximative du mouvement de la barre.

Un contrôleur en mode de glissement comporte deux sous-contrôleurs :

1. un contrôleur cinétique qui fournit une vitesse de référence $\dot{q}^*$ à partir de la vitesse désirée et de l'erreur de position. Cette vitesse de référence est telle que si la vitesse réelle égale la vitesse de référence alors la position réelle tend vers la position désirée.
2. un contrôleur de vitesse qui fournit les commandes : la force F et le moment τ , à partir de l'équation de la dynamique de la barre. Ces commandes entraînent la convergence de la vitesse réelle vers la vitesse de référence.

3.2.2 Notre contrôleur

La commande est un asservissement de position de déformation de la barre. Elle est effectuée par rétroaction de position de déformation et de vitesse de déformation de la fin de la barre.

Le contrôleur que nous avons retenu pour résoudre notre problème est quasiment similaire à un contrôleur par mode de glissement. En effet notre solution est basée sur la théorie de l'article (Zhu et al., 1997) que nous avons détaillée dans le chapitre 2 sur la décomposition virtuelle.

La variable d'état considérée est la déformation ω à la fin de la barre ($x = L$). Afin de commander cet état par la force et le moment en début de poutre, on fait intervenir une variable intermédiaire : la *déformation de référence* ω_r .

Comme énoncé dans le chapitre 2, section *variables intermédiaires*, cette variable est construite de façon similaire à celle de la méthode de Slotine-Li. Et sa forme est telle qu'elle respecte le concept de *flux virtuel de puissance* dans la décomposition virtuelle. On choisit de la faire intervenir sous forme de dérivée temporelle première et deuxième. Ces dérivées sont calculées à l'aide des équations suivantes :

$$\frac{\partial \omega_r(x, t)}{\partial t} = \frac{\partial \omega_d(x, t)}{\partial t} + \lambda_1 [\omega_d(x, t) - \omega(x, t)] \quad (3.4)$$

$$\frac{\partial^2 \omega_r(x, t)}{\partial t^2} = \frac{\partial^2 \omega_d(x, t)}{\partial t^2} + \lambda_1 \left[\frac{\partial \omega_d(x, t)}{\partial t} - \frac{\partial \omega(x, t)}{\partial t} \right] \quad (3.5)$$

Si la vitesse de déformation de référence suit la vitesse de déformation désirée, alors la position actuelle tend vers la position désirée.

Elle est ensuite injectée dans un contrôleur Proportionnel Intégrateur classique qui nous permettra de calculer la force et le moment de référence en fin de poutre. Ces commandes ont pour but de faire converger la vitesse de déformation de référence vers la vitesse de déformation désirée.

Les expressions de la force et du moment seront données ultérieurement, lors de la preuve de la stabilité du contrôleur.

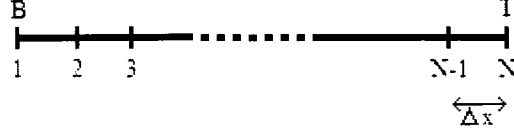


Figure 3.4 Discrétisation de la barre

3.2.3 Deuxième partie du contrôleur : propagation de la commande

3.2.3.1 Discrétisation de la poutre

Le problème considéré est supposé à temps discret de période d'échantillonnage ΔT .

Afin de faciliter la mise en oeuvre, on discrétise également spatialement (voir figure 3.4). On découpe le segment en $N = 100$ portions de longueur ΔX . A chaque point, entre deux portions, on calculera la déformation, sa vitesse, son accélération et ses dérivées spatiales première, deuxième, troisième et quatrième. Ceci permettra de propager l'information de la fin vers le début de la barre. On aura ainsi accès à la force et au moment nécessaires pour produire les commandes désirées.

3.2.3.2 Obtention des valeurs de commande

A l'entrée de la boucle de propagation nous connaissons : $\frac{\partial^4 \omega(N,t)}{\partial x^4}$, $\frac{\partial^3 \omega(N,t)}{\partial x^3}$, $\frac{\partial^2 \omega(N,t)}{\partial x^2}$, $\frac{\partial \omega(N,t)}{\partial x}$, $\omega_r(N,t)$. Il faut alors propager ces données de la fin de la barre vers le début.

Les équations utilisées pour effectuer les calculs sont issues du développement en série de Taylor de la dérivée, tronqué au premier ordre, et de l'équation de la

dynamique. Le développement en série de Taylor utilisé est le suivant :

$$\omega(x, t - \Delta T) = \omega(x, t) - \Delta T \frac{\partial \omega(x, t)}{\partial t} \Rightarrow \frac{\partial}{\partial t} \omega(x, t) = \frac{\omega(x, t) - \omega(x, t - \Delta T)}{\Delta T} \quad (3.6)$$

$$\text{ou } \omega(x - \Delta X, t) = \omega(x, t) - \Delta X \frac{\partial \omega(x, t)}{\partial x} \quad (3.7)$$

Pour pouvoir propager les données, il nous faut calculer à chaque point de la barre discrétisée, les dérivées spatiales et temporelles de la déformation.

Remarque : A la sortie du contrôleur en pseudo mode de glissement, nous obtenons les expressions de la force et du moment en fin de poutre. Or il est plus simple d'utiliser directement les expressions des dérivées spatiales troisième et deuxième respectivement. D'après les équations des conditions limites 3.2 et 3.3, nous écrivons :

$$\frac{\partial^3 \omega_r(L, t)}{\partial x^3} = \frac{F_{Tr}}{EI} \quad \text{et} \quad - \frac{\partial^2 \omega_r(L, t)}{\partial x^2} = \frac{\tau_{Tr}}{EI}$$

De même nous connaissons l'expression de l'accélération de déformation désirée en fin de poutre, mais il est plus simple d'utiliser directement la dérivée spatiale quatrième en fin de poutre. Nous trouvons son expression à l'aide de l'équation de la dynamique 3.1 :

$$\frac{\partial^4 \omega_d(L, t)}{\partial x^4} = - \frac{\rho}{EI} \frac{\partial^2 \omega_d(L, t)}{\partial t^2}$$

De plus nous choisissons la première dérivée spatiale de la déformation en bout de poutre nul : $\frac{\partial \omega_r(L, t)}{\partial x} = 0$.

Sachant que le point en $x = L$ correspond au point N, à l'entrée de la boucle de propagation, nous connaissons bien $\frac{\partial^4 \omega(N, t)}{\partial x^4}$, $\frac{\partial^3 \omega(N, t)}{\partial x^3}$, $\frac{\partial^2 \omega(N, t)}{\partial x^2}$, $\frac{\partial \omega(N, t)}{\partial x}$, $\omega_r(N, t)$.

Nous entrons ensuite dans la boucle de propagation. La figure 3.5 nous montre le

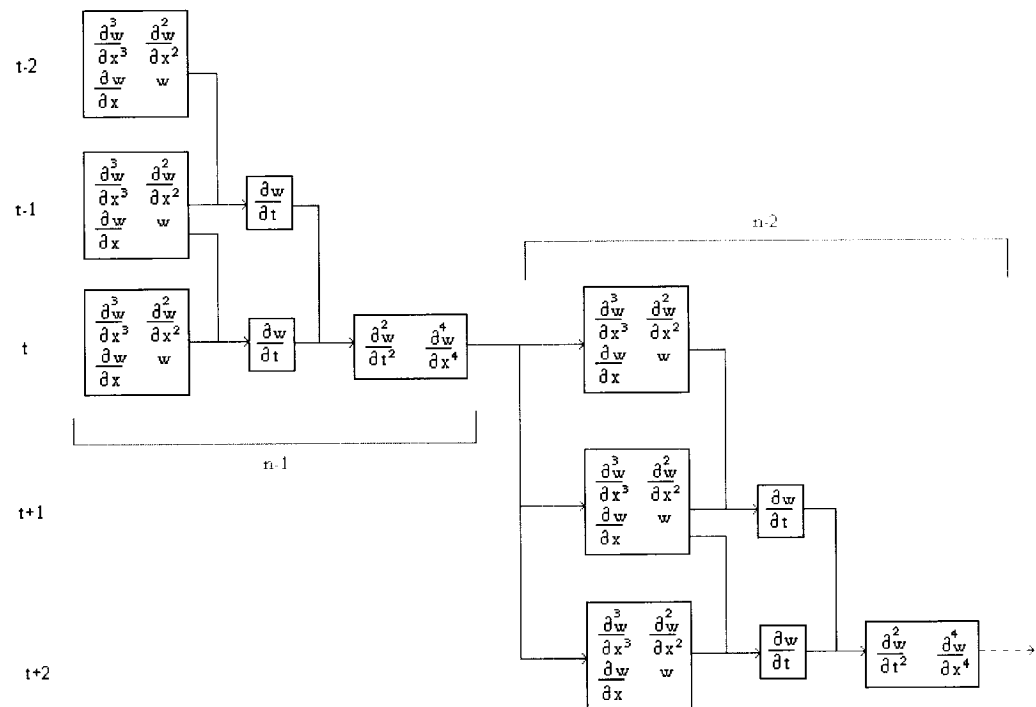


Figure 3.5 Propagation de l'information de commande de la fin vers de le début de la barre discrétisée

déroulement de la propagation. Tout d'abord, nous calculons les trois premières dérivées spatiales et la position de déformation au point $n - 1$ et au temps $t - 2$ à l'aide du développement en série de Taylor 3.7 et des différentes dérivées spatiales et temporelles que nous connaissons au point n :

1. Calcul de la dérivée 3^{ème} au rang précédent

$$\frac{\partial^3}{\partial x^3}\omega(n-1, t) = \frac{\partial^3}{\partial x^3}\omega(n, t) - \Delta X \frac{\partial^4}{\partial x^4}\omega(n, t)$$

2. Calcul de la dérivée 2^{ème} au rang précédent

$$\frac{\partial^2}{\partial x^2}\omega(n-1, t) = \frac{\partial^2}{\partial x^2}\omega(n, t) - \Delta X \frac{\partial^3}{\partial x^3}\omega(n, t)$$

3. Calcul de la dérivée 1^{ère} au rang précédent

$$\frac{\partial}{\partial x}\omega(n-1, t) = \frac{\partial}{\partial x}\omega(n, t) - \Delta X \frac{\partial^2}{\partial x^2}\omega(n, t)$$

4. Calcul de la position au rang précédent

$$\omega(n-1, t) = \omega(n, t) - \Delta X \frac{\partial}{\partial x}\omega(n, t)$$

Puis nous effectuons de nouveau les étapes 1 à 4 au temps $t - 1$. Nous connaissons alors la position de déformation sur deux pas temporels. Nous pouvons donc calculer la vitesse de déformation du point $n - 1$ au temps $t - 1$, à l'aide du développement en série de Taylor 3.6 :

5. Calcul de la vitesse de déformation du point considéré

$$\frac{\partial}{\partial t}\omega(n-1, t) = \frac{\omega(n-1, t) - \omega(n-1, t - \Delta T)}{\Delta T}$$

Au temps t , nous effectuons les étapes 1 à 5. Nous connaissons alors la vitesse de déformation sur deux pas temporels. Nous calculons donc l'accélération de déformation du point $n - 1$ au temps t , à l'aide du développement en série de Taylor 3.6 :

6. Calcul de l'accélération de déformation du point considéré

$$\frac{\partial^2}{\partial t^2}\omega(n-1, t) = \frac{\frac{\partial\omega(n-1, t)}{\partial t} - \frac{\partial\omega(n-1, t-\Delta T)}{\partial t}}{\Delta T}$$

Maintenant que nous connaissons l'accélération de déformation du point $n - 1$ au temps t , nous pouvons calculer la quatrième dérivée spatiale de la déformation du point $n - 1$ au temps t , à l'aide de l'équation de la dynamique 3.1 :

7. Calcul de la dérivée 4^{ème} de déformation au point considéré

$$\frac{\partial^4}{\partial x^4}\omega(n-1, t) = -\frac{\rho}{EI}\frac{\partial^2}{\partial t^2}\omega(n-1, t)$$

Maintenant que nous avons calculé toutes les dérivées temporelles et spatiales de la déformation au point $n - 1$ et au temps t , nous pouvons également calculer les trois premières dérivées spatiales et la position de déformation du point $n - 2$ au temps t . On recommence alors les étapes 1 à 4 pour le point $n - 2$.

Puis au temps $t + 1$, on effectue les étapes 1 à 7 pour le point $n - 1$ et les étapes 1 à 5 pour le point $n - 2$. Au temps $t + 2$, on effectue les étapes 1 à 7 pour les points $n - 1$ et $n - 2$ et les étapes 1 à 4 pour le point $n - 3$.

Et ainsi de suite, jusqu'à avoir propagé les informations de déformation d'un bout à l'autre de la poutre.

Propager l'information de la fin vers le début de la poutre nécessite une période d'initialisation de $2 \times N + 1 = 201$ pas de calcul (si $N = 100$). Une fois la période

d'initialisation terminée, l'ensemble de la boucle de propagation est effectuée pour tous les points de la barre à chaque temps.

Arrivé au début du bras, on en sort les commandes, à savoir la force F_B et le moment τ_B . Leurs expressions seront données ultérieurement lors de l'analyse de la stabilité du système.

Remarque : Pendant la période d'initialisation, on choisit $F_B = \tau_B = 0$. La barre n'est pas commandée durant cette période.

Remarque : La procédure décrite au-dessus a été écrite sous différents langages (Maple, Simulink, Matlab) pour finir par sa version finale sous forme de fonction Matlab injectée dans un fichier Simulink à l'aide d'un bloc "function.m".

3.3 Stabilité du contrôleur

3.3.1 Equation de la dynamique de référence

Afin de pouvoir établir la stabilité du contrôleur que vous venons d'étudier, nous avons besoin de réécrire l'équation de la dynamique 3.1 donnée au début de ce chapitre pour la déformation désirée :

$$\rho \frac{\partial^2 \omega_d(x, t)}{\partial t^2} + EI \frac{\partial^4 \omega_d(x, t)}{\partial x^4} = 0 \quad 0 < x < L$$

Dans cette équation nous réécrivons l'accélération désirée de déformation en fonction de l'expression de l'accélération de la variable intermédiaire, donnée à l'équation 3.5:

$$\rho \left[\frac{\partial^2 \omega_r(x, t)}{\partial t^2} - \lambda \left[\frac{\partial \omega_d(x, t)}{\partial t} - \frac{\partial \omega(x, t)}{\partial t} \right] \right] + EI \frac{\partial^4 \omega_d(x, t)}{\partial x^4} = 0$$

Après remaniement, on obtient l'équation de la dynamique de référence :

$$\rho \frac{\partial^2 \omega_r(x, t)}{\partial t^2} + K_V \left(\frac{\partial \omega_r(x, t)}{\partial t} - \frac{\partial \omega(x, t)}{\partial t} \right) + EI \frac{\partial^4 \omega_d(x, t)}{\partial x^4} = 0 \quad (3.8)$$

3.3.2 Fonction de Lyapunov

La stabilité du contrôleur est ensuite prouvée par une méthode de l'énergie de Lyapunov. La fonction de Lyapunov choisie E_L est la somme d'une pseudo énergie cinétique de la barre :

$$T(t) = \frac{1}{2} \int_0^L \rho \left[\frac{\partial \omega_r(x, t)}{\partial t} - \frac{\partial \omega(x, t)}{\partial t} \right]^2 dx$$

et d'une pseudo énergie potentielle :

$$V(t) = \frac{1}{2} \int_0^L EI \left[\frac{\partial^2 \omega_d(x, t)}{\partial x^2} - \frac{\partial^2 \omega(x, t)}{\partial x^2} \right]^2 dx$$

D'où $E_L = T + V$ est une fonction positive.

3.3.3 Dérivée de la fonction de Lyapunov

Il faut maintenant prouver la négativité de sa dérivée. Pour des raisons de lisibilité, nous avons enlevé l'argument de ω dans les calculs. Nous commençons par calculer la dérivée de la pseudo énergie cinétique :

$$\dot{T} = \int_0^L \rho \left[\frac{\partial \omega_r}{\partial t} - \frac{\partial \omega}{\partial t} \right] \left[\frac{\partial^2 \omega_r}{\partial t^2} - \frac{\partial^2 \omega}{\partial t^2} \right] dx$$

Puis nous remplaçons $\rho \frac{\partial^2 \omega_r}{\partial t^2}$ à l'aide de l'équation de la dynamique de référence 3.8. Nous remplaçons aussi $\rho \frac{\partial^2 \omega}{\partial t^2}$ à l'aide de l'équation de la dynamique 3.1. On obtient

alors :

$$\dot{T} = -K_v \int_0^L \left[\frac{\partial \omega_r}{\partial t} - \frac{\partial \omega}{\partial t} \right]^2 dx - EI \int_0^L \left[\frac{\partial \omega_r}{\partial t} - \frac{\partial \omega}{\partial t} \right] \left[\frac{\partial^4 \omega_d}{\partial x^4} - \frac{\partial^4 \omega}{\partial x^4} \right] dx$$

Puis on effectue une double intégration par partie sur le deuxième terme :

$$\begin{aligned} \dot{T} = & -K_v \int_0^L \left[\frac{\partial \omega_r}{\partial t} - \frac{\partial \omega}{\partial t} \right]^2 dx - EI \left[\left(\frac{\partial \omega_r}{\partial t} - \frac{\partial \omega}{\partial t} \right) \left(\frac{\partial^3 \omega_d}{\partial x^3} - \frac{\partial^3 \omega}{\partial x^3} \right) \right]_0^L \\ & + EI \left[\left(\frac{\partial}{\partial x} \left(\frac{\partial \omega_r}{\partial t} \right) - \frac{\partial}{\partial x} \left(\frac{\partial \omega}{\partial t} \right) \right) \left(\frac{\partial^2 \omega_d}{\partial x^2} - \frac{\partial^2 \omega}{\partial x^2} \right) \right]_0^L \\ & - EI \int_0^L \left[\frac{\partial^2}{\partial x^2} \left(\frac{\partial \omega_r}{\partial t} \right) - \frac{\partial^2}{\partial x^2} \left(\frac{\partial \omega}{\partial t} \right) \right] \left[\frac{\partial^2 \omega_d}{\partial x^2} - \frac{\partial^2 \omega}{\partial x^2} \right] dx \end{aligned}$$

Dans la dernière intégrale, on remplace $\frac{\partial \omega_r}{\partial t}$ par son expression à l'équation 3.4 :

$$\begin{aligned} \dot{T} = & -K_v \int_0^L \left[\frac{\partial \omega_r}{\partial t} - \frac{\partial \omega}{\partial t} \right]^2 dx - EI \left[\left(\frac{\partial \omega_r}{\partial t} - \frac{\partial \omega}{\partial t} \right) \left(\frac{\partial^3 \omega_d}{\partial x^3} - \frac{\partial^3 \omega}{\partial x^3} \right) \right]_0^L \\ & + EI \left[\left(\frac{\partial}{\partial x} \left(\frac{\partial \omega_r}{\partial t} \right) - \frac{\partial}{\partial x} \left(\frac{\partial \omega}{\partial t} \right) \right) \left(\frac{\partial^2 \omega_d}{\partial x^2} - \frac{\partial^2 \omega}{\partial x^2} \right) \right]_0^L \\ & - EI \int_0^L \left[\frac{\partial^2}{\partial x^2} \left(\frac{\partial \omega_d}{\partial t} \right) - \frac{\partial^2}{\partial x^2} \left(\frac{\partial \omega}{\partial t} \right) \right] \left[\frac{\partial^2 \omega_d}{\partial x^2} - \frac{\partial^2 \omega}{\partial x^2} \right] dx \\ & - \lambda EI \int_0^L \left[\frac{\partial^2 \omega_d}{\partial x^2} - \frac{\partial^2 \omega}{\partial x^2} \right]^2 dx \end{aligned}$$

Nous calculons également la dérivée de la pseudo énergie potentielle :

$$\dot{V} = EI \int_0^L \left[\frac{\partial}{\partial t} \left(\frac{\partial^2 \omega_d}{\partial x^2} \right) - \frac{\partial}{\partial t} \left(\frac{\partial^2 \omega}{\partial x^2} \right) \right] \left[\frac{\partial^2 \omega_d}{\partial x^2} - \frac{\partial^2 \omega}{\partial x^2} \right] dx$$

Or comme les variables x et t sont indépendantes, on peut intervertir leurs dérivées partielles. Lorsqu'on additionne la dérivée de la pseudo énergie cinétique avec la dérivée de la pseudo énergie potentielle, $\dot{V}$ vient annuler l'avant dernier terme de $\dot{T}$. De plus à l'aide des conditions aux limites 3.2 et 3.3, nous faisons apparaître la force et le moment à chaque extrémité dans le calcul.

On obtient alors la dérivée de la fonction de Lyapunov $\dot{E}_L = \dot{T} + \dot{V}$:

$$\begin{aligned}\dot{E}_L = & -K_V \int_0^l \left[\frac{\partial \omega_r(x, t)}{\partial t} - \frac{\partial \omega(x, t)}{\partial t} \right]^2 dx \\ & - \lambda \int_0^L EI \left[\frac{\partial^2 \omega_d(x, t)}{\partial x^2} - \frac{\partial^2 \omega(x, t)}{\partial x^2} \right]^2 dx \\ & + P_B - P_T\end{aligned}$$

avec :

$$P_B = \left[\frac{\partial \omega_r(x, t)}{\partial t} \Big|_{x=0} - \frac{\partial \omega(x, t)}{\partial t} \Big|_{x=0} \right] (F_{Br} - F_B) \quad (3.9)$$

$$+ \left[\frac{\partial^2 \omega_r(x, t)}{\partial x \partial t} \Big|_{x=0} - \frac{\partial^2 \omega(x, t)}{\partial x \partial t} \Big|_{x=0} \right] (\tau_{Br} - \tau_B) \quad (3.10)$$

et

$$P_T = \left[\frac{\partial \omega_r(x, t)}{\partial t} \Big|_{x=L} - \frac{\partial \omega(x, t)}{\partial t} \Big|_{x=L} \right] (F_{Tr} - F_T) \quad (3.11)$$

$$+ \left[\frac{\partial^2 \omega_r(x, t)}{\partial x \partial t} \Big|_{x=L} - \frac{\partial^2 \omega(x, t)}{\partial x \partial t} \Big|_{x=L} \right] (\tau_{Tr} - \tau_T) \quad (3.12)$$

On constate que les termes P_B et P_T sont liés au *flux virtuel de puissance*. En effet leurs expressions rappellent celles du *flux virtuel de puissance* comme défini dans le chapitre 2 à l'équation 2.4.

Pour garantir la stabilité du contrôleur, il faut que $\dot{E}_L \leq 0$. Il est donc judicieux de commander également la force et le moment aux deux extrémités. Nous assurerons ainsi la négativité du terme P_B et la positivité du terme P_T .

D'après l'expression de P_B , nous avons deux possibilités pour l'écriture de la force et du moment en début de poutre :

$$F_B = F_{Br} \quad \text{ou} \quad F_B = F_{Br} + K_F \left(\frac{\partial \omega_r(x, t)}{\partial t} - \frac{\partial \omega(x, t)}{\partial t} \right) \Big|_{x=0} \quad (3.13)$$

$$\tau_B = \tau_{Br} \quad \text{ou} \quad \tau_B = \tau_{Br} + K_\tau \left(\frac{\partial^2 \omega_r(x, t)}{\partial x \partial t} - \frac{\partial^2 \omega(x, t)}{\partial x \partial t} \right) \Big|_{x=0} \quad (3.14)$$

On suppose que les constantes K_F et K_τ sont positives.

Ces expressions impliquent que $P_B \leq 0$. Elles correspondent aux formules utilisées pour calculer la force et le moment en sortie du contrôleur.

En ce qui concerne le terme P_T , rappelons que nous étudions le cas d'une poutre en libre mouvement. Il n'y a donc ni force, ni moment appliqués par la poutre sur son environnement à son extrémité, donc $F_T = 0$ et $\tau_T = 0$. Il suffit donc de calculer la force et le moment dans le contrôleur par pseudo mode de glissement de la façon suivante :

$$F_{Tr} = EIK_{v1} \left(\frac{\partial \omega_r(x, t)}{\partial t} - \frac{\partial \omega(x, t)}{\partial t} \right) \Big|_{x=L} \quad (3.15)$$

$$\tau_{Tr} = EIK_{v2} \left(\frac{\partial^2 \omega_r(x, t)}{\partial x \partial t} - \frac{\partial^2 \omega(x, t)}{\partial x \partial t} \right) \Big|_{x=L} \quad (3.16)$$

On suppose que les constantes K_{v1} et K_{v2} sont positives. Ceci assure la positivité du terme P_T .

Nous avons donc prouvé que $\dot{E}_L \leq 0$. Donc, nous en déduisons que le contrôleur choisi est stable.

3.4 Conclusion

Nous cherchons à commander une barre flexible libre en mouvement plan. Nous basons notre solution sur le concept de décomposition virtuelle. Cette approche nous fournit une solution stable au sens de Lyapunov à notre problème. Ainsi les trois contrôleurs que nous allons développer par la suite sont basés sur cette

approche.

CHAPITRE 4

PREMIERS CODES DU CONTRÔLEUR

4.1 Premier code de commande simple

4.1.1 Premier choix d'équation pour le contrôleur

4.1.1.1 Contrôleur en pseudo mode de glissement

Dans un premier temps nous avons décidé de commander uniquement la force. En effet le fait de prendre un gain K_{v2} nul dans l'équation 3.16 ne modifie en rien la stabilité du contrôleur établie à la section précédente.

Dans le but d'avoir un modèle simple, nous avons choisi une commande avec un simple proportionnel. En fin de poutre, à la sortie du contrôleur en pseudo mode de glissement, d'après les expressions 3.15 et 3.16 de la force et du moment données dans la section stabilité, nous avons opté pour les équations suivantes :

$$F_{Tr} = EIK_{v1} \left(\frac{\partial \omega_r(L, t)}{\partial t} - \frac{\partial \omega(L, t)}{\partial t} \right)$$

et $\tau_{Tr} = 0$

où $\frac{\partial \omega_r(L, t)}{\partial x}$ est choisi nul par nous et $\frac{\partial \omega(L, t)}{\partial t}$ est mesuré sur le système à l'aide d'un capteur.

4.1.1.2 Boucle de propagation

Comme indiqué à la figure 3.3 et comme expliqué à la section 3.2.3.2, la boucle de propagation comprend cinq entrées : la déformation désirée, les trois premières dérivées spatiales de la déformation en fin de bras ($x = L$) et la dérivée spatiale quatrième de référence en $x = L$, calculée à l'aide de l'accélération désirée.

D'après la section 3.2.3.2, les valeurs qui entrent dans la boucle du contrôleur sont les suivantes :

1. la déformation désirée ω_r au point N, calculée par intégration double (avec conditions initiales nulles) de l'accélération désirée. Nous choisissons le profil de l'accélération désirée.
2. la première dérivée spatiale au point N est choisie nulle $\frac{\partial \omega_r}{\partial x} = 0$
3. comme nous avons choisi dans un premier temps de ne pas commander le moment, la deuxième dérivée spatiale au point N est nulle $\frac{\partial^2 \omega_r}{\partial x^2} = 0$
4. d'après le choix pour la force à la sortie du contrôleur par pseudo mode de glissement, la troisième dérivée spatiale au point N est telle que $\frac{\partial^3 \omega_r}{\partial x^3} = K_{v1} \left(\frac{\partial \omega_r}{\partial t} - \frac{\partial \omega}{\partial t} \right)$
5. la quatrième dérivée spatiale est calculée à l'aide de l'équation de la dynamique 3.1 à partir de l'accélération de référence $\frac{\partial^4 \omega_d}{\partial x^4} = -\frac{\rho}{EI} \frac{\partial^2 \omega_d}{\partial t^2}$

Dans un soucis de simplicité, d'après les équations 3.13 et 3.14 qui assurent la stabilité du système, nous avons d'abord opté pour une sortie de la boucle de propagation de la forme :

$$F_B = F_{Br} = EI \frac{\partial^3 \omega(1, t)}{\partial x^3}$$

$$\tau_B = \tau_{Br} = -EI \frac{\partial^2 \omega(1, t)}{\partial x^2}$$

Ceci correspond aux commandes qui seront injectées au début de la poutre.

4.1.2 Stabilité du système avec ce contrôleur en simulation

La structure de simulation utilisée comprend notre contrôleur modélisé à l'aide de Matlab/Simulink et un système barre modélisé à l'aide du logiciel Symofros. Ce logiciel est développé par l'agence et permet de recréer le comportement dynamique d'une barre en mouvement. Ainsi nous n'avons qu'à injecter notre commande dans le système modélisé et récupérer les données qui nous intéressent en sortie.

4.1.2.1 Constantes du contrôleur

Dans ce cas, les constantes du contrôleur qu'on peut manipuler pour rendre le système stable sont :

- K_{v1} , elle joue sur la vitesse de convergence du système
- λ_1 , elle joue sur le temps de monter de l'erreur de position et donc sur le dépassement

4.1.2.2 Simulation

Afin de faciliter la première étude, nous avons décidé de simuler une barre extrêmement rigide. Les caractéristiques de la poutre sont une longueur $L = 1m$, une masse linéique $\rho = 2kg.m^{-1}$ et une rigidité $EI = 10000N.m^2$. Initialement la barre est à l'horizontale, comme montré en pointillé sur la figure 3.1. On choisit

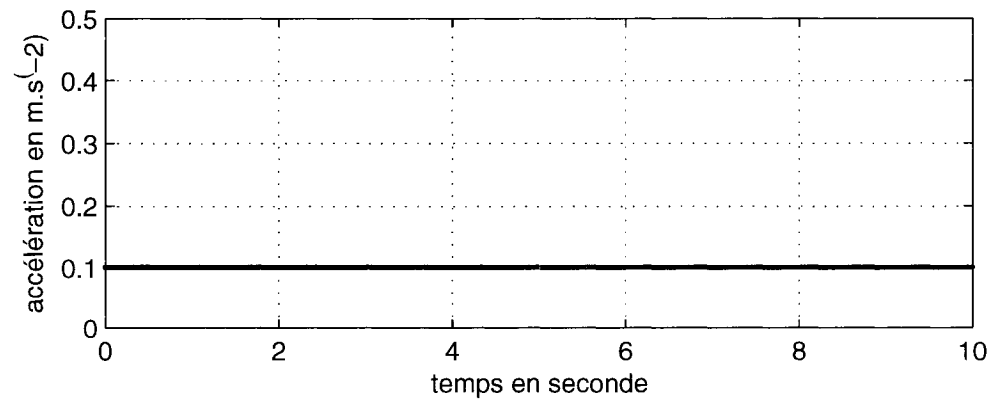


Figure 4.1 Profil d'accélération désirée pour la première loi de commande

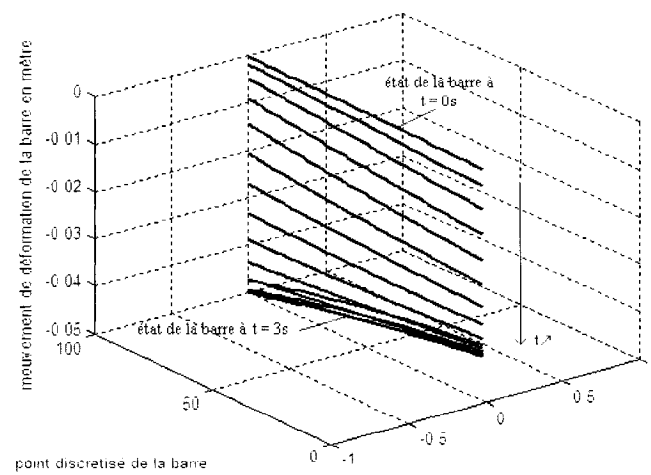


Figure 4.2 Exemple de mouvement d'une barre en simulation

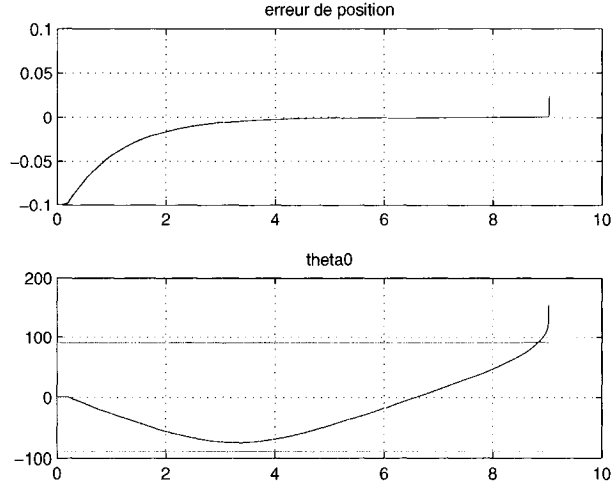


Figure 4.3 Erreur de position et position angulaire initiale avec le premier contrôleur sans contrôleur d'orientation

également un profil d'accélération désirée uniforme de constante $0.1m.s^{-2}$, comme montré à la figure 4.1. La barre bouge alors dans un plan verticale comme indiqué sur la figure 4.2.

Lors de la simulation, nous avons obtenu les courbes à la figure 4.3. Nous avons alors constaté que pour suivre la trajectoire désirée, la barre a tendance à se mettre en position verticale. Or la figure 4.3 nous montre une divergence du système une fois que celui-ci a atteint les 90° . Nous avons mis en évidence les positions $\pm 90^\circ$ sur le graphique pour en faciliter la lecture. L'angle θ_0 , comme montré sur la figure 3.1, est l'angle entre la tangente au premier point de la barre et l'horizontale.

Nous avons refait plusieurs fois les simulations avec d'autres paramètres du contrôleur, mais le résultat est resté le même : une divergence du système quand la barre se met en position verticale.

Nous en avons déduit que la position verticale de la barre est une position singulière et qu'il est préférable de l'éviter si on souhaite commander correctement le système.

4.1.3 Commande de la position angulaire

Afin d'éviter la mise en position singulière de la barre, nous avons décidé de commander également l'angle en fin de poutre. Pour cela nous ajoutons une commande de position angulaire.

4.1.3.1 Modification de notre contrôleur

La commande sera écrite de la même façon que la commande en position linéaire. C'est-à-dire qu'on fait intervenir une variable intermédiaire de commande : la position angulaire de référence θ_r . La figure 3.2 nous indique que l'angle considéré est l'angle que fait la barre avec l'horizontale à une position donnée.

De la même façon que pour la force, elle permet d'assurer que si le moment réel tend vers le moment désiré alors la vitesse angulaire réelle tend vers la vitesse angulaire de référence et donc l'angle en fin de barre tend vers l'angle désiré à cette position.

Pour être en accord avec notre approche basée sur le *flux virtuel de puissance*, la vitesse angulaire de référence est écrite de manière similaire à la vitesse de déformation de référence à l'équation 3.4:

$$\frac{\partial \theta_r(x, t)}{\partial t} = \frac{\partial \theta_d(x, t)}{\partial t} + \lambda_2 [\theta_d(x, t) - \theta(x, t)]$$

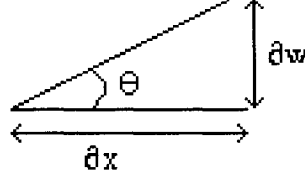


Figure 4.4 Théorie des petits angles

4.1.3.2 Contrôleur en pseudo mode de glissement

Nous n'avons pas modifié le contrôleur en force. L'équation de sortie du contrôleur pour cette partie reste donc :

$$F_{Tr} = EIK_{v1} \left[\frac{\partial \omega_r}{\partial t} - \frac{\partial \omega}{\partial t} \right]$$

Comme énoncé précédemment, afin de commander la position angulaire, il nous faut commander le moment. Pour cela, nous utilisons un asservissement en angle par rétroaction de la vitesse angulaire de la fin de la barre. Nous choisissons alors un contrôleur Proportionnel Intégrateur afin d'avoir une convergence rapide, sans atteinte de la position singulière. La sortie du contrôleur en pseudo mode de glissement s'écrit alors :

$$\tau_{Tr} = -EIK_{v2} \left[\frac{\partial \theta_r(N, t)}{\partial t} - \frac{\partial \theta(N, t)}{\partial t} \right] - EIK_{i2} \int_0^t \left[\frac{\partial \theta_r(N, t)}{\partial t} - \frac{\partial \theta(N, t)}{\partial t} \right] dt \quad (4.1)$$

Remarque : pour de petits angles, d'après le schéma 4.4, nous avons la relation suivante :

$$\frac{\partial \omega}{\partial x} = \tan \theta \approx \theta$$

En remplaçant dans l'équation 4.1, nous obtenons :

$$\begin{aligned}\tau_{Tr} = & -EIK_{v2} \left[\frac{\partial^2 \omega_r(N, t)}{\partial t \partial x} - \frac{\partial^2 \omega(N, t)}{\partial t \partial x} \right] \\ & - EIK_{i2} \int_0^t \left[\frac{\partial^2 \omega_r(N, t)}{\partial t \partial x} - \frac{\partial^2 \omega(N, t)}{\partial t \partial x} \right] dt\end{aligned}$$

Cette formule rappelle l'équation 3.16 qui assure la stabilité du système. Pour que cette dernière soit garantie, il suffit d'ajouter un terme de la forme $K_{I2} \int \left[\frac{\partial \theta_r(N, t)}{\partial t} - \frac{\partial \theta(N, t)}{\partial t} \right]^2 dt$ dans la fonction énergie de Lyapunov E_L . La dérivée de ce terme annulera le terme supplémentaire qui apparaît dans l'expression de P_T à l'équation 3.11, sachant que $\tau_T = 0$.

4.1.3.3 Boucle de propagation

Dans la boucle de propagation, seule l'entrée en rapport avec le moment change. D'après les conditions aux limites 3.3, cela revient à changer l'entrée correspondant à la deuxième dérivée spatiale. Elle aura désormais pour expression :

$$\frac{\partial^2 \omega_r(N, t)}{\partial x^2} = -K_{v2} \left[\frac{\partial \theta_r(N, t)}{\partial t} - \frac{\partial \theta(N, t)}{\partial t} \right] - K_{i2} \int_0^t \left[\frac{\partial \theta_r(N, t)}{\partial t} - \frac{\partial \theta(N, t)}{\partial t} \right] dt$$

Les sorties de la boucle de propagation, soient les commandes du système, restent inchangées :

$$F_B = F_{Br} = EI \frac{\partial^3 \omega_r(1, t)}{\partial x^3} \quad \text{et} \quad \tau_B = \tau_{Br} = -EI \frac{\partial^2 \omega_r(1, t)}{\partial x^2}$$

Remarque : Comme on veut éviter la mise en position singulière et qu'il nous faut respecter la théorie des petits angles, on choisit un angle nul à suivre en fin de

poutre $\theta_r(N, t) = 0$, c'est-à-dire $\frac{\partial \omega_r(N, t)}{\partial x} = 0$. On retrouve la condition que nous avons imposé.

4.1.4 Résultats et limites de ce contrôleur

Afin de comprendre comment se comporte le contrôleur, nous avons effectué des tests pour différentes valeurs des constantes K_v et λ . Ceci nous a permis de déterminer les effets engendrés sur le système par le changement de différentes variables.

4.1.4.1 Variables et leurs effets

4.1.4.1.1 λ_1 et λ_2

λ_1 et λ_2 sont en relation avec le temps de montée, c'est-à-dire le temps que met la courbe d'erreur pour couper, pour la première fois, la droite de la position finale (ici $y = 0$). Plus ces constantes sont grandes plus le temps de montée est petit. Ceci peut éventuellement entraîner une convergence plus rapide du système. Cependant les dépassements engendrés par un petit temps de montée peuvent être mauvais pour le système et surtout entraîner sa divergence, le système n'ayant pas alors le temps de corriger comme il faut l'erreur engendrée si rapidement.

4.1.4.1.2 K_V et K_I

K_V et K_I sont les gains d'un contrôleur Proportionnel Intégrateur classique. Ainsi plus le gain K_V est grand plus la convergence du système est rapide, mais un gain trop grand entraîne sa divergence. Il faut donc faire un compromis lors du choix de K_V .

Pour le gain K_I , plus il est grand plus l'erreur statique du système diminue et

λ_1	1
K_{v1}	10^{-3}
λ_2	1
K_{v2}	10^{-6}
K_{i2}	10^{-4}
N	100

Tableau 4.1 Paramètres du premier contrôleur pour une barre de rigidité $EI = 10000N.m^2$

plus le système converge rapidement. Cependant un grand gain K_I entraîne des oscillations du système ce qui n'est pas souhaitable. Il faut donc faire un compromis lors du choix de K_I .

4.1.4.1.3 N

N est le nombre de points de calcul que l'on prend sur la barre pour effectuer la propagation de la déformation d'un bout à l'autre de la barre.

N a deux limites : une supérieure et une inférieure. Au-delà de ces limites, le système diverge. La première limite, la limite inférieure, est une limite physique. En effet, si on ne prend pas assez de point pour propager le calcul alors le système calcule de façon trop grossière d'un point sur l'autre. Les approximations de la déformation ne sont alors pas adéquates ce qui entraîne la divergence du système. La deuxième limite, la limite supérieure, est une limite numérique. En effet, si on prend trop de points pour effectuer le calcul, on arrive dans des problèmes d'arrondis numériques. Dans ce cas, l'algorithme de commande ne peut faire les calculs correctement et ceci entraîne la divergence du système.

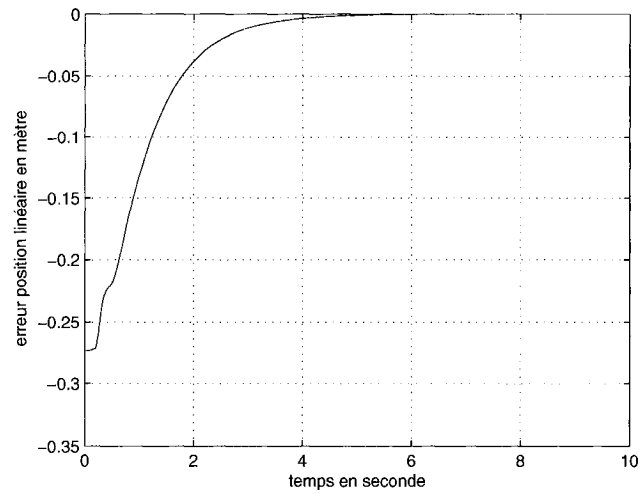


Figure 4.5 Erreur de position linéaire avec le premier contrôleur et une barre très rigide ($EI = 10000N.m^2$)

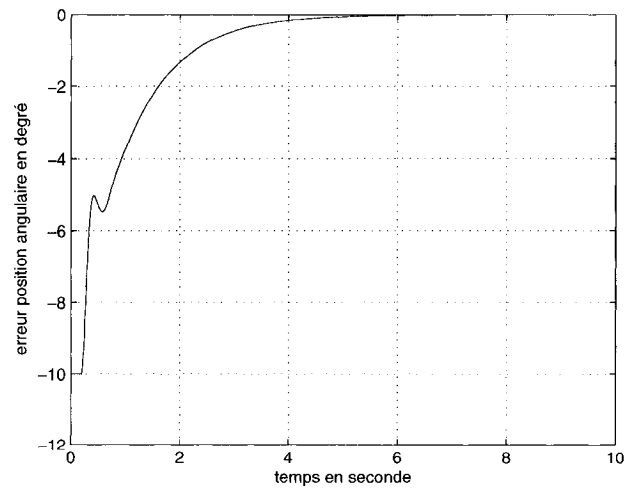


Figure 4.6 Erreur de position angulaire avec le premier contrôleur et une barre très rigide ($EI = 10000N.m^2$)

	$EI = 10000N.m^2$	$EI = 1000N.m^2$	$EI = 510N.m^2$
λ_1	1	0.1	0.01
K_{v1}	10^{-3}	10^{-4}	10^{-4}
λ_2	1	0.05	0.005
K_{v2}	10^{-6}	10^{-8}	10^{-9}
K_{i2}	10^{-4}	10^{-4}	10^{-4}
N	100	100	1000

Tableau 4.2 Paramètres du premier contrôleur pour différentes barres

4.1.4.2 Limites de ce modèle

Nous avons ensuite vérifié qu'avec ce nouveau contrôleur, nous étions capable de stabiliser la barre très rigide avec un profil d'accélération uniforme de valeurs $0.1m.s^{-2}$. Après un ensemble de test essai-erreur, nous avons trouvé les constantes au tableau 4.1 qui satisfont le problème.

D'après les figures 4.5 et 4.6, nous constatons que l'erreur en position linéaire converge en environ 2s (temps de réponse à 5%) et que l'erreur en position angulaire converge en 5s environ. Notre contrôleur semble donc valide.

Le but du contrôleur étant son emploi dans le cadre d'un robot flexible voire très flexible, nous avons décidé de le tester avec une barre extrêmement flexible, de rigidité $EI = 1N.m^2$ pour une longueur L de 1m et une masse linéique ρ de $2kg.m^{-1}$. Cependant nous avons vite constaté qu'avec de tels paramètres, il nous était impossible de trouver les paramètres du contrôleur qui permettaient la convergence du système.

Nous avons alors décidé de réduire la rigidité de la barre progressivement et ainsi trouver la limite du modèle. Après un ensemble de tests sur différents modèles, nous avons constaté que pour une longueur de 1m et une masse linéique de $2kg.m^{-1}$, nous ne pouvions diminuer la rigidité au-delà de $510N.m^2$ sans engendrer la divergence du système.

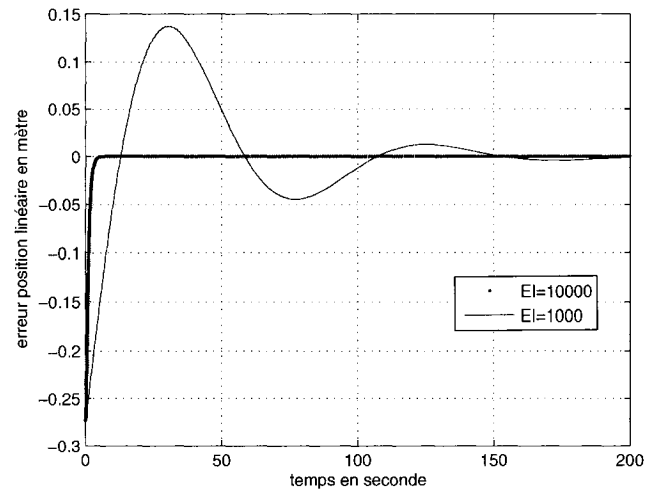


Figure 4.7 Erreur de position linéaire avec le premier contrôleur et différentes barres

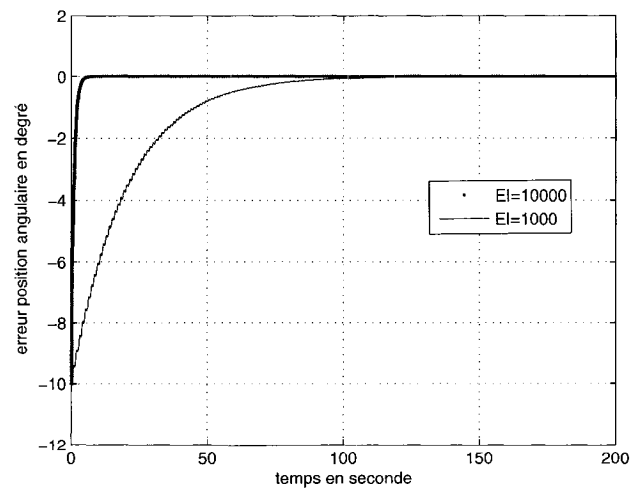


Figure 4.8 Erreur de position angulaire avec le premier contrôleur et différentes barres

Les paramètres du contrôleur correspondant aux trois barres de rigidité différentes simulées sont donnés au tableau 4.2. On constate que plus la rigidité diminue, plus les valeurs de λ_1 , λ_2 et K_{v2} diminuent. En effet le système étant plus flexible, il faut aller moins vite c'est pourquoi on diminue les valeurs de λ . On peut donc supposer que les temps de convergence vont augmenter avec la diminution de la rigidité. Mais il faut également s'assurer qu'on n'atteint pas la position singulière, c'est pourquoi on diminue également K_{v2} . Cela constitue également une raison de l'augmentation du temps de convergence en position angulaire.

Les figures 4.7 et 4.8 donnent les résultats de la simulation de deux barres de rigidité $10000N.m^2$ et $1000N.m^2$. On constate bien que plus la rigidité du système diminue, plus le temps de convergence en position linéaire et angulaire augmente. Par exemple en position linéaire, il faut dix fois plus de temps pour la barre de rigidité $1000N.m^2$ pour converger que pour une barre de rigidité $10000N.m^2$. En effet l'erreur en position linéaire d'une barre de rigidité $1000N.m^2$ converge en 50s environ, pour moins de 2s environ pour la barre de rigidité $10000N.m^2$. Il en va de même en position angulaire, pour la barre de rigidité $1000N.m^2$, il faut 107s pour converger alors qu'il ne faut que 5s pour la convergence angulaire de la barre de rigidité $10000N.m^2$. Ceci est bien conforme à la constatation que nous avons faite sur les valeurs des paramètres du contrôleur. On constate également que plus la barre est flexible plus les dépassements en position linéaire sont importants.

Les figures 4.9 et 4.10 donnent les courbes pour une barre de rigidité $510N.m^2$, à la limite du premier contrôleur, comparées aux autres barres. On constate que le temps de convergence en position linéaire (485s environ) et angulaire (plus de 800s) est beaucoup plus élevé qu'avec des barres de rigidité moindre. Le dépassement en position linéaire est également énorme : 5m d'erreur pour une barre de 1m. Nous en déduisons que nous avons effectivement atteint les limites de commande de ce

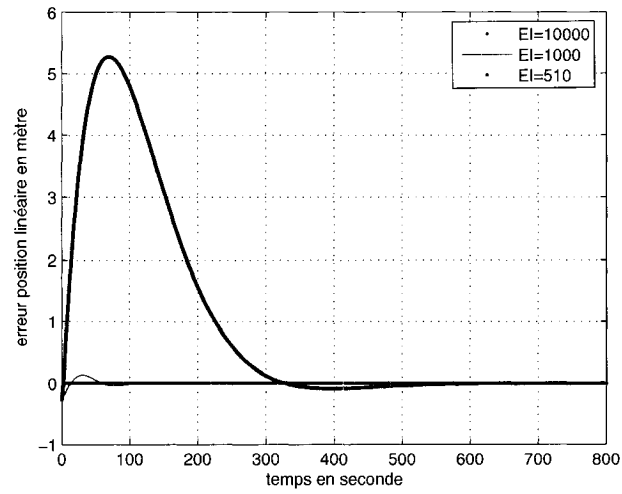


Figure 4.9 Erreur de position linéaire avec le premier contrôleur et sa limite

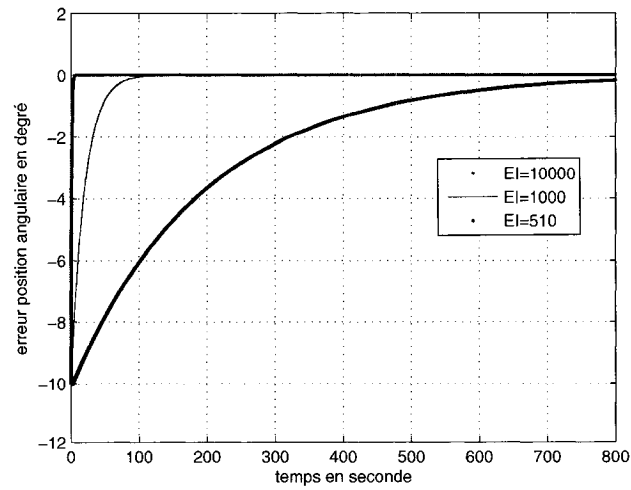


Figure 4.10 Erreur de position angulaire avec le premier contrôleur et sa limite

contrôleur.

Les deux phénomènes, le temps de convergence plus élevé et le dépassement plus conséquent pour des barres plus flexibles, peuvent s'expliquer par la flexibilité de la barre. En effet étant plus flexible, elle se plie plus facilement, d'où les dépassements plus importants. Et les dépassements plus importants impliquent une convergence plus lente car il faut remettre le bon angle et continuer à suivre la bonne trajectoire.

Remarque : Il se peut que pour une longueur plus petite, il soit possible de diminuer encore la rigidité mais nous n'avons pas testé ce cas.

4.1.5 Conclusion

Nous avons montré que ce contrôleur est valide pour faire suivre une trajectoire désirée à une poutre en libre mouvement dans un plan verticale. Cependant nous n'avons pu commander une poutre ($L = 1m$ et $\rho = 2kg.m^{-1}$) de rigidité inférieure à $510N.m^2$. Notre but étant de commander des barres très flexibles, ce contrôleur n'est pas adapté. C'est pourquoi nous avons décidé de le modifier afin de commander des modèles très flexibles.

4.2 Deuxième code du contrôleur

Face aux limites de notre premier contrôleur, nous avons eu recours à l'article (Zhu et al., 1997) pour modifier notre contrôleur.

4.2.1 De nouvelles équations de commandes

4.2.1.1 Contrôleur en pseudo mode de glissement

Afin de diminuer le temps de convergence en position linéaire et d'éviter les éventuelles erreurs statiques, nous avons décidé d'ajouter un intégrateur dans le contrôleur en force :

$$F_{Tr} = EI \left(K_{v1} \left[\frac{\partial \omega_r(L, t)}{\partial t} - \frac{\partial \omega(L, t)}{\partial t} \right] + K_{i1} \int \left[\frac{\partial \omega_r(L, t)}{\partial t} - \frac{\partial \omega(L, t)}{\partial t} \right] dt \right)$$

Cette expression rappelle celle à l'équation 3.15 qui assure la stabilité du système. Pour garantir cette dernière, il suffit d'ajouter un terme en $K_{i1} \int \left[\frac{\partial \omega_r}{\partial t} - \frac{\partial \omega}{\partial t} \right]^2 dt$ à la fonction énergie de Laypunov E_L . La dérivée de ce terme annulera le terme supplémentaire dans l'expression de P_T à l'équation 3.11, sachant que $F_T = 0$.

L'expression du moment à la sortie du contrôleur reste inchangée.

4.2.1.2 Boucle de propagation

Comme dans le premier modèle du contrôleur, les équations de propagation utilisées ainsi que les méthodes de calculs sont les mêmes. Nous avons décidé de changer uniquement les commandes à la sortie du contrôleur. Nous avons essayé de trouver d'autres expressions pour la force et le moment, qui soient similaires aux équations 3.13 et 3.14. Ainsi la stabilité du système sera toujours assurée.

4.2.1.3 Changement des équations liées à la force

Dans le premier modèle la force de commande avait pour expression :

$$F_B(1, t) = EI \frac{\partial^3 \omega(1, t)}{\partial x^3}$$

Comme énoncé dans le chapitre précédent à l'équation 3.13, il existe une autre manière d'écrire la force au début de la barre et garantir la stabilité du système.

Nous choisissons d'exprimer la nouvelle force de commande de la façon suivante :

$$F_B(1, t) = EI \left(\frac{\partial^3 \omega_r(1, t)}{\partial x^3} + K_F \left[\frac{\omega_r(1, t) - \omega_r(1, t - \Delta T)}{\Delta T} - \frac{\partial \omega_0(1, t)}{\partial t} \right] \right)$$

où ω_0 est la déformation en début de poutre mesurée sur le système.

Or d'après le développement en série de Taylor tronquée au premier ordre, donné à l'équation 3.6, on a :

$$\frac{\omega_r(1, t) - \omega_r(1, t - \Delta T)}{\Delta T} = \frac{\partial \omega_r(1, t)}{\partial t}$$

On peut alors réécrire la force de la façon suivante :

$$F_B(1, t) = EI \left(\frac{\partial^3 \omega_r(1, t)}{\partial x^3} + K_F \left[\frac{\partial \omega_r(1, t)}{\partial t} - \frac{\partial \omega_0(1, t)}{\partial t} \right] \right)$$

Ceci correspond bien à la formule énoncée à l'équation 3.13. La stabilité du système est toujours assurée.

4.2.1.4 Changement du moment de commande

Dans le premier modèle du contrôleur le moment de commande avait pour expression :

$$\tau_B(1, t) = -EI \frac{\partial^2 \omega(1, t)}{\partial x^2}$$

Comme énoncé dans le chapitre précédent à l'équation 3.14, il existe une autre façon d'écrire le moment qui agit au début de la barre. Nous choisissons l'expression suivante :

$$\tau_B(1, t) = -EI \left(\frac{\partial^2 \omega_r(1, t)}{\partial x^2} + K_\tau \left[\frac{1}{\Delta T} \left(\frac{\partial \omega_r(1, t)}{\partial x} - \frac{\partial \omega_r(1, t - \Delta T)}{\partial x} \right) - \frac{\partial \theta_0(1, t)}{\partial t} \right] \right)$$

où θ_0 est l'angle entre la tangente au premier point de la poutre et l'horizontale, mesuré sur le système.

Or d'après le développement en série de Taylor tronqué au premier ordre, donné à l'équation 3.6, on a :

$$\frac{1}{\Delta T} \left(\frac{\partial \omega_r(1, t)}{\partial x} - \frac{\partial \omega_r(1, t - \Delta T)}{\partial x} \right) = \frac{\partial}{\partial t} \left(\frac{\partial \omega_r(1, t)}{\partial x} \right)$$

Comme on suppose qu'on a affaire à de petits angles, on a également :

$$\frac{\partial \omega}{\partial x} = \theta$$

On peut alors réécrire le moment de la façon suivante :

$$\tau_B(1, t) = -EI \left(\frac{\partial^2 \omega_r(1, t)}{\partial x^2} - K_\tau \left[\frac{\partial \theta_r(1, t)}{\partial t} - \frac{\partial \theta_0(1, t)}{\partial t} \right] \right)$$

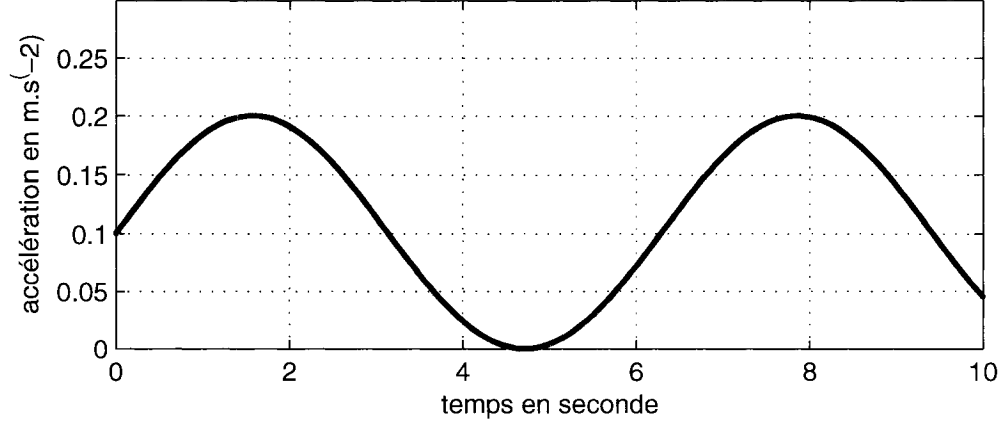


Figure 4.11 Profil d'accélération suivi par les deux dernières lois de commande

Ceci correspond bien à la formule énoncée à l'équation 3.14. La stabilité du système est toujours assurée.

4.2.2 Résultats et limites de ce modèle

Remarque : dans toutes les simulations qui suivent, on suppose qu'initialement la barre est à l'horizontale avec un angle en début de poutre de 10° . Le profil d'accélération à suivre est sinusoïdal (figure 4.11), pour diminuer les risques d'atteinte de la position singulière.

Nous avons tout d'abord vérifié que ce contrôleur était capable de stabiliser une barre très rigide ($L = 1m$, $\rho = 2kg.m^{-1}$, $EI = 10000N.m^2$). Les figures 4.12 et 4.13 nous permettent d'affirmer que ce contrôleur est fonctionnel. En effet, l'erreur de position linéaire converge en environ 2s et l'erreur de position angulaire converge en environ 5s.

Puis nous avons réduit la rigidité progressivement pour trouver la limite de ce système. Diminuer progressivement la rigidité, nous a permis de changer les paramè-

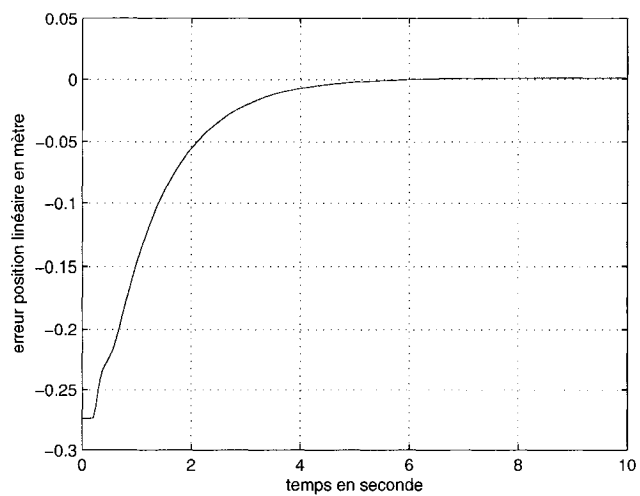


Figure 4.12 Erreur de position linéaire avec le deuxième contrôleur et une barre très rigide

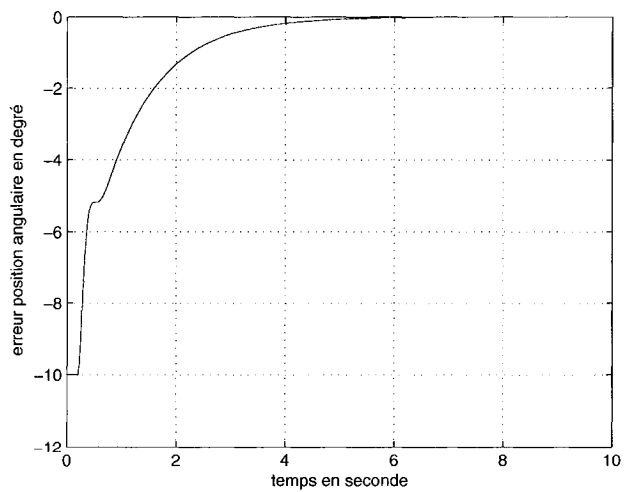


Figure 4.13 Erreur de position angulaire avec le deuxième contrôleur et une barre très rigide

	$EI = 10000N.m^2$	$EI = 500N.m^2$	$EI = 50N.m^2$
λ_1	1	0.1	0.01
K_{v1}	10^{-3}	10^{-5}	10^{-8}
K_{i1}	10^{-5}	10^{-7}	10^{-5}
λ_2	1	0.05	0.01
K_{v2}	10^{-6}	10^{-8}	10^{-10}
K_{i2}	10^{-4}	10^{-5}	10^{-6}
K_F	2.10^{-5}	2.10^{-3}	2.10^{-4}
K_T	10^{-4}	10^{-3}	10^{-4}

Tableau 4.3 Paramètres du deuxième contrôleur pour différentes barres

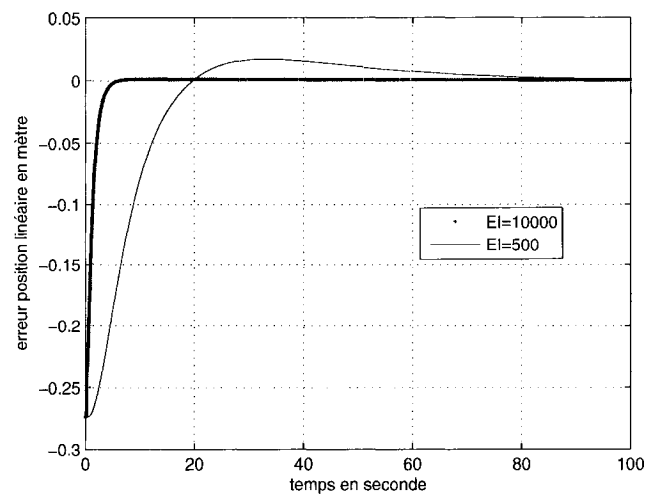


Figure 4.14 Erreur de position linéaire avec le deuxième contrôleur pour différentes barres

tres du contrôleur au fur et à mesure. Ceci a facilité la découverte de la limite du modèle.

Nous avons alors constaté que pour une longueur de 1m et une masse linéique de $2kg.m^{-1}$, nous ne pouvions diminuer la rigidité au-delà de $39N.m^2$ sans engendrer la divergence du système.

Le tableau 4.3 donne les paramètres du contrôleur pour trois barres différentes de rigidité $10000N.m^2$, $500N.m^2$ et $50N.m^2$, de longueur 1m et de masse linéique

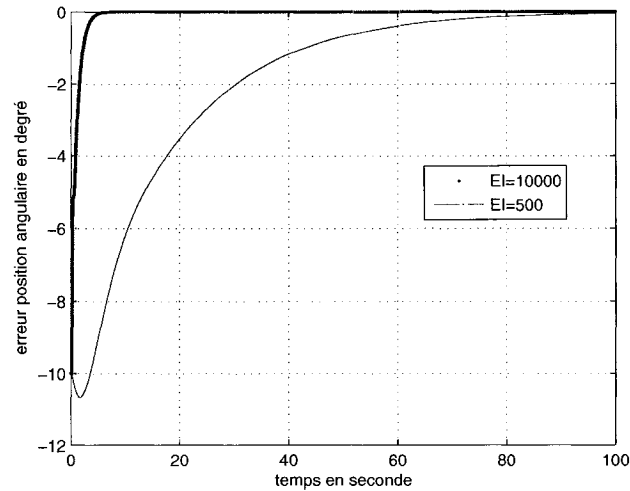


Figure 4.15 Erreur de position angulaire avec le deuxième contrôleur pour différentes barres

$2kg.m^{-1}$. Ces valeurs sont trouvées par test essai-erreur. On constate que plus la rigidité diminue plus les constantes du contrôleur diminuent. Le temps de convergence sera donc vraisemblablement de plus en plus long au fur et à mesure que l'on diminue la rigidité.

Les figures 4.14 et 4.15 donnent les résultats de la simulation des deux premières barres ($EI = 10000N.m^2, EI = 500N.m^2$). On constate que plus la barre est flexible, plus le temps de convergence en position linéaire et angulaire est grand. Ceci est conforme à ce que nous avons constaté avec les valeurs du contrôleur.

De plus d'après les figures 4.16 et 4.17 on peut déduire qu'on commence à atteindre les limites du système avec une barre de rigidité $EI = 50N.m^2$. En effet le système ne suit pas du tout les deux autres. Il converge en réalité en 170s pour la position linéaire et en 535s pour la position angulaire. Ces valeurs sont 5 à 10 fois plus grandes qu'avec les deux autres barres.

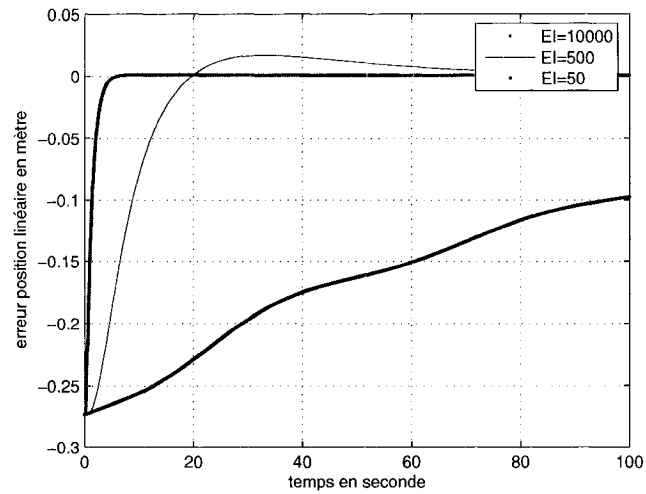


Figure 4.16 Erreur de position linéaire avec le deuxième contrôleur pour différentes barres

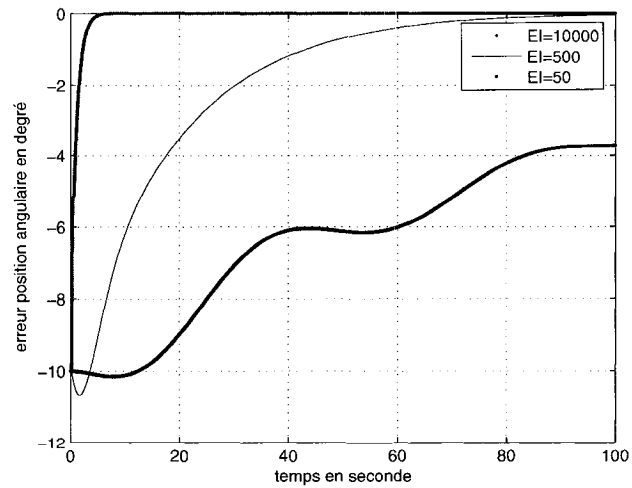


Figure 4.17 Erreur de position angulaire avec le deuxième contrôleur pour différentes barres

4.2.3 Conclusion

Nous avons montré que ce contrôleur est valide pour stabiliser une poutre en libre mouvement à l'horizontale. Nous avons aussi diminué d'un facteur 10 la limite de la rigidité contrôlable, ce qui est tout à fait acceptable. Cependant nous n'avons pas encore atteint notre but. C'est pourquoi nous avons décidé de modifier notre contrôleur dans le but de stabiliser des modèles très flexibles.

CHAPITRE 5

TROISIÈME CODE DU CONTRÔLEUR

Comme nous n'avons pu commander des barres très flexibles avec les méthodes précédentes, nous avons eu recours à une autre méthode. Etant donné que le deuxième code était le plus performant, nous l'avons repris pour le modifier en nous aidant de la méthode des filtres de Kalman. Cette dernière permet de résoudre les problèmes de stabilité numérique que nous rencontrons lors du calcul de notre algorithme de commande.

Dans la théorie du filtre de Kalman, les informations de mesure sont utilisées pour générer des corrections et augmenter la compensation du système vis-à-vis des sources d'erreurs critiques. Face à notre problème, plus la rigidité diminue plus il faut diminuer les gains. Nous arrivons donc à des problèmes d'arrondi numérique. L'avantage de la théorie des filtres de Kalman est que cela va créer un filtrage passe-bas. On va donc réduire la propagation des erreurs d'arrondi et des bruits de mesure, ce qui va nous donner de meilleures valeurs de calcul.

Dans notre cas, le système étudié est discret. Nous utiliserons donc la méthode des filtres discrets de Kalman pour calculer de manière prédictives les valeurs dans la propagation du contrôleur d'un bout à l'autre de la barre. Dans cette section, nous allons énoncer les principales lignes de fonctionnement de cette méthode et en discuter les avantages et les inconvénients.

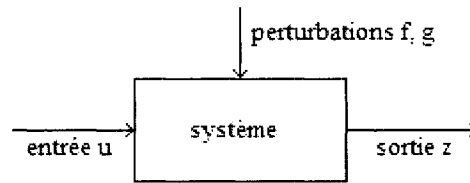


Figure 5.1 Schéma d'un système réel

5.1 Equations d'état d'un système réel

Soit le système, représenté à la figure 5.1, de vecteur d'entrée u , de vecteur de sortie ou d'observation z et de vecteur d'état x . Les perturbations f et g , qui agissent sur le système sont des bruits blancs gaussiens. On suppose le système invariant avec le temps. Les équations d'état du système sont les suivantes :

$$\begin{aligned} x[k+1] &= Ax[k] + Bu[k] \\ z[k] &= Cx[k] + Du[k] \end{aligned}$$

avec

- A la matrice d'état du système
- B la matrice de commande
- C la matrice d'observation
- D une matrice qu'on suppose nulle.

Comme il s'agit d'un système linéaire, la solution est :

$$x[k+1] = \phi[k+1, 0]x[k] + \sum_{i=0}^{k+1} \phi[k+1, i]Bu[k]$$

où $\phi[k+1, k]$ est la matrice de transition du système. Elle transforme un état donné à l'instant k en un autre à l'instant $k+1$.

On peut alors calculer l'état au pas suivant à partir de celui au pas juste avant :

$$x[k+1] = \phi[k+1, k]x[k] + \phi[k+1, k]Bu[k]$$

En réalité, le système soumis à un certain bruit de façon interne, les bruits du système, et de façon externe, les bruits de mesure. Ce sont les perturbations du système. L'équation d'état du système devrait donc être écrite :

$$\begin{aligned} x[k+1] &= Ax[k] + Bu[k] + \Gamma_1 f[k] \\ z[k] &= Cx[k] + Du[k] + \Gamma_2 f[k] + g[k] \end{aligned}$$

où

- $f[k] \equiv N(0, Q(k))$ est un bruit blanc gaussien et Q la matrice de covariance des bruits du système
- $g[k] \equiv N(0, R(k))$ est un bruit blanc gaussien et R la matrice de covariance des bruits de mesure
- Γ_1 et Γ_2 sont les matrices des coefficients des bruits du système

5.2 Rappel sur la méthode de Kalman

Cette partie est basée sur l'ouvrage de Siouris (Siouris, 1996).

5.2.1 Méthode de prédiction-correction

La méthode du filtre de Kalman opère d'une façon prédiction-correction. A partir d'un état actuel au temps k , prédit, elle permet de prédire l'état au temps $k + 1$. Puis une fois l'état au temps k observé, on corrige la valeur de l'état prédit au temps $k + 1$. L'équation utilisée pour se faire est la suivante :

$$\hat{x}(k+1) = \underbrace{\phi\hat{x}(k)}_{\text{a- prédiction}} + \underbrace{K(k+1)\left[z(k+1) - C\phi\hat{x}(k)\right]}_{\text{b- correction}}$$

avec

- $\hat{x}(k)$ le meilleur estimé du signal $x(k)$ à l'instant k
- a prédit l'estimé du signal au temps $k + 1$ en projetant l'estimée du signal au temps k dans le futur à l'aide de la matrice de transition du système
- la première partie de b est l'observation faite au temps $k + 1$ et la deuxième partie de b est l'observation prédite au temps k . On pondère l'erreur d'observation à l'aide de la matrice des *gains de Kalman* K . b permet de corriger l'observation prédite.

Cette équation détermine de combien on va modifier le premier estimé au temps $k + 1$ sur la base de notre nouvelle observation au temps $k + 1$. Elle fournit ainsi le meilleur estimé du signal au temps $k + 1$.

5.2.2 Gains du filtre

La matrice des gains de Kalman est trouvée à l'aide de l'équation suivante :

$$K(k) = \underbrace{P(k|k-1)H^T(k)}_{\text{erreur antérieure}} \underbrace{\left[H(k)P(k|k-1)H^T(k) + R(k) \right]^{-1}}_{\text{bruit de mesure}} \quad (5.1)$$

où $P(k) = E\left[(x(k) - \hat{x}(k))(x(k) - \hat{x}(k))^T\right]$. Il s'agit de la matrice de covariance des erreurs de prédiction. K est calculée de telle sorte qu'on minimise ces erreurs.

Le gain $K(k)$ correspond à l'importance accordée aux mesures nouvelles pour modifier les estimés de l'état. Il dépend fortement de notre façon de caractériser le bruit de mesure $R(k)$.

Remarque : Dans le cas d'un système variant dans le temps, $K(k)$ change avec le temps pour assurer l'optimum. Nous considérons un système invariant, les gains du filtre seront donc constants.

D'après l'équation 5.1:

- Si la mesure est effectuée avec un bruit de mesure relativement plus petit que l'erreur antérieure, alors le gain K est plus grand. Cela signifie qu'on a une plus grande confiance dans les mesures d'observation.
- Si la mesure effectuée est bruitée, relativement à l'erreur antérieure, alors le gain K est plus petit. Cela signifie qu'on accorde moins de confiance aux mesures mais plus au modèle, donc à nos estimées.

5.2.3 Matrices de covariance des bruits

Q est la matrice de covariance des bruits du système. Elle dépend des paramètres d'état estimés et compense les imprécisions du modèle. Cette matrice détermine le taux de confiance qu'on met dans nos estimés, donc dans notre modèle. Plus Q sera petite (proche de 0), plus la confiance accordée aux estimés sera grande.

R est la matrice de covariance des bruits de mesure. Elle dépend de la précision de l'appareil de mesure. Cette matrice détermine le taux de confiance qu'on met dans nos mesures. Plus R sera petite (proche de 0), plus la confiance accordée aux mesures sera grande.

5.2.4 Procédure du filtre de Kalman

Le schéma d'estimation a deux étapes distinctes, à savoir :

1. la mise à jour temporelle des estimées et de la matrice de covariance des erreurs
2. la mise à jour des quantités correspondantes dans les mesures

Remarque : Dans le logiciel Simulink, la fonction *Kalman* calcule directement la matrice des gains du filtre K à partir d'un système $SY S$, de la matrice de covariance des bruits du système Q , de la matrice de covariance des bruits de mesure R , de la matrice corrélation N entre les bruits du système et de mesure et des matrices Γ_1 et Γ_2 des coefficients des bruits du système. Nous choisissons ces six paramètres.

5.3 Application de cette méthode à notre contrôleur

5.3.1 Nouvelles équations

5.3.1.1 Equations du système

Les équations d'état du système considérées sont :

$$\begin{aligned}x[k+1] &= Ax[k] + Bu[k] + \Gamma_1 f[k] \\ y[k] &= Cx[k] + Du[k] + \Gamma_2 f[k] + g[k]\end{aligned}$$

avec

- . f et g les bruits du système et de mesure respectivement, ce sont des bruits blancs gaussiens
- . x le vecteur d'état. Il comprend les dérivées spatiales et temporelles de la déformation de la poutre
- . u le vecteur de la quatrième dérivée spatiale de la déformation.

Comme l'application de la méthode des filtres de Kalman nécessite une mise à jour à la fois temporelle et spatiale, nous découpons le système en fonction des dérivées spatiales et temporelles. Nous obtenons deux systèmes d'équations:

- pour le spatial :

$$X_x = \left[\omega_x(n, t) \quad \frac{\partial \omega_x(n, t)}{\partial x} \quad \frac{\partial^2 \omega_x(n, t)}{\partial x^2} \quad \frac{\partial^3 \omega_x(n, t)}{\partial x^3} \right]'$$

$$A_x = \begin{bmatrix} 1 & -\Delta X & \frac{1}{2}(\Delta X)^2 & -\frac{1}{6}(\Delta X)^3 \\ 0 & 1 & -\Delta X & \frac{1}{2}(\Delta X)^2 \\ 0 & 0 & 1 & -\Delta X \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$B_x = \begin{bmatrix} 0 & 0 & 0 & -\Delta X \end{bmatrix}'$$

$$\Gamma_{1x} = \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix}'$$

$$C_x = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}$$

$$D_x = 0$$

$$\Gamma_{2x} = 0$$

- pour le temporel :

$$X_t = \begin{bmatrix} \omega_t(n, t) & \frac{\partial \omega_t(n, t)}{\partial t} & \frac{\partial^2 \omega_t(n, t)}{\partial t^2} \end{bmatrix}'$$

$$A_t = \begin{bmatrix} 1 & -\Delta T & \frac{1}{2}(\Delta T)^2 \\ 0 & 1 & -\Delta T \\ 0 & 0 & 1 \end{bmatrix}$$

$$B_t = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}'$$

$$\Gamma_{1t} = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}'$$

$$C_t = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}$$

$$D_t = 0$$

$$\Gamma_{2t} = 0$$

Le développement en série de Taylor écrit jusqu'à l'ordre 3 a pour formule :

$$\omega(n-1, t) = \omega(n, t) - \Delta X \frac{\partial \omega(n, t)}{\partial x} + \frac{\Delta X^2}{2} \frac{\partial^2 \omega(n, t)}{\partial x^2} - \frac{\Delta X^3}{6} \frac{\partial^3 \omega(n, t)}{\partial x^3}$$

Nous constatons alors que pour calculer la déformation et ses dérivées au pas spatial ou temporel suivant, le développement en série de Taylor est utilisé à travers les matrices A_x et A_t . Les propagations spatiale et temporelle sont juste exprimées dans les équations d'état à l'aide de ce développement plus un certain bruit.

De plus on constate que le bruit n'intervient que sur la dernière variable des vecteurs d'état X_x et X_t . En effet pour calculer la valeur de $\frac{\partial^3 \omega(n-1,t)}{\partial x^3}$ (resp. de $\frac{\partial^2 \omega(n,t-\Delta T)}{\partial t^2}$), nous n'avons accès qu'à la valeur de $\frac{\partial^3 \omega(n,t)}{\partial x^3}$ ($\frac{\partial^2 \omega(n,t)}{\partial t^2}$). Ceci implique une grande erreur en ΔX (ΔT resp.) relativement aux autres variables. C'est pourquoi on lui ajoute un bruit.

Remarque : Pour calculer les gains des filtres de Kalman dans les deux cas, il nous faut connaître les matrices de corrélations et de covariances des bruits. Nous choisissons pour commencer :

$$\cdot Q_x = Q_t = .00001$$

$$\cdot R = 1$$

$$\cdot N = 0$$

5.3.1.2 Boucle de propagation

Remarque : En ce qui concerne le contrôleur en pseudo mode de glissement, ses équations restent inchangées à celles utilisées dans le chapitre précédent. A savoir, à la sortie du contrôleur, nous avons accès à la force et au moment en fin de poutre de la forme :

$$\begin{aligned} F_{Tr} &= EI \left(K_{v1} \left[\frac{\partial \omega_r(L,t)}{\partial t} - \frac{\partial \omega(L,t)}{\partial t} \right] + K_{i1} \int \left[\frac{\partial \omega_r(L,t)}{\partial t} - \frac{\partial \omega(L,t)}{\partial t} \right] dt \right) \\ \tau_{Tr} &= -EI \left(K_{v2} \left[\frac{\partial \theta_r(L,t)}{\partial t} - \frac{\partial \theta(L,t)}{\partial t} \right] + K_{i2} \int \left[\frac{\partial \theta_r(L,t)}{\partial t} - \frac{\partial \theta(L,t)}{\partial t} \right] dt \right) \end{aligned}$$

En ce qui concerne la boucle de propagation, nous avons changé les équations de propagation pour faire intervenir une mise à jour à la fois spatiale et temporelle qui s'inspirent de la théorie des filtres de Kalman. Tout d'abord on initialise la propagation à partir des données à la sortie du contrôleur en pseudo mode de glissement :

$$X_x(N) = \left[\omega_r(N, t) \quad \frac{\partial \omega(N, t)}{\partial x} \quad \frac{\partial^2 \omega(N, t)}{\partial x^2} \quad \frac{\partial^3 \omega(N, t)}{\partial x^3} \right]' \text{ et } u(N, t) = \frac{\partial^4 \omega_r(N, t)}{\partial x^4} :$$

$$X_x(N-1) = A_x X_x(N) + B_x u(N, t)$$

Ceci nous donne un premier estimé de l'état à la position $N-1$ au temps t .

Puis on propage cet estimé en position de la déformation de la fin vers le début de la barre, selon la méthode prédiction-correction de Kalman :

$$X_x(n-1) = \underbrace{A_x X_x(n) + B_x \left(-\frac{\rho}{EI} \frac{\partial^2 \omega_t(n, t)}{\partial t^2} \right)}_{\text{prédiction}} + \underbrace{L_x \left(\omega_t(n, t) - \omega_x(n) \right)}_{\text{correction}}$$

$n \leq N-1$

où

- $\omega_t(n, t)$ est l'estimé temporel au temps t de la déformation au point n
- $\omega_x(n)$ est l'estimé spatial de la déformation au point n
- L_x est la matrice des gains du filtre de Kalman pour la partie spatiale

Le première partie de l'équation (deux premiers termes) rappelle l'équation d'état et donc le développement en série de Taylor. Elle permet donc de calculer un premier estimé au pas $n-1$.

La deuxième partie (dernier terme) est le filtrage de Kalman. Elle permet de corriger l'estimé calculé dans la première partie à partir de l'erreur de prédiction.

Remarque : Nous cherchons à connaître précisément la déformation de la barre à chaque point. Ainsi tous les estimés des dérivées spatiales et temporelles que l'on calcule au point n , ne servent qu'à estimer la déformation au point $n - 1$ avec précision. C'est donc cette erreur de prédiction qui nous intéresse, c'est la plus pertinente.

De plus la prédiction de la déformation au point n est calculée à la fois spatialement par $\omega_x(n)$ et temporellement par $\omega_t(n, t)$. Dans le cas de la propagation spatiale, on suppose que l'estimée temporelle joue le rôle d'observation ; d'où l'expression de la deuxième partie de l'équation.

Remarque : Initialement on suppose que $X_t = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}'$.

Puis, on recalcule l'estimé temporel selon la méthode prédiction correction de Kalman :

$$X_t(n, t) = \underbrace{A_t X_t(n, t - 1)}_{\text{prédiction}} + \underbrace{L_t \left(\omega_x(n) - \omega_t(n, t - 1) \right)}_{\text{correction}}$$

où L_t est la matrice des gains du filtre de Kalman pour la partie temporelle.

La première partie de l'équation correspond à l'équation d'état donc le développement en série de Taylor. Elle permet de calculer un premier estimé temporelle au temps t à partir de celle au temps $t - 1$.

La deuxième partie de l'équation correspond au filtrage de Kalman. De la même façon que pour la propagation spatiale, seule la déformation intervient de façon pertinente. Mais dans ce cas, comme on effectue une propagation spatiale, on suppose que l'estimé spatial joue le rôle d'observation et l'estimé temporel au temps $t - 1$, celui de premier estimé.

Comme nous avons décidé d'utiliser un modèle de contrôleur similaire à celui du

chapitre précédent, les équations de sorties de la boucle de propagation doivent être de la forme :

$$\begin{aligned} F_B(1, t) &= EI \left(\frac{\partial^3 \omega_r(1, t)}{\partial x^3} + K_F \left[\frac{\partial \omega_r(1, t)}{\partial t} - \frac{\partial \omega_0(1, t)}{\partial t} \right] \right) \\ \tau_B(1, t) &= -EI \left(\frac{\partial^2 \omega_r(1, t)}{\partial x^2} + K_\tau \left[\frac{\partial \theta_r(1, t)}{\partial t} - \frac{\partial \theta_0(1, t)}{\partial t} \right] \right) \end{aligned}$$

Nous constatons que nous avons également besoin de prédire la valeur de l'angle en début de poutre. Pour cela nous choisissons le vecteur angulaire suivant : $\theta_t(t) = \left[\theta_B(t) \quad \frac{\partial \theta_b(t)}{\partial t} \quad \frac{\partial^2 \theta_B}{\partial t^2} \right]$ où θ_B est l'angle estimé au premier point de la poutre.

On propage les valeurs temporelles de cet angle selon la méthode de prédiction-correction de Kalman :

$$\theta_B(t) = \underbrace{A_t \theta_B(t-1)}_{\text{prédiction}} + \underbrace{L_t \left(\frac{\partial \omega_x(1)}{\partial x} - \theta_B(t-1) \right)}_{\text{correction}}$$

De la même façon que pour les deux autres propagation, la première partie de l'équation correspond au calcul d'un premier estimé de l'état angulaire à l'instant t .

La deuxième partie de l'équation correspond à la partie correction à l'aide du filtrage de Kalman. Dans ce cas, comme la première dérivée spatiale est liée à l'angle d'après la théorie des petits angles, on suppose que $\frac{\partial \omega_x(1)}{\partial x}$ est l'observation qu'on fait de l'angle initial au temps t .

Nous avons ainsi propagé toutes les variables pertinentes d'un bout à l'autre de la poutre selon la méthode des filtres de Kalman.

Il ne nous reste plus qu'à écrire nos équations de commande, conformément aux équations 3.13 et 3.14, à l'aide de nos prédictions et de l'observation faite sur le

système de l'angle et de la déformation au premier point de la barre :

$$\begin{aligned} F_B &= \frac{\partial^3 \omega_x(1)}{\partial x^3} + K_F \left(\frac{\partial \omega_t(1, t)}{\partial t} - \frac{\partial \omega_0(1, t)}{\partial t} \right) \\ \tau &= \frac{\partial^2 \omega_x(1)}{\partial x^2} - K_\tau \left(\frac{\partial \theta_B(t)}{\partial t} - \frac{\partial \theta_0(1, t)}{\partial t} \right) \end{aligned}$$

5.4 Résultats et limites de ce contrôleur

Le contrôleur qui s'inspire de la méthode des filtres de Kalman est le plus complet de tous les contrôleurs que nous avons développés. En effet, il stabilise toutes sortes de système, du très rigide comme un robot spatiale typique de rigidité $EI = 1.08e^7 N.m^2$ au très flexible de rigidité $EI = 1 N.m^2$ pour une longueur de 1m et une masse linéique de $2kg.m^{-1}$.

Remarque : dans toutes les simulations qui suivent, on suppose qu'initialement la barre est à l'horizontale avec un angle en début de poutre de 10° . Le profil d'accélération à suivre est sinusoïdal (figure 4.11), pour diminuer les risque d'atteinte de la position singulière.

5.4.1 Modèle très flexible

Tout d'abord nous avons vérifié que ce contrôleur comblait les lacunes des deux autres, à savoir stabilisait une poutre très flexible. Nous avons pris pour exemple, une barre de longueur $L = 1m$, de masse linéique $\rho = 2kg.m^{-1}$ et de rigidité de flexion $EI = 1 N.m^2$. Initialement on suppose la barre en position horizontale avec un angle θ_0 de 10° .

Après un ensemble de test essai-erreur, nous avons trouvé les paramètres du contrôleur au tableau 5.1. On constate que Q est relativement petit, on fait donc

λ_1	0.1
K_{v1}	10^{-3}
K_{i1}	10^{-5}
λ_2	0.05
K_{v2}	10^{-5}
K_{i2}	10^{-9}
K_F	10^{-3}
K_T	10^{-3}
Q_x	10^{-5}
Q_t	10^{-5}

Tableau 5.1 Paramètres du troisième contrôleur pour une barre très flexible

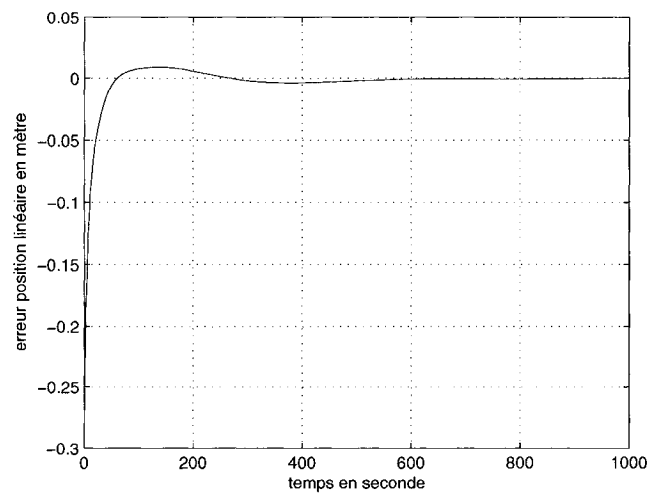


Figure 5.2 Erreur de position linéaire d'une barre très flexible ($EI = 1N.m^2$) avec le troisième contrôleur

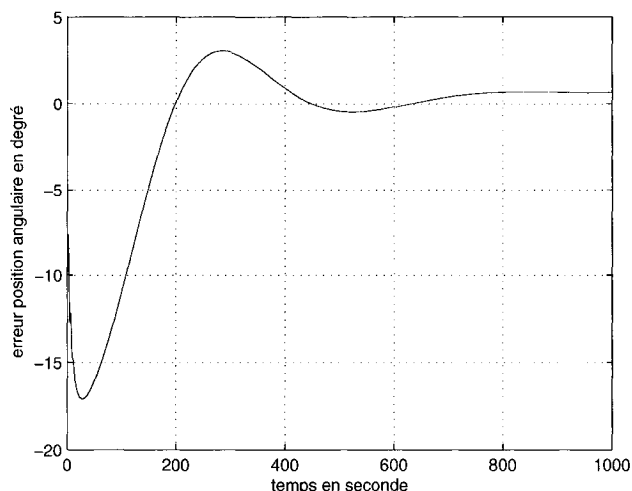


Figure 5.3 Erreur de position angulaire d’une barre très flexible avec le troisième contrôleur ($EI = 1N.m^2$)

confiance à notre modèle.

D’après les figures 5.2 et 5.3 nous voyons que le système converge en position linéaire en environ 7s (temps de réponse à 5%), ce qui est très rapide. Cependant l’erreur angulaire se stabilise en environ 600s et elle ne s’annule pas. Elle se stabilise autour de 0.2° . Cette erreur angulaire étant très petite, on peut considérer qu’elle est due à la flexibilité de la barre.

Cependant un tel modèle n’est pas très efficace en ce qui concerne le temps de calcul. En effet nous avons constaté qu’au cours de la simulation, qu’il fallait 3100s ($\cong 52min$) environ pour effectuer la simulation sur 1000s. Ceci nous donne à peu près un ratio de 3s de calculs pour 1s de simulation.

Remarque : Une animation du mouvement de ce type de barre (“initialRAP.avi”) est fournie sur le CD connexe. On y voit bien le mouvement flexible des trois premières secondes.

λ_1	0.2
K_{v1}	10^{-5}
K_{i1}	10^{-5}
λ_2	100
K_{v2}	10^{-6}
K_{i2}	0
K_F	2.10^{-5}
K_T	10^{-4}
Q_x	10^{-9}
Q_t	10^{-3}

Tableau 5.2 Paramètres du troisième contrôleur pour un robot spatial typique

5.4.2 Modèle très rigide

Puis nous avons testé le contrôleur avec un modèle très rigide de robot spatial. Ces caractéristiques sont données au chapitre suivant. Initialement, le bras est en position horizontale avec un angle en début de barre de 10° . Après un ensemble de test essai-erreur, nous obtenons les paramètres les plus performants du contrôleur, donnés au tableau 5.2. On constate que Q_x est très petit, nous considérons donc que le modèle que nous avons créé donne de très bons estimés en position. Cependant Q_t est relativement grand. Ainsi on estime que le modèle ne calcule pas de façon exacte la déformation en temps, on lui accorde donc moins d'importance.

D'après les figures 5.4 et 5.5, nous constatons que le système converge en position linéaire en environ 19s et en position angulaire en environ 10s. Les dépassements sont de 7.5 degrés pour l'angle et de .8m pour la position. Ceci nous donne 7.5% de dépassement angulaire et 11.8% (le robot spatial spécifique totalisant une longueur de 6.8m) de dépassement linéaire. Il faudra donc faire attention lors de l'application de ce modèle, de laisser un espace libre suffisant autour du bras pour permettre un tel débattement.

Là aussi, nous avons constaté au cours de la simulation que le temps de calcul était extrêmement long. En effet, il nous a fallu 140s (soit 2min 20s) pour effectuer une

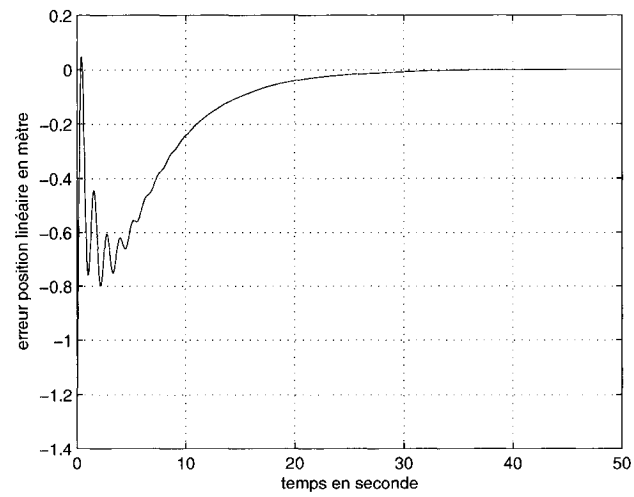


Figure 5.4 Erreur de position linéaire d'un robot spatial typique avec le troisième contrôleur

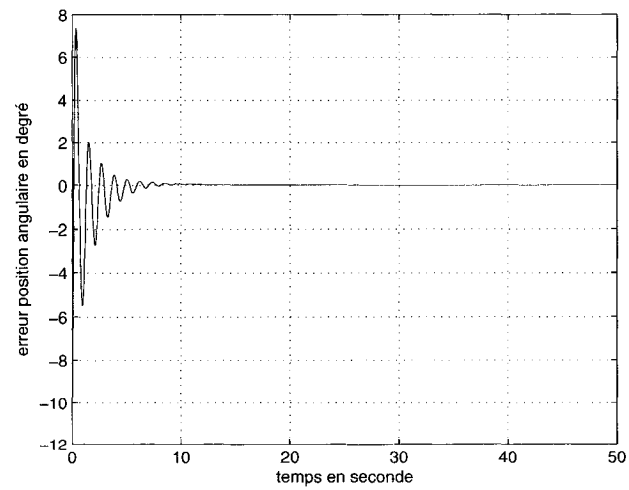


Figure 5.5 Erreur de position angulaire d'un robot spatial typique avec le troisième contrôleur

simulation de 50s. 1s de simulation nécessite donc bien 3s de calculs.

Remarque : Une animation du mouvement de ce type de barre (“robotspatialspe.avi”) est fournie sur le CD connexe. On constate que la barre rigide bouge en un seul bloc, lors des vingt premières secondes les plus sujettes aux oscillations. une autre animation (“robotspatialspeDETAIL.avi”) se concentre sur les cinq premières secondes.

5.5 Conclusion

La méthode de commande avec filtre de Kalman semble complexe à mettre en oeuvre.

Ce contrôleur est très complet, car il permet la commande de toutes sortes de barres, du très rigide au très flexible. Les performances qu’il dégage en convergence sont très bonnes.

Mais il a un inconvénient majeur, son temps de calcul est relativement plus élevé que les deux autres contrôleurs.

CHAPITRE 6

COMPARAISONS

Le but de cette partie est de mettre en évidence les avantages et les inconvénients de chacun des contrôleurs, que nous avons développés, par rapport aux autres.

6.1 Trois exemples

Afin de mieux comparer les différents modèles nous allons les simuler avec deux ou trois exemples identiques. On choisit trois types de barres différentes qui représentent le plus largement possible les rigidités de barre que l'on peut rencontrer.

Dans tous les exemples les paramètres des contrôleurs sont trouvés par un ensemble de test essai-erreur afin de choisir ceux qui entraînent la meilleure convergence du système.

6.1.1 Barre très flexible

Il s'agit d'une barre de rigidité $EI = 50N.m^2$, de longueur $L = 1m$ et de masse linéique $\rho = 2kg.m^{-1}$.

6.1.2 Antenne de satellite

Il s'agit d'une barre semi-rigide de longueur $L = 1.5m$, de masse linéique $\rho = .95kg.m^{-1}$ et de rigidité $EI = 500N.m^2$.

6.1.3 Robot spatial typique

Il s'agit d'une barre super rigide sur Terre de section circulaire de diamètre extérieur 335mm et d'épaisseur 6mm. La barre est faite dans un alliage de carbone-epoxy et elle a une longueur $L = 6.8m$.

D'après les données que nous avons trouvées sur Internet à propos des alliages graphite-epoxy, nous avons choisi un matériau de module de Young $E = 115GPa$ et de masse volumique $\rho = 1.8g.cm^{-3}$.

A l'aide de l'équation du moment d'inertie pour un cylindre :

$$I = \frac{\pi r^4}{4}$$

nous obtenons un moment d'inertie $I = 9.3935e^{-5}m^{-4}$. Ce qui nous donne une rigidité $EI = 1.08e^7N.m^2$.

A l'aide de la section de la barre nous pouvons aussi trouver la masse linéique qui est $\rho = 11.8kg.m^{-1}$.

6.2 Comparaisons sur des simulations idéales

Le critère utilisé pour évaluer le temps de convergence est le temps de réponse à 5%, c'est-à-dire le temps au bout duquel le système reste dans un cône de $\pm 5\%$ de la valeur finale.

Dans toutes les simulations qui suivent, on suppose qu'initialement la barre est en position horizontale, avec un angle en début de poutre de 10° . On prend un profil d'accélération sinusoïdale. Ceci diminue les chances d'atteindre la position

	Premier modèle	Deuxième modèle	Troisième modèle
λ_1	0.1	0.5	0.2
K_{v1}	10^{-7}	10^{-7}	10^{-5}
K_{i1}	0	10^{-6}	10^{-5}
λ_2	0.01	0.5	100
K_{v2}	10^{-9}	10^{-9}	10^{-9}
K_{i2}	10^{-5}	10^{-5}	0
K_F	N/A	$2 \cdot 10^{-5}$	$2 \cdot 10^{-5}$
K_T	N/A	10^{-4}	10^{-4}
Q_x	N/A	N/A	10^{-9}
Q_t	N/A	N/A	10^{-3}

Tableau 6.1 Paramètres des contrôleurs pour le robot spatial typique

singulière.

6.2.1 Robot spatial typique

Les paramètres des trois contrôleurs qui permettent la stabilisation de ce système sont donnés dans le tableau 6.1. On constate que le deuxième modèle a la plus grande valeur de λ_1 . Il est donc fort probable que ce modèle converge le plus rapidement en position linéaire.

On remarque aussi que pour le troisième modèle λ_2 est très grand. Cela risque d'engendrer beaucoup d'oscillations sur la position angulaire.

Les figures 6.1, 6.2 et 6.3 nous montrent les résultats de la simulation du système avec ces trois contrôleurs. On constate que le deuxième contrôleur converge beaucoup plus rapidement en position linéaire que les deux autres. On constate également de nombreuses oscillations sur l'erreur en position angulaire pour le troisième modèle. Les deux remarques sont conformes avec ce qu'on avait observé avec les paramètres des contrôleurs.

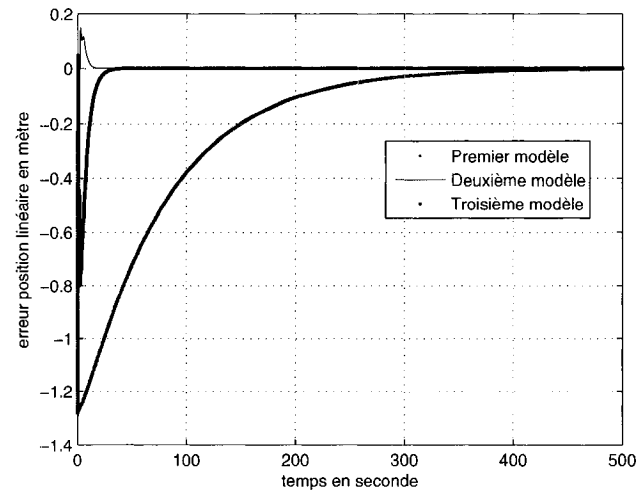


Figure 6.1 Erreur de position linéaire du robot spatial typique avec les trois contrôleurs

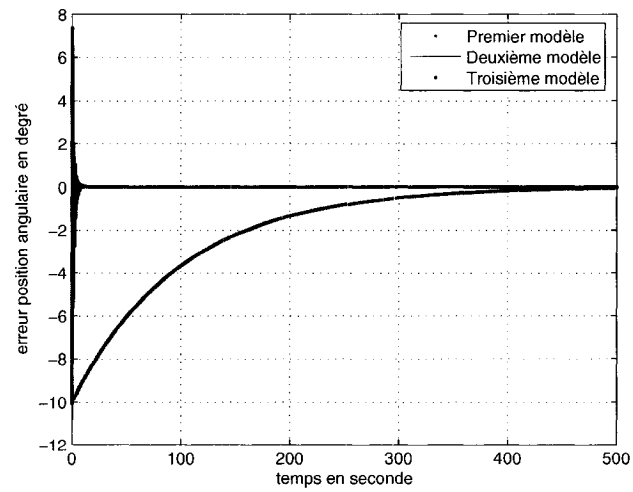


Figure 6.2 Erreur de position angulaire du robot spatial typique avec les trois contrôleurs

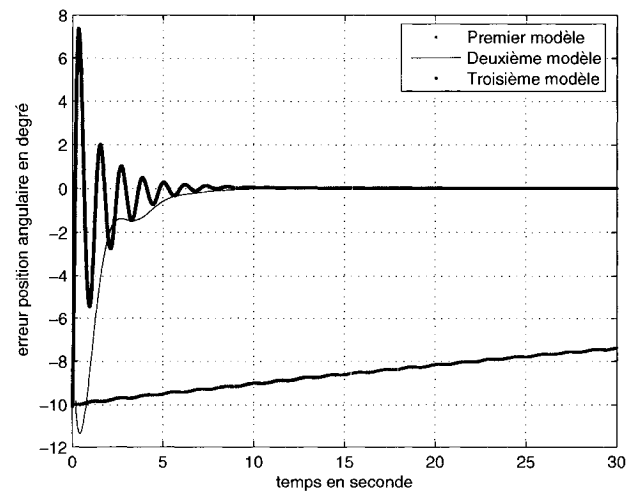


Figure 6.3 Détails de l'erreur de position angulaire du robot spatial typique avec les trois contrôleurs

	Deuxième modèle	Troisième modèle
temps de convergence en position	8.03s	18.98s
temps de convergence en angle	9.52s	9.87s
dépassement en position	0.15	0.8
dépassement en angle	0	7.5
temps de simulation pour 50s	33s	140s

Tableau 6.2 Performances des deux derniers contrôleurs pour le robot spatial typique

Ces figures nous montrent que le premier modèle que nous avons développé est le moins performant, comme on pouvait se douter. En ce qui concerne les performances du deuxième et du troisième modèle, elles sont résumées dans le tableau 6.2. On constate que le deuxième modèle converge plus rapidement que le troisième et avec beaucoup moins de dépassement et d'oscillations (vu sur les figures). De plus le temps de calcul du deuxième modèle est 5 fois moins élevé que celui du dernier modèle. Ceci est une caractéristique très intéressante.

Ainsi pour un système très rigide le contrôleur le plus performant est le deuxième modèle que nous avons développé. Le troisième modèle a également de bonnes performances, mais un inconvénient majeur, un temps de calcul très élevé.

6.2.2 Antenne typique

Remarque : d'après ce qui a été dit précédemment à propos du premier contrôleur, celui-ci ne pourra pas commander une antenne de satellite. En effet pour une barre de 1m et de rigidité $EI = 510N.m^2$, ce contrôleur atteint ces limites. Donc puisqu'on augmente la longueur et qu'on diminue la rigidité, l'antenne se situe au delà de la limite de rigidité admissible du premier contrôleur. Il ne pourra pas commander ce système. Cet exemple ne sera donc pas pertinent pour comparer ce contrôleur avec les deux autres.

Les paramètres des deux derniers modèles de contrôleurs qui stabilisent le système antenne sont donnés au tableau 6.3. D'après les valeurs de λ_2 , K_{v2} et K_{i2} il semblerait que le troisième modèle converge plus vite en position angulaire. On constate également que les valeurs de Q sont très petites, ce qui signifie qu'on fait extrêmement confiance à notre modèle lors du calcul des estimées.

Les figures 6.4 et 6.5 nous montrent les résultats de la simulation de l'antenne avec

	Deuxième modèle	Troisième modèle
λ_1	0.1	0.03
K_{v1}	10^{-5}	10^{-3}
K_{i1}	10^{-3}	10^{-5}
λ_2	0.1	0.1
K_{v2}	10^{-8}	10^{-7}
K_{i2}	10^{-5}	10^{-3}
K_F	$2 \cdot 10^{-3}$	0.1
K_T	10^{-3}	10^{-4}
Q_x	N/A	10^{-9}
Q_t	N/A	10^{-9}

Tableau 6.3 Paramètres des deux derniers contrôleurs pour l'antenne

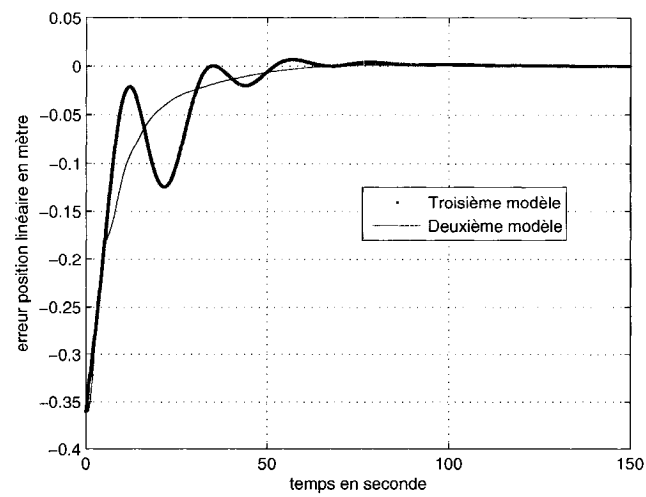


Figure 6.4 Erreur de position linéaire de l'antenne avec les deux derniers modèles

	Deuxième modèle	Troisième modèle
temps de convergence en position	28.3s	28.5s
temps de convergence en angle	146.38s	55.88s
dépassement en position	0	≈ 0
dépassement en angle	1.25	0
temps de simulation pour 150s	114s	420s

Tableau 6.4 Performances des différentes commandes sur le modèle antenne

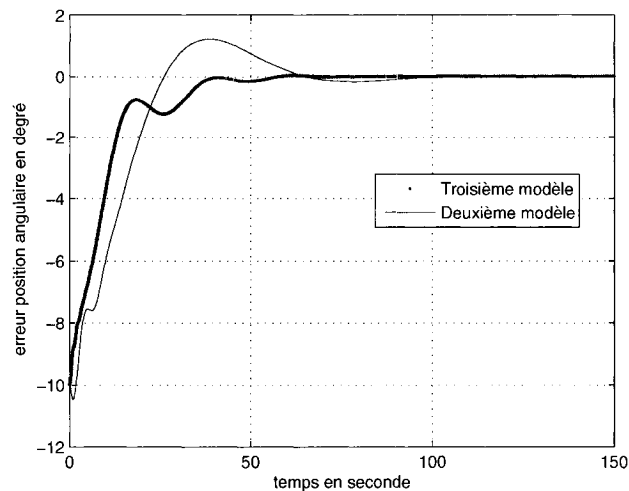


Figure 6.5 Erreur de position angulaire de l'antenne avec les deux derniers modèles

ces deux contrôleurs. On constate bien que l'erreur en position angulaire converge plus vite dans le cas du troisième contrôleur. Ces figures nous permettent d'établir le tableau 6.4 des performances des deux contrôleurs. On constate que dans ce cas, c'est le troisième contrôleur qui est le plus performant. En effet ces valeurs de dépassement et de temps de convergence sont plus petites ou équivalentes à celles du deuxième modèle.

Cependant, ce dernier modèle nécessite toujours environ 3s de calcul pour simuler 1s, contrairement au deuxième modèle qui est quasiment en 1s de calcul pour 1s de simulation.

Le modèle de l'antenne étant plus flexible que le robot spatial typique, les performances du deuxième contrôleur commencent à se dégrader alors que celle du troisième contrôleur s'améliore en comparaison. Cependant les différences ne sont pas encore flagrantes et les deux contrôleurs offrent de bonnes performances.

	Deuxième modèle	Troisième modèle
λ_1	0.01	0.01
K_{v1}	10^{-8}	10^{-3}
K_{i1}	10^{-5}	10^{-5}
λ_2	0.03	0.01
K_{v2}	10^{-10}	10^{-5}
K_{i2}	10^{-6}	10^{-9}
K_F	2.10^{-4}	10^{-3}
K_T	10^{-4}	10^{-3}
Q_x	N/A	10^{-3}
Q_t	N/A	10^{-3}

Tableau 6.5 Paramètres des deux derniers contrôleurs pour la barre très flexible

	Deuxième modèle	Troisième modèle
temps de convergence en position	170s	80s
temps de convergence en angle	535s	600s
dépassement en position	0	0.02
dépassement en angle	0	0
temps de simulation pour 1000s	969s	3319s

Tableau 6.6 Caractéristiques des deux derniers contrôleurs pour la barre très flexible

6.2.3 Barre très flexible

Remarque : Cette barre est au-delà de la limite de rigidité contrôlable par le premier modèle. Cet exemple ne sera pas non plus pertinent pour comparer ce modèle avec les deux autres.

Les paramètres des deux derniers modèles de contrôleurs qui stabilisent la barre très flexible sont donnés au tableau 6.5. D'après les valeurs de λ_1 , K_{v1} et K_{i1} il semblerait que le troisième modèle converge plus vite en position linéaire, puisque ses valeurs sont plus grandes ou égales. A priori, il en va de même pour la position angulaire. On constate également que les valeurs de Q sont relativement grandes, ce qui signifie qu'on ne fait pas aveuglément confiance à notre modèle lors du calcul des estimées.

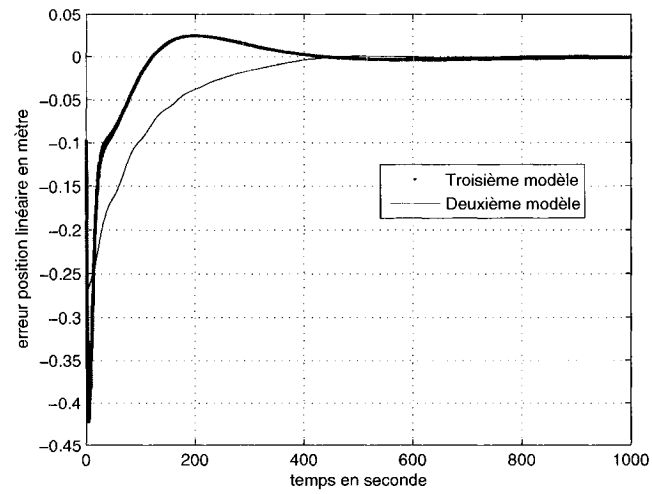


Figure 6.6 Erreur de position linéaire de la barre très flexible

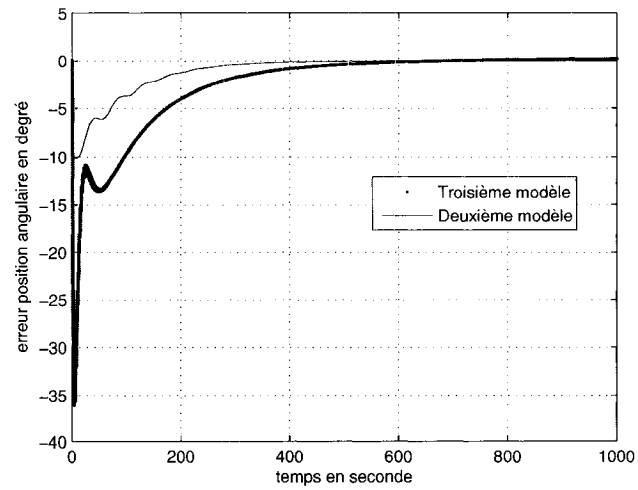


Figure 6.7 Erreur de position angulaire de la barre très flexible

Les figures 6.6 et 6.7 nous montrent les résultats de la simulation de la barre très flexible avec ces deux contrôleurs. Afin de mieux voir les résultats, nous avons établi le tableau 6.6 des performances des deux contrôleurs. On constate bien que l'erreur en position linéaire converge plus vite pour le troisième contrôleur que pour le deuxième. Cependant contrairement à ce qu'on pensait, la convergence en position angulaire est légèrement plus petite avec le troisième contrôleur. Mais elle reste tout de même proche de celle du deuxième contrôleur. Les dépassement des deux contrôleurs sont quasiment équivalents.

Comme lors des autres exemples, on remarque que le troisième contrôleur a un temps de calcul beaucoup plus élevé que le deuxième. Ainsi même s'il présente de bonnes performances, celles-ci sont diminuées par ce temps de calcul important.

Là encore le troisième contrôleur est le plus performant, dans le sens où sa convergence en position linéaire est deux fois plus rapide que celle du deuxième contrôleur et que les convergences en position angulaire sont quasi identiques. Mais il possède toujours l'inconvénient majeur d'avoir un temps de calcul de l'ordre de 3s de calcul pour 1s de simulation.

6.3 Comparaisons sur la simulation d'une situation réelle

Dans un soucis de simuler plus exactement ce qui se produirait dans le cas d'une expérimentation, nous avons modifié les variables que nous pouvions mesurer. Ainsi seules la position de déformation et la position angulaire de la fin de la poutre seront prises en compte. Il faudra ensuite reconstituer la vitesse de déformation et la vitesse angulaire.

Nous avons en plus ajouter un bruit sur les positions mesurées pour représenter les bruits induits par un capteur.

6.3.1 Reconstitution de la vitesse

Pour commencer nous avons uniquement considéré la position sans bruit. Nous avons alors construit la vitesse à partir de la formule suivante, issue du développement en série de Taylor tronqué au premier ordre :

$$\frac{\partial \omega(N, t)}{\partial t} = \frac{\omega(N, t) - \omega(N, t - \Delta T)}{\Delta T}$$

Pour le premier temps de calcul nous avons choisi de sortir une vitesse nulle. Ceci évite la présence d'une trop grande erreur sur le calcul de la vitesse initialement. Nous diminuons ainsi les chances de divergence du système.

6.3.1.1 Premier code du contrôleur

Pour le premier contrôleur nous avons effectué la modification sur le modèle du bras de rigidité $EI = 10000N.m^2$. Les résultats de la simulation de ce système avec et sans reconstitution de la vitesse sont exactement les mêmes. Les courbes étant identiques, nous n'avons pas trouvé pertinent de les mettre dans le rapport.

Le contrôleur n'est donc pas affecté par la modification, ces performances sont les mêmes que précédemment.

Remarque : Nous avons également vérifié avec le robot spatial typique, les résultats sont là aussi identiques.

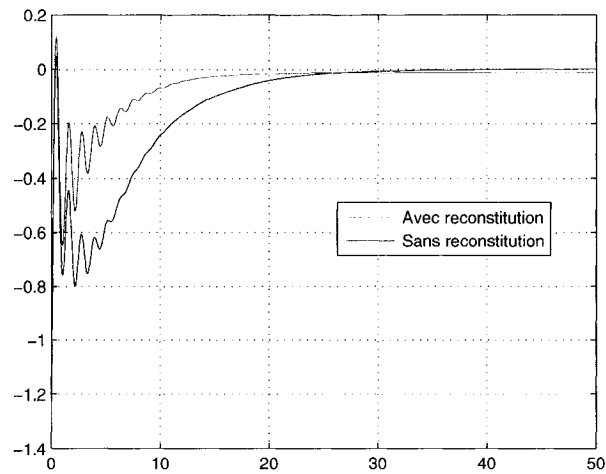


Figure 6.8 Erreur de position linéaire du robot spatial typique avec reconstitution de la vitesse et le troisième code du contrôleur

6.3.1.2 Deuxième code du contrôleur

Avec le deuxième contrôleur développé, nous n'avons pu obtenir de résultats probants lors de simulation avec reconstitution de la mesure, quelque soit le modèle utilisé. Dans tous nos essais, le système s'est mis à diverger rapidement.

Nous en avons donc déduit que ce contrôleur n'est pas robuste à une modification de la mesure de vitesse.

6.3.1.3 Troisième code du contrôleur

Nous avons d'abord effectué les modifications sur le modèle du robot spatial typique. Les résultats de la simulation sont données aux figures 6.8 et 6.9. Nous constatons que les performances du contrôleur sont peu affectées par un tel changement. Le contrôleur serait presque plus performant dans ce cas ! En effet la position linéaire converge en 12s environ au lieu de 19s, par contre le dépassement est plus conséquent, 12% au lieu de 8%. Pour la position angulaire, les performances sont

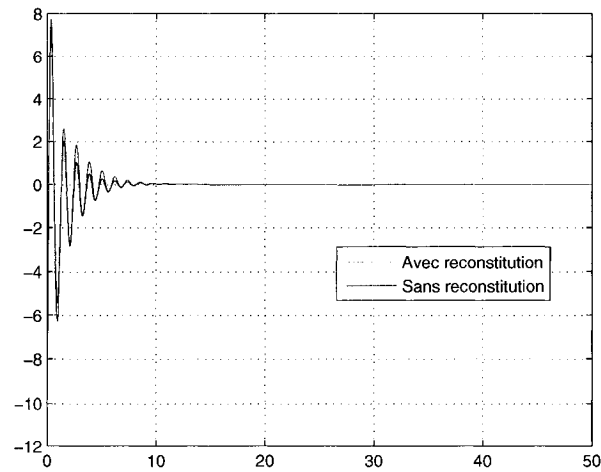


Figure 6.9 Erreur de position angulaire du robot spatial typique avec reconstitution de la vitesse et le troisième code du contrôleur

quasiment les mêmes.

Remarque : Les résultats obtenus par simulation sur une barre très flexible ($EI = 1N.m^2$) et sur l'antenne, sont identiques à ceux obtenus sans reconstitution de la vitesse.

Le contrôleur n'est donc pas affecté par un changement sur la vitesse.

6.3.2 Ajout de bruit sur la position

Nous avons ensuite bruité la mesure de position afin de se rapprocher d'un capteur réel. Il s'agit d'un bruit aléatoire de puissance 10^{-8} . Donc on ajoute à la valeur du signal ± 0.001 unités.

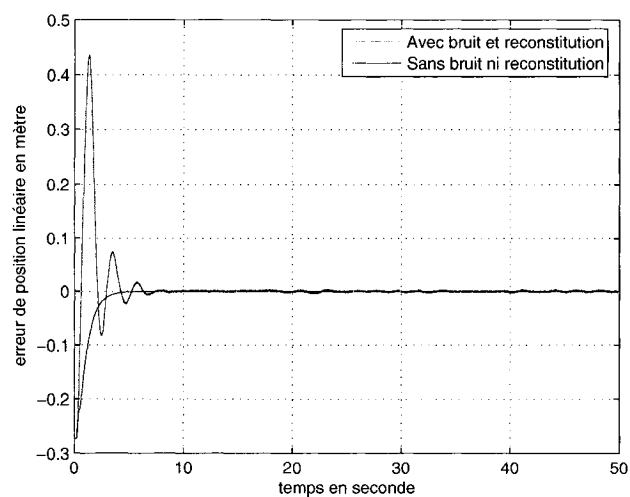


Figure 6.10 Erreur de position linéaire d'un bras très rigide avec bruitage de la mesure et le premier code du contrôleur

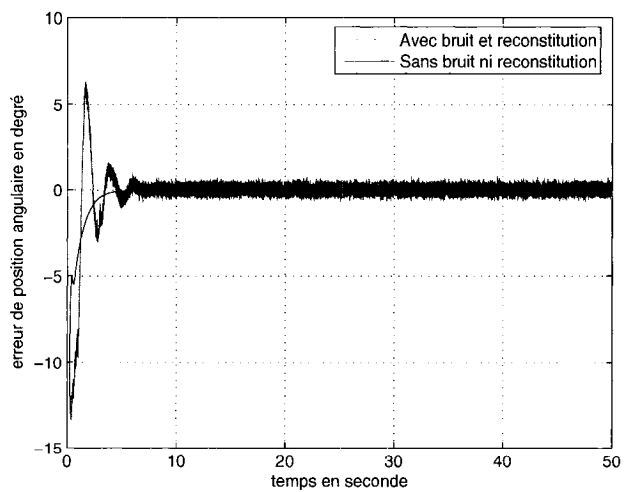


Figure 6.11 Erreur de position angulaire d'un bras très rigide avec bruitage de la mesure et le premier code du contrôleur

6.3.2.1 Premier code du contrôleur

Les figures 6.10 et 6.11 donnent les résultats pour la simulation d'un bras de rigidité $EI = 10000N.m^2$. On constate que le contrôleur est légèrement moins performant qu'avec des mesures parfaites. Il converge en position linéaire en 3.8s, au lieu de 2s, avec un dépassement de 44% de la valeur finale au lieu d'un dépassement nul. Pour la position angulaire, le dépassement est de 6° , au lieu d'un dépassement nul, et la mesure ne rentre jamais dans le cône à 5%. En effet la position angulaire oscille sur $\pm 0.5^\circ$. Cette oscillation est minime, mais existera toujours, comme un parasite.

En regardant ces résultats, on peut en déduire que ce contrôleur est robuste au bruitage sur la mesure.

Cependant, nous n'avons pu mettre plus de bruit sur la commande. En effet pour un bruit de puissance 10^{-7} , le système diverge rapidement. Il faudra donc prévoir des capteurs avec une très grande précision, de l'ordre de $5^\circ/\infty$.

6.3.2.2 Deuxième code du contrôleur

Avec le deuxième contrôleur, nous n'avons réussi qu'à faire diverger les systèmes dans le cas d'un bruitage de la position mesurée en plus de la reconstitution de la vitesse. Ceci n'est pas étonnant, puisqu'une simple reconstitution de la vitesse entraînait la divergence du système.

Ce contrôleur n'est donc pas robuste du tout à une modification sur la mesure de position.

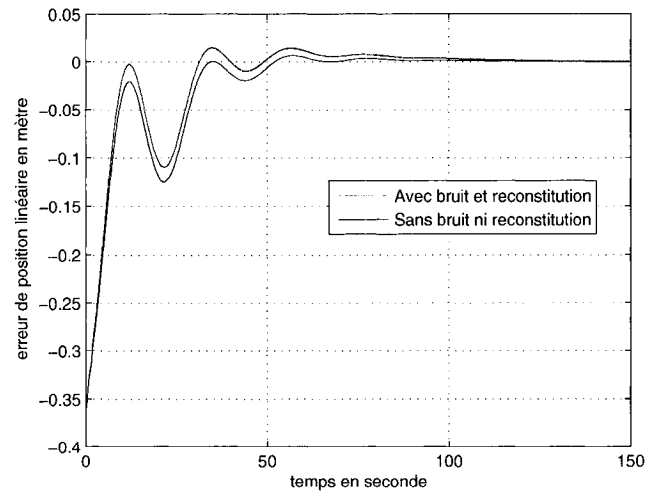


Figure 6.12 Erreur de position linéaire de l'antenne avec bruitage de la mesure et le troisième code du contrôleur

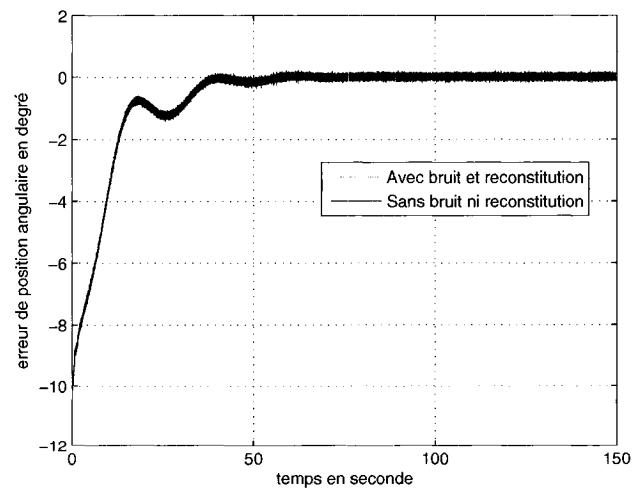


Figure 6.13 Erreur de position angulaire de l'antenne avec bruitage de la mesure et le troisième code du contrôleur

6.3.2.3 Troisième code du contrôleur

Lors du test d'une mesure bruitée de la position sur le modèle du robot spatial typique, les résultats étaient les mêmes que ceux obtenus lors de la reconstitution de la commande seule.

De même avec la barre très flexible, les résultats ne différaient pas de ceux obtenus lors de la reconstitution de la commande seule.

Nous avons alors décidé de tester ce contrôleur avec des bruits de mesure sur le système antenne. Les résultats obtenus sont donnés aux figures 6.12 et 6.13. Nous constatons que les performances du contrôleur sont assez équivalentes à celles obtenues sans bruitage de la mesure de déformation, ni reconstitution de la vitesse. Seule l'erreur en position angulaire est légèrement plus bruitée, la courbe oscille entre $\pm 0.2^\circ$, ce qui est somme toute très peu.

On peut donc déduire que ce contrôleur est robuste aux bruits de mesures. Dans le cas de ce contrôleur, le filtrage de Kalman réduit les erreurs dues aux arrondis numériques et également celles dues aux bruits de mesure. C'est pourquoi un léger bruit ne perturbe pas le système.

Remarque : Tout comme le premier contrôleur, celui-ci ne peut stabiliser le système, si la puissance du bruit atteint 10^{-7} . Il faudra donc là aussi, un capteur très précis.

6.4 Conclusion

Le premier modèle est le moins performant des trois. Il ne peut pas commander de barres semi-rigides à flexibles. Et ses performances dans le cas de commande de barres rigides sont moins bonnes que les deux autres modèles. On laissera donc

ce modèle de côté bien que ces performances en simulation réelle soient bien plus intéressantes que le deuxième contrôleur.

Ensuite en regardant les temps de convergence de nos trois exemples, on constate que le troisième modèle du contrôleur est de plus en plus efficace par rapport aux autres modèle de contrôleurs, plus la barre devient flexible. En effet plus la barre est flexible plus ses temps de convergence en position sont petits par rapport à ceux des autres contrôleurs.

De la même façon on constate que plus la barre est rigide plus les performances du deuxième modèle (et a fortiori du premier modèle) s'améliorent relativement au troisième modèle.

Nous avons donc deux types de contrôleurs qu'il s'agira de choisir en fonction du type de barre à commander. Pour une barre rigide, il sera préférable d'utiliser le deuxième modèle. Alors que pour une barre flexible, il sera préférable d'utiliser le troisième modèle. Pour une barre semi-rigide, on pourra opter pour l'un ou l'autre des deux derniers modèles.

Cependant pour une application temps réel, seuls le premier et le dernier contrôleur que nous avons développés sont intéressants. En effet se sont les seuls à résister relativement bien aux bruits de mesure. Ils nécessitent tout de même un capteur extrêmement précis, ce qui n'est pas forcément évident à trouver. Il faudra donc penser à développer un nouveau contrôleur plus robuste aux bruits de mesure. Ou alors revoir le dernier contrôleur, le seul intéressant pour une application étendue et en temps réel, pour diminuer ses temps de calculs.

CONCLUSION

Dans ce mémoire, nous avons développé une méthode de contrôle pour une barre flexible libre de tous mouvements plans.

Après une revue de littérature dans laquelle nous avons décrit une liste non exhaustive des contrôleurs qui avaient été développés jusqu'à maintenant, nous avons décidé de développer notre contrôleur dans le cadre de la décomposition virtuelle. Ceci nous a donné deux choix pour les lois de commande en force et en moment.

Après avoir testé les deux lois de commande, nous en avons déduit que celle prenant en compte les dérivées temporelles de la déformation et de l'angle était plus efficace. Cependant aucune des deux ne remplissaient le but de la recherche qui était de contrôler des bras extrêmement flexibles. C'est pourquoi nous avons également développé un contrôleur avec la méthode des filtres de Kalman. Bien qu'efficace pour le contrôle de barres très flexibles, ce contrôleur présente l'inconvénient d'un temps de calcul très grand.

Une suite à cette étude serait une optimisation de la troisième loi de commande, afin de réduire son temps de calcul. L'optimisation devrait avoir lieu au niveau de la boucle de propagation. Une fois ceci effectué, l'application temps réel du dernier modèle développé serait intéressant. Enfin une application mettant en jeu deux barres flexibles physiquement connectées permettrait d'illustrer le principe de décomposition virtuelle.

RÉFÉRENCES

- (Zhu et al., 1997) W.H. Zhu, Y.G. Xi, Z.J. Zhang, Z. Bien et J. De Schutter, "Virtual decomposition based control for generalized High Dimensional Robotic systems with complicated structure", *IEEE Transactions on Robotics and Automation*, vol.13 no.3, Juin 1997.
- (Simo et Vu-Quoc, 1986) J.C Simo et L. Vu-Quoc, "On the dynamics of flexible beams under large overall motions-the plane case", *Journal of Applied Mechanics*, vol.53, 849-863, Décembre 1986.
- (Yang et al., 2001) T.W. Yang, W.L. Xu et S.K. Tso, "Dynamic modeling based on real-time deflection measurement and compensation control for flexible multi-link manipulators", *Dynamics and Control* 11, 5-24, 2001.
- (Dubowsky et Papadopoulos, 1993) S. Dubowsky et E. Papadopoulos, "The kinematics, dynamics, and control of free-flying and free-floating space robotic systems", *IEEE Transactions on Robotics and Automation* vol.9 no.5, Octobre 1993.
- (Queiroz et al., 1999) M.S. de Queiroz, D.W. Dawson, M. Agarwal et F. Zhang, "Adaptative nonlinear boundary control of a flexible link robot arm", *IEEE Transactions on Robotics and Automation* vol.15 no.4 Août 1999.
- (Bouaziz et al., 1993) B. Bouaziz, D. Dochain, J.C. Piedboeuf et R. Hurteau, "Singular perturbation approach to modeling and control of flexible manipulators", *Second IEEE Conference on Control Applications* Septembre 13-16 1993, Vancouver (B.C.).
- (Kane et al., 1987) T.R. Kane, R.R. Ryan et A.K. Banerjee, "Dynamics of a cantilever beam attached to a moving base", *Journal of Guidance* vol.10 no.2 Mars-Avril 1987.

(Morgül, 1991) O. Morgül, "Orientation and stabilization of a flexible beam attached to a rigid body : planar motion", *IEEE Transactions on Automatic Control* vol.36 no.8 Août 1991.

(Morgül, 1992) O. Morgül, "Dynamic boundary control of a Euler-Bernoulli beam", *IEEE Transactions on Automatic Control* vol.37 no.5 Mai 1992.

(Martins et al., 2002) J. Martins, M.A. Botto et J. sa da Costa, "Modeling of felxible beams for robotic manipulators", *Multibody System Dynamics* 7: 79-100, 2002, *Kluwer Academic Publishers*.

(Sun et Mills, 1998) D. Sun et J.K. Mills, "Combined PD feedback and distributed piezoelectric-polymer vibration control of a single-link flexible manipulator", *Conference on Intelligent Robots and Systems* Octobre 1998, Victoria (B.C.).

(Wen et Kreutz, 1991) J.T.Y Wen et K. Kreutz-Delgado, "The attitude control problem", *IEEE Transactions on Automatic Control* vol.36 no.10 Octobre 1991.

(Rhim et al., 2001) S. Rhim, A.P. Hu, N. Sadegh, W.J. Book, "Combining a multirate repetitive learning controller with command shaping for improved flexible manipulator control", *Journal of Dynamic Systems, Measurement and Control* vol.123 Septembre 2001.

(Jonkers et Aarts, 2001) J.B. Jonkers et R.G.K.M. Aarts, "A perturbation method for dynamic analysis and simulation of flexible manipulators", *Multibody System Dynamics* 6:245-266, 2001, *Kluwer Academic Publishers*.

(Slotine et Li, 1987) J.J.E. Slotine et W. Li, "On the adaptive control of robot manipulators", *The International Journal of Robotics Research* vol.6 no.4 Automne 1987.

(Slotine et Li, 1988) J.J.E Slotine et W. Li, "Adaptive manipulator control : a case study", *IEEE Transactions on Automatic Control* vol.33 no.11 Novembre 1988.

(Sadegh et Horowitz, 1990) N. Sadegh et R. Horowitz, "Stability and robustness analysis of a class of adaptative controllers for robotic manipulators", *The International Journal of Robotics Research* vol.9 no.3 Juin 1990.

(Walker, 1990) M.W. Walker, "Adaptative control of manipulator containing closed kinematic loops", *IEEE Transactions on Robotics and Automation* vol.6 no.1 Février 1990.

(Garcia et Feliu, 2000) A. Garcia et V. Feliu, "Force control of a single-link flexible robot based on a collision detection mechanism", *IEE Proc-Control Theory Application* vol.147 no.6 Novembre 2000.

(Senda et Murotsu, 2000) K. Senda et Y. Murotsu, "Methodology for control of a space robot with flexible links", *IEE Proc-Control Theory Application* vol.147 no.6, Novembre 2000.

(Sun, 2002) Q. Sun, "Control of flexible link multiple manipulators", *Journal of Dynamic Systems, Measurement and Control* vol.124 Mars 2002.

(Meirovitch, 1997) L. Meirovitch, "Principles and Techniques of vibrations", *Upper Saddle River, N.J. : Prentice Hall* 1997.

(Siouris, 1996) G.M. Siouris, "An engineering approach to optimal control and estimation theory", *New York ; Toronto : John Wiley and Sons, inc.* 1996.

**Un cédérom accompagne ce
mémoire de maîtrise**

**Toute personne intéressée à se le
procurer doit contacter :**

**An accompanying CD-ROM is
included with the original thesis.**

**To obtain a copy of the CD-ROM,
please contact :**

École Polytechnique de Montréal

Service du prêt entre bibliothèques

B.P. 6079, Succursale Centre-Ville

Canada