Network-based Haptic Systems with Time-Delays

Bogdan Cristian Liacu

N° d'ordre : 2012-22-TH

THÈSE DE DOCTORAT

DOMAINE : STIC

SPECIALITE : Automatique

Ecole Doctorale « Sciences et Technologies de l'Information des Télécommunications et des Systèmes »

Présentée par :

Bogdan-Cristian LIACU

Sujet :

Systèmes haptiques en réseau avec retards de communication

(Network-based Haptic Systems with Time-Delays)

Soutenue le 20 Novembre 2012 devant les membres du jury :

M. Claude ANDRIOT	CEA-LIST, Fontenay-aux-Roses	Co-encadrant, invité
M. Patrick BOUCHER	SUPELEC, Gif sur Yvette	Co-encadrant, invité
M. Frédéric COLLEDANI	CEA-LIST, Fontenay-aux-Roses	Co-encadrant
M. Gilles DUC	SUPELEC, Gif sur Yvette	Examinateur
M. Didier DUMUR	SUPELEC, Gif sur Yvette	Co-encadrant
Mme Sabine MONDIÉ	Cinvestav-IPN, México	Examinatrice
M. Silviu-Iulian NICULESCU	LSS, Gif sur Yvette	Directeur de thèse
M. Sorin OLARU	SUPELEC, Gif sur Yvette	Co-encadrant, invité
M. Hitay ÖZBAY	Bilkent University, Turquie	Invité
M. Vladimir RĂSVAN	Universitatea din Craiova, Romania	Rapporteur
M. Jean-Pierre RICHARD	Ecole Centrale de Lille	Rapporteur
Mme Sophie TARBOURIECH	LAAS, Toulouse	Examinatrice

Acknowledgments

First, I would like to express my sincere appreciation and to thank Professor Silviu-Iulian Niculescu, for his guidance, encouragement and constant support throughout my thesis.

I would also like to thank Professor Patrick Boucher who encouraged me to start the thesis and who offered me realistic feedbacks and support for continuing my thesis. Special thanks go to Professor Didier Dumur for his guidance and support in my thesis work. I thank also Professor Sorin Olaru for his valuable comments and advices regarding my work.

My thesis was made in collaboration with CEA LIST, and in this sense, I would like to express my gratitude and to thank my supervisors Dr. Frédéric Colledani and Dr. Claude Andriot for their confidence and support. Special thanks go to Dr. Florian Gosselin for his constant support concerning my experiments as well as for his valuable advices from a mechanical point of view. I thank also Dr. Alain Micaelli, Dr. Frank Geffard and Dr. Sylvain Bouchigny.

I would like to express my gratitude to Professor Jean-Pierre Richard and Professor Vladimir Răsvan for accepting to review my thesis and for their insightful comments that helped me to improve my manuscript. Special thanks to Professor Gill Duc, Professor Sophie Tarbouriech and Professor Sabine Mondié for accepting to be part of my thesis jury.

I would like to express my sincere appreciation and to thank Professor Hitay Özbay for his warm welcome at Ankara and for our pleasant collaboration. I thank Professor Joono Cheong (Korea University) for valuable discussions and suggestions.

I am grateful to my colleagues from Supélec, L2S and CEA LIST who made my student life memorable and joyful.

Special thanks are addressed to my parents and my family for their endless support in all aspects of my life.

I save my last thank for my wife Raluca, who conferred me the support I needed to acquire all my achievement and without which I could not be the person that I am today.

Bogdan Liacu

Contents

Résumé **1**
Introduction Générale . . . 1
Chapitre 1 . . . 5
Chapitre 2 . . . 10
Chapitre 3 . . . 18
Chapitre 4 . . . 23
Conclusions et perspectives . . . 30

Abbreviations and acronyms **33**

General Introduction **35**
General Context. Teleoperation & Haptic Systems . . . 35
Motivation and Objectives . . . 40
Thesis Plan . . . 42
List of publications related to the PhD . . . 45

1 Methods and control algorithms used in haptics and teleoperation. A state of the art **47**
1.1 Introduction . . . 47
1.2 Haptic & teleoperation systems' schemes . . . 48
1.3 Performance Criteria . . . 50
1.4 Commonly used algorithms (state of the art) . . . 54
1.4.1 Classic Proportional-Derivative (PD) . . . 55
1.4.2 Proportional-Derivative (PD) control with local dissipation . . . 55
1.4.3 Proportional-Derivative (PD) control with passivity observer . . . 56
1.4.4 Proportional-Derivative (PD) control with Passive Set-Position Modulation . . . 59
1.4.5 Wave-Scattering Transform . . . 60
1.4.6 Smith Predictor . . . 62
1.5 Experimental comparative study of the most common algorithms . . . 64
1.5.1 Presentation of the 1-DOF haptic experimental platform . . . 64
1.5.2 Results' Analysis . . . 65
1.5.2.1 Comparison Criteria . . . 65
1.5.2.2 Results . . . 66
1.6 Conclusions . . . 74

2 Methods of Analysis **75**
2.1 Introduction . . . 75

2.2 PID control for time-delay systems. An overview. . . . 77
2.3 Stability of Proportional Derivative (PD) controllers used in haptics under time-delays . . . 81
2.3.1 Constant time-delays . . . 82
2.3.2 Fragility of PD controllers . . . 86
2.3.2.1 Preliminaries . . . 87
2.3.2.2 Direction of Crossing . . . 89
2.3.2.3 Fragility of PD controllers . . . 90
2.3.2.4 Some Illustrative Examples . . . 93
2.3.3 Distributed time-delays . . . 96
2.3.3.1 Uncertain delays: uniform distribution . . . 98
2.3.3.2 Uncertain delays: gamma distribution with gap . . . 100
2.3.3.3 Illustrative examples . . . 102
2.4 Analysis of Smith predictor-based controllers used in haptics . . . 103
2.4.1 Preliminaries . . . 103
2.4.2 Stability analysis . . . 105
2.4.2.1 Stability regions . . . 106
2.4.3 Stability in delay parameters-space . . . 107
2.4.4 Fragility of Smith predictors . . . 110
2.4.5 Fixed and known delays . . . 111
2.4.6 Uncertain Delays . . . 111
2.4.6.1 Uncertain delays: constant uncertainty . . . 111
2.4.6.2 Uncertain delays: uniform distribution . . . 114
2.4.6.3 Uncertain delays: gamma distribution with gap . . . 115
2.4.6.4 Uncertain delays: normal distribution . . . 116
2.5 Tuning and Physical Limitations . . . 119
2.6 Conclusions . . . 121

3 Smith Predictor with distance feedback 123
3.1 Problem analysis . . . 123
3.2 Proposed solution . . . 125
3.3 3-DOF experimental platform . . . 129
3.4 Results and analysis . . . 130
3.4.1 Constant time-delays . . . 131
3.4.2 Uncertain time-delays: constant uncertainty . . . 137
3.4.3 Uncertain time-delays: uniform distribution . . . 145
3.4.4 Uncertain time-delays: gamma distribution with gap . . . 151
3.4.5 Uncertain time-delays: normal distribution . . . 157
3.4.6 Discussions . . . 162
3.5 Conclusions . . . 165

4 PD with gain scheduling depending on the distance 167
4.1 Problem analysis . . . 167
4.2 Proposed solution . . . 172
4.2.1 Proposed algorithm . . . 172
4.2.2 Discussions . . . 174
4.3 Results and analysis . . . 179

4.3.1 Constant time-delays . . . 180
4.3.2 Uncertain time-delays: uniform distribution . . . 182
4.3.3 Uncertain time-delays: gamma with gap distribution . . . 185
4.3.4 Discussions . . . 187
4.4 Conclusions . . . 188

Conclusions and Perspectives **191**

A Results' Analysis for 10ms time-delay **195**

B Stability and transparency for teleoperated systems **201**
B.1 Preliminaries . . . 201
B.2 Scalar gains . . . 206
B.3 Illustrative Examples . . . 209

C Reduction of the stability conditions **215**

D PD-like Controller **221**

E Fragility analysis of PD controllers **223**

Bibliography **223**

Résumé

Introduction Générale

Pendant les dernières décennies, de nombreuses technologies nouvelles ont été développées et introduites dans le milieu industriel. C'est en particulier le cas des systèmes de téléopération ainsi que des systèmes haptiques. Selon [19], un système de téléopération permet l'interaction humaine avec des environnements inaccessibles au contact direct pour l'homme, en raison de leur localisation, ou de circonstances dangereuses. De façon générale, l'objectif des systèmes de téléopération est de remplacer la manipulation directe par l'homme. En ce sens, ces systèmes sont utilisés pour travailler dans des environnements dangereux (comme la radioactivité ou la présence de poussière [88]), pour des opérations de haute précision (comme les opérations chirurgicales [182]), ainsi que pour des opérations spatiales qui sont faites à distance [171]. La téléchirurgie et la télérobotique dans l'espace sont deux exemples d'applications de téléopération plus récents concernant la communication sur une longue distance entre les robots maître et esclave [28, 61, 167]. L'intérêt récent pour la chirurgie assistée par des robots [60, 139] est lié aux nombreux avantages qu'elle apporte, comme des actions mini invasives [45], une meilleure précision et dextérité [31, 169], et une sécurité et une fiabilité augmentées [147, 181]. La téléchirurgie permet d'aller encore plus loin grâce à son potentiel fournissant un accès à des soins médicaux spécialisés pour un grand groupe de patients, plus efficacement et de manière rentable. Le retour haptique a démontré sa capacité à améliorer l'exécution des tâches pendant les applications de téléopération à la fois chirurgicale et dans l'espace [81, 187].

Selon [53], l'intérêt pour les systèmes téléopérés et les technologies en cours de développement pour la téléopération sont en augmentation. Les nouvelles technologies telles que le télétravail, télé-assistance ou e-learning ont un grand nombre d'objectifs communs qui favorisent le développement d'interfaces plus avancées et de nouvelles techniques de travail et de communication. Le développement essentiel en ce qui concerne l'interface de téléopération est concentré sur un retour visuel et haptique [2, 82]. D'autres points intéressants à considérer sont les reconnaissances vocales et des gestes qui établissent des dialogues [117, 148]. À la suite de cela, un

effort important est fait pour améliorer la communication entre l'opérateur et l'environnement, en particulier lors de l'utilisation des réseaux publics tels qu'Internet.

Globalement, un système de téléopération est composé de deux robots, disposés en des lieux géographiques différents, travaillant selon le principe du maître/esclave. En général, l'opérateur humain impose une force et/ou une position au robot maître qui transmet la commande au robot esclave par l'intermédiaire du réseau de communication. Selon les situations rencontrées par le robot esclave, le maître doit également recevoir des informations relatives à ces situations par le retour d'effort (une étude complète sur les systèmes de téléopération peut être trouvée dans [76]). Donc, l'idée de base d'un système de téléopération est de permettre à l'opérateur humain d'interagir avec des environnements distants en fournissant le sentiment de télé-présence [1].

Du point de vue de la commande, selon [186], le sujet est très difficile, car la boucle de commande (où les données de mouvement et la force sont échangées entre le manipulateur maître et l'esclave) est fermée via une communication réseau, par exemple Internet. Le réseau de communication introduit des problèmes de fiabilité, dus par exemple aux retards (variables) et aux pertes de paquets, qui non seulement dégradent la perception haptique humaine de l'environnement à distance, mais qui peuvent déstabiliser l'ensemble du système. Ces dernières années, les approches de commande basées sur le principe de la passivité et les variables d'onde ont été développées afin de stabiliser le système de téléopération en présence des incertitudes de communication, voir, par exemple, [5, 73].

Les systèmes haptiques représentent une *extension* des systèmes de téléopération. Un système haptique n'a que le robot maître, généralement appelé interface haptique. Le deuxième robot est remplacé par un robot/objet virtuel qui fonctionne dans un environnement virtuel. Le principe est le même, l'utilisateur doit ressentir les contacts rencontrés par l'objet/robot virtuel à travers le retour d'effort [177].

Les chercheurs qui travaillent dans le domaine de l'haptique cherchent à recréer le sentiment du *contact* d'une manière aussi réaliste que possible pour les utilisateurs de la réalité virtuelle [93]. Idéalement, l'interaction avec un environnement virtuel ou à distance devrait être aussi simple et réel, similaire au ressenti un outil à main, ou avec ses propres doigts. Selon [93], la technologie disponible aujourd'hui ne permet pas encore d'atteindre cet objectif ambitieux, qui voudrait que les systèmes soient simplifiés pour ne contenir que l'information la plus importante pour la tâche à accomplir. Afin de fournir des forces de réaction fidèles, il est toujours nécessaire d'avoir un bon modèle virtuel de la scène avec laquelle l'utilisateur interagit. En haptique, l'interaction entre l'utilisateur et le modèle virtuel est fournie via l'interface haptique. Ces modèles virtuels peuvent être construits à partir de fonctions mathématiques (objets implicites et paramétriques) ou une collection de primitives géométriques [87, 142].

[1] le sentiment de télé-présence est le fait que l'utilisateur a le sentiment d'être présent et d'agir directement dans l'environnement, sans utiliser de dispositifs supplémentaires.

Les environnements virtuels sont devenus également très populaires et sont utilisés dans de nombreux domaines, comme le prototypage [99], des formations pour différents dispositifs d'aide et dans l'exécution des tâches difficiles [43, 56], montage virtuel [7, 176]. De nombreuses tentatives de simulation de collaboration ont été orientées vers des applications avec environnements virtuels (EV) multi-utilisateurs. Ainsi, pour en citer quelques-unes, SIMNET [30] et NPSNET [110] étaient des prototypes d'EV en temps réel pour des formations militaires. Ils utilisent la méthode de dead-reckoning[2] pour donner des scènes aussi réalistes que possible (ou état de synchronisation) de l'environnement simulé pour les participants. Certains systèmes multi-utilisateurs de réalité virtuelle ont intégré des interfaces haptiques pour permettre aux utilisateurs d'interagir directement (mécaniquement). Basdogan *et al.* [22]ont développé un jeu collaboratif grâce à l'interaction haptique, et Hudson *et al.* [79] ont mis en place un système de Nanomanipulateur avec des microscopes à force atomique.

Comme on l'a dit précédemment, le problème de commande majeur de ces systèmes est la présence de retards. Des retards importants sont largement considérés depuis plusieurs années, conduisant *naturellement* à la téléopération/haptique sur Internet. La présence de retards peut induire des instabilités et des comportements complexes ; ces sujets ont été largement traités dans la littérature (voir, par exemple, [57, 98, 129, 199] et les références qui s'y trouvent). Une excellente vue d'ensemble de certains résultats existants, ainsi que de certains problèmes ouverts, est donnée dans [153]. Les systèmes distribués avec des retards sont présents dans de nombreux domaines scientifiques tels que l'économie [86], la dynamique des populations [42], le contrôle du trafic [83], les systèmes à base de réseau/Internet [188, 193], ce ne sont juste que quelques exemples de domaines où les retards apparaissent naturellement dans des systèmes dynamiques.

Les retards apparaissent souvent dans la boucle fermée en raison de l'acquisition des données issues des signaux de sortie et d'excitation, la transmission d'informations, le traitement des données en ligne, le calcul et l'application des commandes en force. En dépit des efforts pour minimiser les retards, ceux-ci ne peuvent pas être totalement éliminés, même avec les progrès technologiques, en raison des limites physiques. Le retard de l'information est souvent négligeable, mais, dans certains cas, elle peut être cruciale. Il existe une abondante littérature sur la commande des systèmes à retard (voir, par exemple, [138, 168] et les références qui s'y trouvent).

Comme pour la téléopération, en haptique le problème des retards et leur influence sur les performances reste ouvert. Pour ce qui concerne les systèmes haptiques via Internet, qui sont aujourd'hui beaucoup plus fréquemment utilisés, les problèmes rencontrés en commande sont très

[2]Méthode utilisée pour la navigation, qui caractérise le processus de calcul de la position actuelle en utilisant une position préalablement déterminée, en fonction des vitesses connues ou estimées.

similaires. Il existe deux sources de retards : le canal de communication et le temps de traitement de la réalité virtuelle. Pour des environnements virtuels complexes, le temps de traitement peut augmenter considérablement et peut présenter des effets et des comportements indésirables [47].

Un système haptique/téléopération idéal doit avoir :

- une erreur de suivi de position aussi faible que possible entre l'interface haptique et l'objet virtuel (maître et esclave pour les systèmes de téléopération) ;
- un degré élevé de transparence, c'est-à-dire en mouvement libre, le retour d'effort ressenti à l'extrémité de l'interface haptique doit être aussi faible que possible et en cas de contact dur, une réaction raide est souhaitée.

Plus précisément, en mouvement libre, l'effet des retards peut être ressenti par le phénomène de viscosité (retour d'effort élevé de l'interface haptique), et dans le cas d'un contact avec l'environnement, l'effet de l'impact ne sera pas raide ; dans le pire cas, on peut perdre la stabilité du système en raison des retards. Par conséquent, les retards doivent être pris en compte et intégrés lors de la conception des lois de commande. Cependant, un compromis entre stabilité, erreur de suivi et transparence sera toujours nécessaire.

La première étape pour résoudre ce problème a été de faire évoluer les algorithmes existants en téléopération dans le domaine de l'haptique. Comme il existe un grand nombre d'algorithmes et de méthodes, afin d'évaluer leurs performances, des études comparatives des systèmes de téléopération, ainsi que pour les systèmes haptiques, peuvent être trouvées dans la littérature, comme [112, 154] ou [104, 164], respectivement. Dans un deuxième temps, il s'agit d'utiliser les données supplémentaires disponibles à partir de l'environnement virtuel : l'information de l'environnement virtuel sur la distance entre les objets, les collisions possibles et de nombreux autres détails concernant le système, en vue d'améliorer les algorithmes de commande.

Une méthode simple et fiable pour contrôler les systèmes à retard reste le prédicteur de Smith [174]. Cette approche s'avère concluante dans diverses applications. Plus précisément, un système de commande à base de prédicteur de Smith pour la synchronisation de mouvement dans des environnements virtuels en présence de retards importants est présenté dans [35]. En téléopération, parmi de nombreuses solutions proposées pour résoudre le problème lié au retard, des bons résultats sont également obtenus avec le prédicteur de Smith, voir [173, 189] pour plus de détails. De bons résultats sont également obtenus pour la commande des systèmes avec communication via des réseaux sans fil, voir, par exemple, [46]. Les exemples sont nombreux, l'idée centrale reste le prédicteur de Smith, qui a été modifié/adapté en fonction des besoins ou capacités des systèmes. Une étude complète sur les variantes du prédicteur de Smith peut être trouvée dans [137].

D'une manière générale, afin d'obtenir des performances élevées, les effets induits par les retards doivent être soigneusement analysés et pris en compte. Sinon, les performances du système seront réduites, ou, comme il a été dit, au pire le système peut devenir instable et donc inutilisable. Même s'il y a beaucoup d'études et de solutions dans ce sens, le problème principal lié aux retards reste encore ouvert. Parmi les approches proposées dans la littérature, il existe des solutions proposant des garanties supplémentaires sur la stabilité, mais sans tenir compte de la transparence [14, 100, 101, 159, 160].

Le travail de cette thèse est axé sur les systèmes haptiques, élaborant de nouvelles stratégies en tirant profit du fait que les systèmes haptiques offrent des interactions avec l'environnement plus claires et prévisibles, des informations qui peuvent être facilement utilisées afin d'améliorer les performances du système.

L'objectif principal est donc de trouver des nouvelles solutions de commande pour prévenir les situations indésirables liées aux retards, augmentant ainsi les performances globales des systèmes. Partant de la situation idéale, lorsque le système n'est pas affecté par des retards (de communication et/ou de calcul), les performances et les garanties du système en présence de retards devront se rapprocher au maximum de ce cas idéal.

Chapitre 1

Dans le cadre de la téléopération, [112, 154] proposent des études comparatives de nombreuses stratégies de commande, tandis que pour l'haptique [164] compare seulement trois méthodes classiques (Proportionnel-Dérivé, PD avec observateur et passivité et variables d'onde) dans le cas de l'haptique collaboratif sur Internet. Le premier chapitre présente donc une étude comparative plus approfondie des algorithmes de commande existants pour les interfaces haptiques et les environnements virtuels soumis à des retards de communication. Plus précisément, des méthodes comme le régulateur Proportionnel-Dérivé (PD), PD avec dissipation locale [101], PD avec observateur de passivité [14, 159, 160], PD avec changement de consigne [100], par variable d'onde [130] et prédicteur de Smith [36], ont été introduites dans le domaine haptique directement à partir de la téléopération.

Les équations du mouvement classiques (non linéaires) pour deux robots similaires dans le cadre des systèmes haptiques/téléopération [136] (obtenues à partir de l'égalité des forces, des accélérations, des frictions et sur la base du principe fondamental de la dynamique) sont données par :

$$M_h(x_h)\ddot{x}_h(t) + B_h(x_h,\dot{x}_h)\dot{x}_h = -F_h(t) + F_{op}(t), \quad (1)$$

$$M_v(x_v)\ddot{x}_v(t) + B_v(x_v,\dot{x}_v)\dot{x}_v = F_v(t) - F_e(t), \quad (2)$$

où x_h, x_v représentent les positions de l'interface haptique/objet virtuel, F_{op}, F_e sont les forces de l'opérateur humain/l'environnement, F_h, F_v - les signaux de commande en effort, M_h, M_v représentent les matrices d'inertie, symétriques définies positives, et B_h, B_v sont les matrices de Coriolis de l'interface haptique et de l'objet virtuel, respectivement. L'idée principale est d'utiliser deux régulateurs PD, l'un pour contrôler l'interface haptique (robot maître), l'autre pour l'objet virtuel (robot esclave).

Dans le cas de PD classiques, les équations des lois de commande sont données comme suit :

$$F_h(t) = \underbrace{K_{d_h}(\dot{x}_h(t) - \dot{x}_v(t-\tau_2))}_{\text{action D retardée}} \underbrace{+K_{p_h}(x_h(t) - x_v(t-\tau_2))}_{\text{action P retardée}}, \tag{3}$$

$$F_v(t) = \underbrace{K_{d_v}(\dot{x}_h(t-\tau_1) - \dot{x}_v(t))}_{\text{action D retardée}} \underbrace{+K_{p_v}(x_h(t-\tau_1) - x_v(t))}_{\text{action P retardée}}, \tag{4}$$

où τ_1, τ_2 représentent les retards finis en amont et en aval (constants ou variables) $K_{p_h}, K_{d_h}, K_{p_v}, K_{d_v}$ sont les gains des régulateurs PD pour l'interface haptique et l'objet virtuel, respectivement.

La plupart des méthodes (PD avec dissipation locale, PD avec observateur de passivité, PD avec changement de consigne et variable d'ondes) s'appuient sur la théorie de la passivité. Un système est dit passif si et seulement si :

$$\int_0^t F(\tau)\dot{x}(\tau)d\tau + E(0) \geq 0, \quad \forall t > 0, \tag{5}$$

où $\dot{x}$ et F sont les variables indiquant la vitesse et la force respectivement, et $E(0)$ est l'énergie stockée initialement dans le système à $t = 0$. La passivité est aussi une condition suffisante pour la stabilité [130].

Ainsi, l'idée principale de la méthode PD avec dissipation locale est d'inclure un terme supplémentaire pour ajouter un plus de stabilité. Ce terme est une dissipation locale agissant dans le but de maintenir la passivité du système. La méthode a été proposée par [101], basée sur le concept de passivité, la technique de Lyapunov-Krasovskii pour les systèmes retardés, et l'identité de Parseval. Dans ce cas, les forces $F_h(t)$, $F_v(t)$ à partir des relations (3)-(4) sont réalisés comme suit:

$$F_h(t) = \underbrace{K_{d_h}(\dot{x}_h(t) - \dot{x}_v(t-\tau_2))}_{\text{action D retardée}} \underbrace{+(-K_{diss} + P_e)\dot{x}_h(t)}_{\text{dissipation}} \underbrace{+K_{p_h}(x_h(t) - x_v(t-\tau_2))}_{\text{action P retardée}}, \tag{6}$$

$$F_v(t) = \underbrace{K_{d_v}(\dot{x}_h(t-\tau_1) - \dot{x}_v(t))}_{\text{action D retardée}} \underbrace{-(-K_{diss} + P_e)\dot{x}_v(t)}_{\text{dissipation}} \underbrace{+K_{p_v}(x_h(t-\tau_1) - x_v(t))}_{\text{action P retardée}}, \tag{7}$$

où K_{diss} ($K_{diss} = 0,05K_p$) est le gain de dissipation qui assure la passivité sur l'action dérivée retardée et P_e est un amortissement supplémentaire pour assurer la coordination entre l'interface haptique et l'objet virtuel (pour plus de détails, voir [101]).

La méthode PD avec observateur de passivité propose de mesurer l'énergie du système à chaque étape, et si cette énergie devient négative, d'introduire de l'énergie pour maintenir la stabilité. Il y a deux configurations possibles : parallèle ou série ; parmi les deux configurations possibles, seule la connexion en série est envisagée, suite à l'absence de capteur d'effort dans les systèmes considérés. Ainsi, l'observateur de passivité est donné par :

$$F_h(t) = F_h'(t) + \alpha_h(t)\dot{x}_h(t), \quad \text{ou} \quad \alpha_h(t) = \begin{cases} \frac{-E_{obs}(t)}{\Delta T \dot{x}_1(t)^2} \text{ if} & E_{obs} < 0 \\ 0 & E_{obs} \geq 0 \end{cases} \tag{8}$$

Avec ces considérations, les équations de commande (3)-(4) deviennent :

$$F_h(t) = \underbrace{K_{d_h}(\dot{x}_h(t) - \dot{x}_v(t-\tau_2))}_{\text{action D retardée}} \underbrace{+K_{p_h}(x_h(t) - x_v(t-\tau_2))}_{\text{action P retardée}} \underbrace{+\alpha_h(t)\dot{x}_h(t)}_{\text{CP}}, \tag{9}$$

$$F_v(t) = \underbrace{K_{d_v}(\dot{x}_h(t-\tau_1) - \dot{x}_v(t))}_{\text{action D retardée}} \underbrace{+K_{p_v}(x_h(t-\tau_1) - x_v(t))}_{\text{action P retardée}} \underbrace{-\alpha_v(t)\dot{x}_v(t)}_{\text{CP}}, \tag{10}$$

où α_h est le coefficient correspondant pour l'interface haptique et α_v est le coefficient de l'objet virtuel, défini de façon similaire.

Basée sur la même théorie que la méthode précédente, comme indiqué dans [100], l'approche par changement de consigne a une grande tolérance à la perte de paquets et retards variables. Afin de faire respecter la passivité, l'action de commande est limitée avant son application, une stratégie qui ne possède pas de singularité dans le résultat. Plus précisément, dans notre cas, selon [100] le PD avec changement de consigne peut être exprimé comme suit :

$$\begin{aligned} &\min_{\bar{x}_h(t)} \parallel x_h(t) - \bar{x}_h(t) \parallel \\ &\text{s.t. } E_h(t) = E_h(t-1) + K_{d_h}\dot{x}_h(t-1)^2 - \tfrac{1}{2}K_{p_h}(-\bar{x}_h(t) + \\ &\quad + \bar{x}_h(t-1))(2x_h(t) - \bar{x}_h(t) - \bar{x}_h(t-1)) \geq 0 \end{aligned} \tag{11}$$

$$\begin{aligned} &\min_{\bar{x}_v(t)} \parallel x_v(t) - \bar{x}_v(t) \parallel \\ &\text{s.t. } E_v(t) = E_v(t-1) + K_{d_v}\dot{x}_v(t-1)^2 - \tfrac{1}{2}K_{p_v}(-\bar{x}_v(t) + \\ &\quad + \bar{x}_v(t-1))(2x_v(t) - \bar{x}_v(t) - \bar{x}_v(t-1)) \geq 0 \end{aligned} \tag{12}$$

où $\bar{x}_h(t)$ et $\bar{x}_v(t)$ représentent les positions modulées définies pour l'interface haptique et l'objet virtuel, et E_h, E_v sont les énergies accumulées côté haptique et virtuel, respectivement. Avec ces considérations, les équations (3)-(4) peuvent être réécrites comme suit :

$$F_h(t) = \underbrace{K_{d_h}(\dot{\bar{x}}_h(t) - \dot{x}_v(t-\tau_2))}_{\text{action D retardée}} \underbrace{+K_{p_h}(\bar{x}_h(t) - x_v(t-\tau_2))}_{\text{action P retardée}}, \tag{13}$$

$$F_v(t) = \underbrace{K_{d_v}(\dot{x}_h(t-\bar{\tau}_1) - \dot{\bar{x}}_v(t))}_{\text{action D retardée}} \underbrace{+K_{p_v}(x_h(t-\tau_1) - \bar{x}_v(t))}_{\text{action P retardée}}. \tag{14}$$

Une dernière méthode basée sur la théorie de passivité, les variables d'onde, présente une extension qui assure une certaine robustesse dans le cas de retards arbitraires [133]. Selon [134], la transmission des variables d'onde fournit un moyen efficace et simple de mise en œuvre tout en garantissant la stabilité du système pour des retards inconnus. La méthode est applicable aux systèmes non-linéaires et peut manipuler des modèles inconnus et des incertitudes importantes, ce qui convient pour l'interaction avec des environnements physiques réels. La transformation d'onde de base concerne la vitesse, la force, et les ondes de part et d'autre [133]. Dans le cas présenté dans ce travail, $\dot{x}_h$ et $\dot{x}_v$ seront convertis en u_m et v_s :

$$u_m(t) = \frac{b\dot{x}_h(t)+\dot{x}_{v_d}(t)}{\sqrt{2b}}, \qquad v_s(t) = \frac{\dot{x}_v(t)-b\dot{x}_{h_d}(t)}{\sqrt{2b}}, \tag{15}$$

où b est l'impédance caractéristique et peut être une constante positive ou une matrice symétrique définie positive et $\dot{x}_{h_d}$, $\dot{x}_{v_d}$ sont les sorties retardées après l'application de la transformation des variables d'onde. L'impédance caractéristique b a également le rôle d'un paramètre de réglage qui peut réduire la vitesse du mouvement par rapport au niveau des forces et influence de nombreuses autres caractéristiques du point de vue perception de l'utilisateur. Aucune information n'est perdue ou gagnée par le codage des variables de cette manière. Dans ce cas, les équations (3)-(4) deviennent :

$$F_h(t) = \underbrace{K_{d_h}(\dot{x}_h(t) - \dot{x}_{v_d}(t))}_{\text{action D retardée}} \underbrace{+K_{p_h}(x_h(t) - x_{v_d}(t))}_{\text{action P retardée}}, \tag{16}$$

$$F_v(t) = \underbrace{K_{d_v}(\dot{x}_{h_d}(t) - \dot{x}_v(t))}_{\text{action D retardée}} \underbrace{+K_{p_v}(x_{h_d}(t) - x_v(t))}_{\text{action P retardée}}, \tag{17}$$

avec :

$$\begin{aligned} \dot{x}_{h_d}(t) &= \tfrac{1}{b}(-\dot{x}_v(t) + \sqrt{2b}u_m(t-\tau_1)), \\ \dot{x}_{v_d}(t) &= b\dot{x}_h(t) + \sqrt{2b}v_s(t-\tau_2). \end{aligned} \tag{18}$$

Une autre méthode couramment utilisée pour contrôler les systèmes à retards est fondée sur le prédicteur de Smith [174]. L'idée de base est de construire un correcteur pour un système équivalent plus simple, pour lequel le retard est rejeté en dehors de la boucle. Il est important de noter que cette méthode ne fonctionne que sous certaines contraintes sur la dynamique du système et/ou les retards. Un système de commande à base de prédicteur de Smith pour la synchronisation des mouvements dans des environnements virtuels sous des grands retards est présenté dans [35]. En téléopération, parmi les nombreuses solutions proposées pour résoudre le problème de retards, de bons résultats sont également obtenus avec le prédicteur de Smith, voir [12, 173, 189] pour plus des détails. Dans notre cas, toute la partie virtuelle est considérée comme un processus retardé, et ainsi un seul prédicteur côté haptique est utilisé ; les équations

(3)-(4) deviennent :

$$\begin{aligned} F_h(t) = \quad & \underbrace{K_{d_h}(\dot{x}_h(t) - \dot{x}_v(t-\tau_2) + \dot{\hat{x}}_v(t-(\tau_1+\tau_2)) - \dot{\hat{x}}_v(t)}_{\text{action D retardée}} \\ & \underbrace{+K_{p_h}(x_h(t) - x_v(t-\tau_2) + \hat{x}_v(t-(\tau_1+\tau_2)) - \hat{x}_v(t))}_{\text{action P retardée}}, \end{aligned} \tag{19}$$

$$F_v(t) = \underbrace{K_{d_v}(\dot{x}_h(t-\tau_1) - \dot{x}_v(t))}_{\text{action D retardée}} \underbrace{+K_{p_v}(x_h(t-\tau_1) - x_v(t))}_{\text{action P retardée}}, \tag{20}$$

où $\dot{\hat{x}}_v$, $\hat{x}_v$ représentent la vitesse et la position estimées de l'objet virtuel.

Avant de présenter les résultats de la comparaison, définissons les trois critères de performance retenus pour établir cette comparaison :

- la stabilité du système doit être garantie dans toutes les situations ;
- l'erreur de suivi entre l'interface haptique et l'objet virtuel (maître et esclave) en mouvement libre ainsi qu'en mouvement restreint doit être faible ;
- la transparence du système, qui peut être définie comme le sentiment de l'utilisateur final d'agir directement avec la réalité virtuelle ou avec un environnement distant, sans dispositifs supplémentaires, doit être assurée (voir, par exemple [76]).

La mise en œuvre expérimentale des lois présentées ci-dessus permet d'en dégager les principales caractéristiques. Comme déjà mentionné, une architecture position-position est utilisée, ce qui est défini par la transmission de la position entre les correcteurs haptique et virtuel. L'objectif principal est de mettre en évidence les points forts, ainsi que les inconvénients, de chaque méthode par rapport aux retards, du point de vue de l'erreur de suivi et du degré de transparence. Pour cette étude, une plateforme haptique à un degré de liberté (1-ddl) est utilisée. Afin d'assurer une maitrise totale des retards de communication et du temps de calcul, tous les algorithmes de commande (pour l'interface haptique/objet virtuel) et les simulations de l'environnement virtuel seront exécutés sur le même ordinateur. A partir du cas idéal (sans retards), deux types de réglage sont considérés : un permettant d'obtenir les meilleures performances en mouvement libre, un autre conférant les meilleures performances en mouvement restreint. La figure 1 résument les résultats obtenus, plus précisément l'erreur de suivi maximale en mouvement restreint, et le retour d'effort moyen en mouvement libre (mesuré à une vitesse de 8 rad/s) pour chaque méthode.

Après analyse des résultats expérimentaux, les différences entre les méthodes n'apparaissent pas comme étant significatives. Aucune de ces méthodes ne peut fournir un degré élevé de transparence et une erreur de suivi faible en même temps. Le compromis entre la transparence et l'erreur de suivi en position est évident. La présence des retards dégrade les performances du

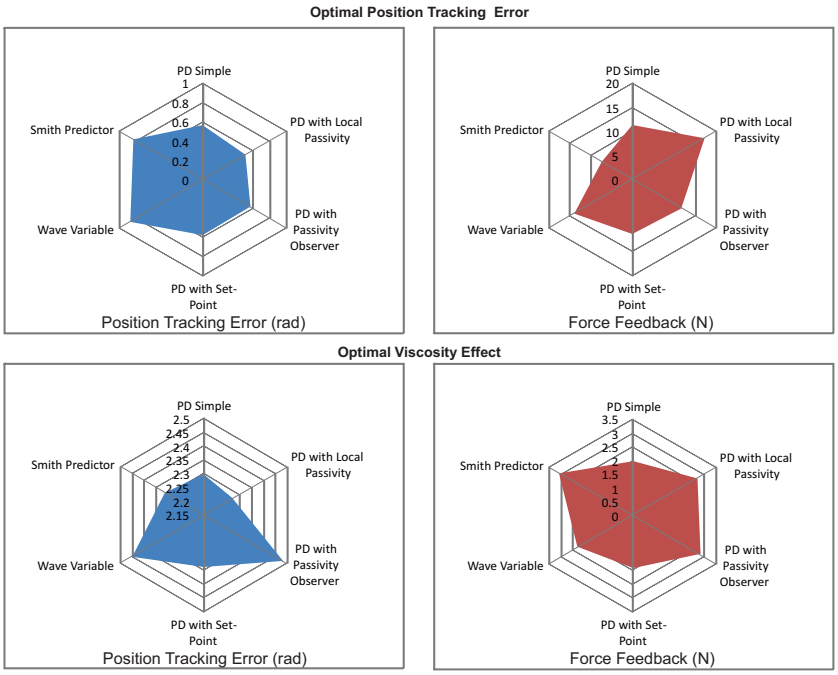


FIGURE 1: Erreur de suivi et retour d'effort dans le cas d'un retard de 50 ms.

système. D'une manière générale, même si l'on est capable de détecter de petites différences entre les méthodes, le réglage optimal pour une configuration détériorera les performances pour l'autre.

Chapitre 2

Le deuxième chapitre introduit les outils théoriques nécessaires à l'analyse de la stabilité des systèmes retardés dans différentes configurations, ainsi que pour certaines limitations physiques, et fournit des conseils de réglage d'un point de vue pratique.

Comme mentionné, les systèmes haptiques, ainsi que des systèmes téléopérés, sont caractérisés par la présence de retards de communication qui doivent être pris en compte pour définir la stratégie de commande. Plus précisément, les exigences potentiellement conflictuelles, comme une réponse rapide et la stabilité robuste, ont besoin d'une analyse de la commande afin de prendre la bonne décision, car ces systèmes sont soumis à des changements de position fréquents. Il est intéressant de souligner que ces systèmes ont des fonctionnalités basées sur l'erreur de suivi et de vitesse. Dans ces circonstances, la stratégie de commande la plus appropriée est d'utiliser une commande basée sur les actions Proportionnelle Dérivée, puisque l'action intégrale confère une erreur nulle mais induit une perte souvent importante de stabilité.

La figure 2 présente le schéma général de commande d'une interface haptique connectée à un environnement virtuel.

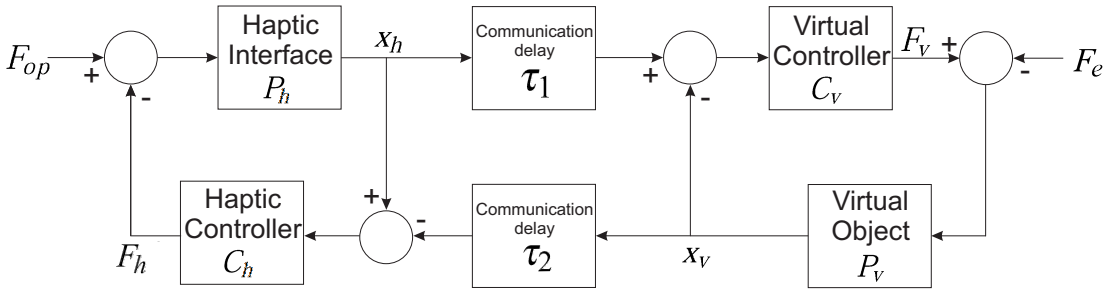


FIGURE 2: Schéma général de contrôle pour systemés haptique

Les équations décrivant la réponse du système peuvent être écrites comme suit :

$$X_h(s) = P_h(s)\left(F_{op}(s) - C_h(s)\left(X_h(s) - e^{-\tau_2 s}X_v(s)\right)\right), \tag{21}$$

$$X_v(s) = P_v(s)\left(-F_e(s) + C_v(s)\left(-X_v(s) + e^{-\tau_1 s}X_h(s)\right)\right), \tag{22}$$

Les fonctions de transfert $P_{h/v}(s)$ (l'interface haptique ainsi que l'objet virtuel sont modélisés par un système du deuxième ordre masse-ressort-amortisseur, et $C_{h/v}(s)$ sont considérées comme suit :

$$P_h(s) = P_v(s) = \frac{1}{s(ms+b)} =: P(s), \tag{23}$$

$$K_{p_h} = K_{p_v} =: K_p, \quad K_{d_h} = K_{d_v} =: K_d, \tag{24}$$

$$C_h(s) = C_v(s) = K_p + K_d s =: C(s), \tag{25}$$

avec m représentant la masse, et b le coefficient de frottement visqueux. Les robots sont modélisés comme des systèmes linéaires, puisque l'interface haptique ne présente aucun comportement particulier qui ne serait pas couvert par un modèle linéaire, et le robot virtuel est modélisé dans un cas idéal.

Par conséquent, avec les définitions du processus et des correcteurs, l'équation caractéristique du système en boucle fermée peut être écrire comme suit :

$$(1 + P(s)C(s))^2 - (P(s)C(s))^2 e^{-(\tau_1+\tau_2)s} = 0, \tag{26}$$

qui est simplement équivalent à :

$$\left(1 + P(s)C(s) + P(s)C(s)e^{-\tau s}\right)\left(1 + P(s)C(s) - P(s)C(s)e^{-\tau s}\right) = 0, \quad \text{avec} \quad \tau := \frac{(\tau_1 + \tau_2)}{2}.$$

L'analyse de la stabilité en présence de retards (τ_1, τ_2) se base surune approche géométrique, considérant les régions de stabilité dans l'espace des gains du régulateur. Plus précisément, la stabilité du système en boucle fermée est donnée par la localisation des racines de l'équation caractéristique (26). Introduisons pour cela la fonction caractéristique $\Delta : \mathbb{C} \times \mathbb{R} \times \mathbb{R} \times \mathbb{R}_+ \to \mathbb{C}$ donnée par :

$$\Delta(s; K_p, K_d, \tau) = (1 + P(s)(1 + e^{-s\tau})(K_p + K_d s))(1 + P(s)(1 - e^{-s\tau})(K_p + K_d s)), \tag{27}$$

qui possède un nombre infini de racines (voir, par exemple [129]). Il est important de souligner que les racines d'une telle fonction caractéristique sont dépendantes continument des paramètres du système, et une perte ou un gain de stabilité peuvent être détectés si les racines caractéristiques traversent l'axe imaginaire (voir [152] ou [116] et les références qui s'y trouvent). Alors, pour des retards positifs connus $(\tau_1,\ \tau_2)$ et pour une fréquence ω dans $\mathbb{R}_+ \setminus \{k\pi/\tau; k \in \mathbb{N}_+\}$ les points de passage correspondants sont donnés par :

$$\begin{cases} K_p^{\pm} = -\Re\left(\frac{1}{P(j\omega)(1 \pm e^{-\tau j\omega})}\right) \\ K_d^{\pm} = -\frac{1}{\omega}\Im\left(\frac{1}{P(j\omega)(1 \pm e^{-\tau j\omega})}\right) \end{cases}, \tag{28}$$

où (K_p^+, K_d^+) représentent l'ensemble des solutions correspondant à la première partie de (27), et (K_p^-, K_d^-) correspondent à la seconde partie. Dès lors, la variation d'ω dans un intervalle donné $\Omega_l\ (0, \infty)$, (28)-(28) définit les courbes qui séparent l'espace des paramètres du correcteur en plusieurs régions, telles que chaque région ait le même nombre de racines caractéristiques instables (voir, par exemple [116], pour une discussion plus détaillée sur la problématique de la méthode de D-décomposition proposée par Neimark [127]).

La figure 3 présente la région de stabilité dans l'espace des paramètres (K_p, K_d) pour un système avec : $m = 1\ kg,\ b = 0,1\ Ns/m$, et $\tau_1 = \tau_2 = 0,05\ s$.

En général, ces systèmes sont affectés par des retards variables et moins souvent par des retards fixes. Une approche possible lorsqu'il s'agit de retards variables consiste alors à utiliser des distributions en tant que modèles illustrant le comportement des retards. Dans la plupart des cas, le comportement du retard peut être étudié afin de déterminer sa forme et sa variation, pour ensuite introduire un modèle de distribution dans la loi de commande. Fondamentalement, l'utilisation de distributions de retards est un compromis raisonnable entre le cas moins courant de retards fixes et le cas plus complexe de retards variant dans le temps. En considérant des modèles distribués, la précision de l'analyse et la construction de lois de commande semblent être améliorées dans les cas considérés.

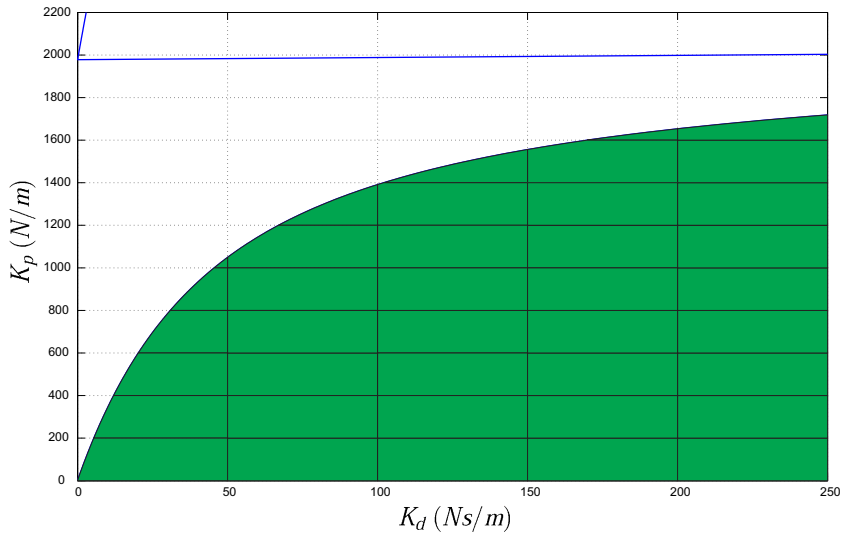


FIGURE 3: Zone de stabilité (en vert) dans l'espace des paramètres (K_p, K_d).

Dans la suite, les retards simples sont donc remplacés par des retards distribués. Les équations décrivant la réponse du système peuvent être écrites comme suit à partir de (21)-(22) :

$$X_h(s) = P_h(s)\left(F_{op}(s) - C_h(s)\left(X_h(s) - D_2(s)X_v(s)\right)\right), \tag{29}$$

$$X_v(s) = P_v(s)\left(-F_e(s) + C_v(s)\left(-X_v(s) + D_1(s)X_h(s)\right)\right), \tag{30}$$

où $D_1(s), D_2(s)$ représentent les retards distribués. Deux types de distribution sont envisagées : la distribution uniforme et la distribution gamma avec gap. Dans le premier cas, l'incertitude sera considérée comme ayant une variation aléatoire, sans aucune information supplémentaire, à part les valeurs minimale et maximale du retard. La figure 4 présente une distribution uniforme du retard.

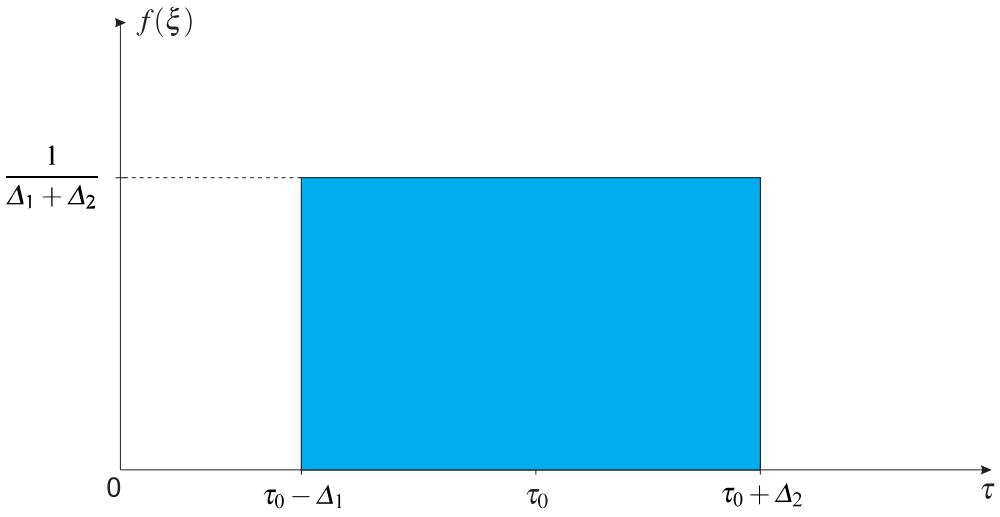


FIGURE 4: Distribution uniforme.

Considérant $\Delta_1, \Delta_2 > 0$ et $\tau_0 \geq \Delta_1$, le noyau de la distribution uniforme est donné par :

$$f(\xi) = \begin{cases} \frac{1}{\Delta_1+\Delta_2}, & \text{if } \tau_0 - \Delta_1 < \xi < \tau_0 + \Delta_2, \\ 0, & \text{autrement}, \end{cases} \tag{31}$$

où τ_0 est le retard de référence et Δ_1, Δ_2 représentent l'incertitude du retard dans les deux sens. La transformée de Laplace de la distribution uniforme est donnée par :

$$D(s) = \frac{e^{-s(\tau_0-\Delta_1)} - e^{-s(\tau_0+\Delta_2)})}{s(\Delta_1+\Delta_2)}. \tag{32}$$

Avec ces considérations et tenant compte du fait que $D_1 = D_2 =: D$, l'équation caractéristique pour ce cas se réécrit comme suit :

$$\Psi(s,\ K_p,\ K_d,\ \tau_0,\ \Delta_1,\ \Delta_2) = \Psi_1(s,K_p,K_d) + \Psi_2(s,K_p,K_d) \left(\frac{e^{-s(\tau_0-\Delta_1)} - e^{-s(\tau_0+\Delta_2)})}{s(\Delta_1+\Delta_2)} \right)^2, \tag{33}$$

En appliquant la méthode de stabilité présentée précédemment, la région de stabilité est obtenue pour ce cas, voir figure 5.

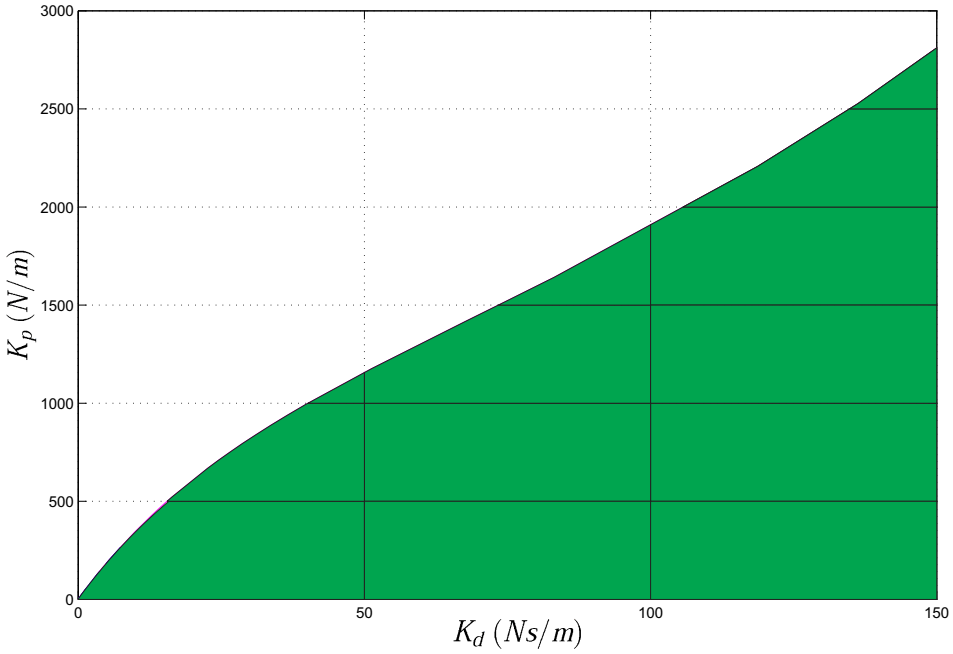


FIGURE 5: Région de stabilité dans le cas d'une distribution uniforme (K_p, K_d), avec $\tau_1 = \tau_2 = 0.05sec$ et $\Delta_1 = \Delta_2 = 0.01sec$.

Les études sur le problème de la commande des objets via des réseaux de communication [157] ont mises en évidence que, dans ce cas, les retards peuvent être modélisés par une distribution gamma avec gap (voir, par exemple, [121] pour plus de détails). Dans le cas de la distribution

gamma avec gap, le noyau f est donné par :

$$f(\xi) = \begin{cases} 0, & \xi < \tau, \\ \frac{(\xi-\tau)^{n-1} e^{-\frac{\xi-\tau}{T}}}{T^n (n-1)!}, & \xi \geq \tau, \end{cases} \tag{34}$$

où $n \in \mathbb{N}$, $T > 0$ et $\tau \geq 0$ définissent le gap. La figure 6 présente un exemple de distribution gamma avec gap. La transforme de Laplace de cette distribution est donnée par :

$$D(s) = L(f) = \frac{e^{-s\tau}}{(1+sT)^n}. \tag{35}$$

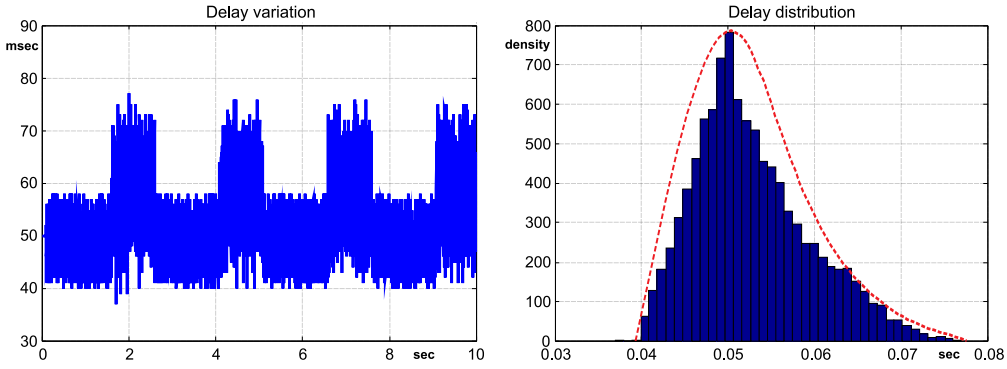


FIGURE 6: Exemple de distribution gamma avec gap ($\tau = 0.04$ *sec*, $T = 0.01$ *sec*, $n = 1$).

Dans le cas d'une distribution gamma avec gap, la fonction caractéristique s'écrit comme suit :

$$\Psi(s,\ K_p,\ K_d,\ \tau,\ T,\ n) = \Psi_1(s,K_p,K_d) + \Psi_2(s,K_p,K_d) \left(\frac{e^{-\tau s}}{(1+sT)^n} \right)^2. \tag{36}$$

A partir de cette équation, une approche similaire à celle mise en œuvre pour la distribution uniforme permet de déterminer la région de stabilité.

Comme mentionné précédemment, une méthode fiable pour résoudre les problèmes liés aux systèmes à retards reste le prédicteur de Smith [174]. L'analyse de stabilité est proposée selon deux points de vue :

- régions de stabilité dans l'espace des paramètres du prédicteur, (similaire au régulateur PD) ;
- stabilité dans l'espace des retards - système et prédicteur de Smith.

Pour le premier cas, une analyse similaire au régulateur PD peut être menée dans le cas du prédicteur de Smith. Il est important de mentionner qu'en cas de retards fixes et connus, l'utilisation du prédicteur de Smith ramène le problème au cas non retardé. Pour les retards incertains, en

plus des deux distributions (uniforme et gamma avec gap), la distribution normale et la situation avec incertitudes fixes sont étudiées. Plus précisément, comme la transformation de Laplace de la distribution normale est probabiliste et non déterministe comme pour les autres distributions considérées, une analyse du comportement moyen de la variation du retard doit être envisagée. Cette technique a permis de conclure que les conditions de stabilité correspondant au cas idéal sont également valables dans le cas de retards gaussiens pour un comportement moyen.

De plus, dans le cas du prédicteur de Smith, une analyse de stabilité doit être menée lorsque les retards du système et du prédicteur sont différents mais fixes. Dans tous les cas, les régions de stabilité sont présentées dans l'espace des paramètres de prédicteur.

Le deuxième cas - stabilité dans l'espace des retards - est brièvement décrit. Considérons maintenant que les gains du régulateur sont fixés $K_p = K_p^*,\ K_d = K_d^*$, et analysons l'influence sur la stabilité vis-à-vis de la variation du retard du système. Selon [124], pour un $\omega \in \Omega$ donnée, l'ensemble $\Im_\omega$, comprenant toutes les paires $((\tau_1+\tau_2),\tau_S)$ peut être trouvé, satisfaisant $h(j\omega,\ K_p^*,\ K_*^d,\ \tau_1,\ \tau_2,\ \tau_S) = 0$ (fonction de transfert en boucle fermée), sous la forme :

$$(\tau_1+\tau_2) = (\tau_1+\tau_2)^{u\pm}(\omega) = \frac{\angle h(j\omega) + (2u-1)\pi \pm q}{\omega}, \quad u = u_0^\pm, u_0^\pm+1, u_0^\pm+2, \ldots \ , \tag{37}$$

$$\tau_S = \tau_S^{v\pm}(\omega) \qquad = \frac{\angle h(j\omega) + 2v\pi \mp q}{\omega}, \qquad v = v_0^\pm(u), v_0^\pm(u)+1, v_0^\pm(u)+2, \ldots \ , \tag{38}$$

où $q \in [0,\pi]$ est donné par :

$$q(j\omega) = \cos^{-1}\left(\frac{1}{2|h(\omega)|}\right) \tag{39}$$

et u_0^+, u_0^- sont les plus petits entiers (peut-être dépendant de ω) tels que les valeurs correspondantes $(\tau_1 + \ tau_2)^{u_0^+ +}, (\tau_1+\tau_2)^{u_0^- -}$ soient positives, et v_0^+ et v_0^- sont des entiers dépendant de u. Afin de mieux illustrer cette approche, un exemple dans le domaine de l'haptique est examiné ci-dessous. Considérant $m = 1\ kg$, $b = 0,1\ Ns/m$, et les gains du régulateur PD choisis $K_p = 1500\ N/m$ et $K_d = 80\ Ns/m$, la figure 7 présente les courbes dans ce cas. La région de stabilité est déterminée en sachant que le système est stable lorsque les deux retards sont égaux, et ensuite les frontières de la stabilité peuvent être déterminées sur le graphique.

Ce résultat donne un aperçu général des limites du système en termes de variations des retards. Plus de détails peuvent être trouvés dans [124]. En règle générale, le retard considéré dans le prédicteur de Smith est supposé fixé. Une fois la configuration complète connue, il est facile de voir quelles sont les limites minimale et maximale des incertitudes correspondant au retard du point de vue de la stabilité.

Ce chapitre se termine par un récapitulatif des objectifs visés. En termes de réglage, pour les systèmes haptiques il y a plusieurs objectifs à assurer. Le premier objectif est de garantir la stabilité du système dans toutes les situations et tous les cas de fonctionnement. En plus, dans

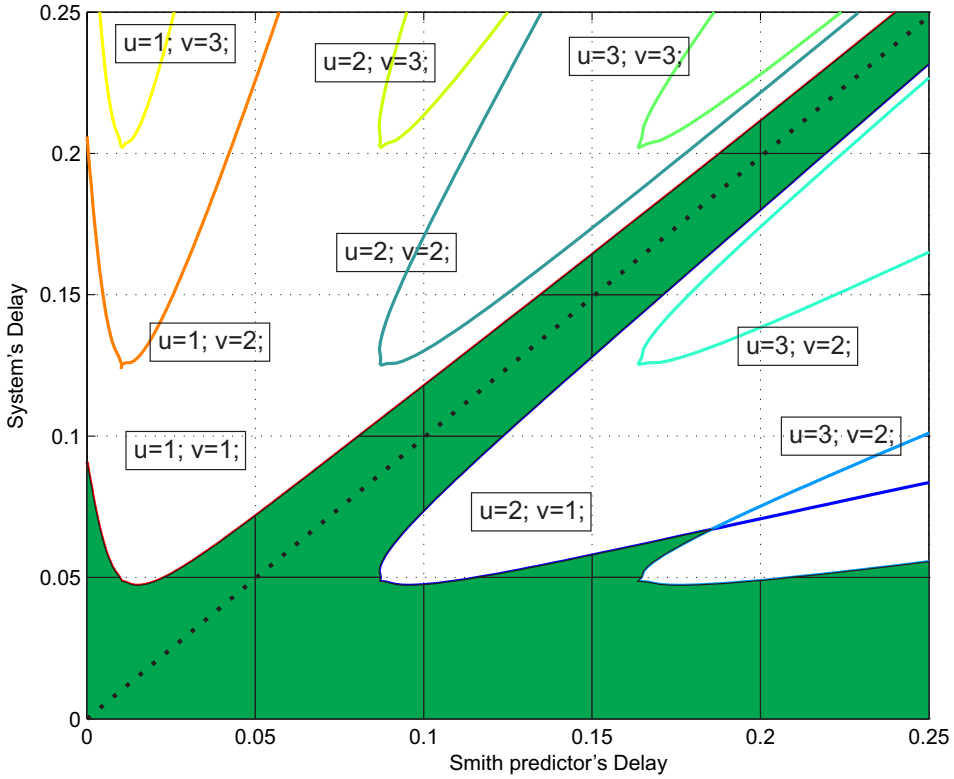


FIGURE 7: Zone de stabilité (en vert) dans l'espace des retards.

l'hypothèse où l'utilisateur ne déplace plus l'interface haptique (la vitesse est nulle et la position est constante), il faut atteindre la convergence asymptotique vers la vitesse nulle de l'objet virtuel et la coordination en position (entre l'interface haptique et l'objet virtuel).

Le deuxième objectif est de garantir une erreur de suivi la plus faible possible entre l'interface haptique et l'objet virtuel. Les deux cas de fonctionnement (mouvement libre et restreint) doivent être pris en compte et correctement analysés.

Le troisième objectif, le plus restrictif dans la majorité des cas, est la contrainte de transparence. Pour les deux systèmes - haptique et téléopération, la transparence caractérise la sensation de l'utilisateur final d'interagir directement avec la réalité virtuelle ou l'environnement à distance sans que le système haptique n'introduise de gêne. Sur la base des deux cas de fonctionnement - mouvement libre et restreint, le niveau de transparence doit être réalisé pour chaque cas selon des critères différents.

En termes de sécurité, une attention importante doit être apportée au retour d'effort maximal. Plus précisément, un retour trop fort pourrait entraîner des actions néfastes pour l'opérateur humain. Par exemple, dans un contexte d'environnements virtuels, la force calculée peut parfois provoquer des chocs élevés ou des forces continues, susceptibles de blesser l'opérateur humain. Avec ces considérations, le retour d'effort, ainsi que la vitesse, doivent être limités afin d'éviter d'éventuels accidents.

Chapitre 3

Le Chapitre 3 propose une nouvelle approche de commande issue du prédicteur de Smith [174], et utilisant un retour additionnel - la distance jusqu'à une possible collision. En règle générale, le prédicteur de Smith fonctionne correctement lorsque les retards sont fixes et connus, et lorsque le modèle utilisé dans le prédicteur est proche du système réel. L'idée centrale est ici d'utiliser un prédicteur uniquement du côté haptique, et ce afin de compenser l'effet de la viscosité et de fournir un sentiment réaliste en cas de contacts. Deux points principaux doivent être soigneusement analysés lors de l'utilisation de cette solution. Le premier est la variation du retard dans le temps. Le deuxième point est la cohérence du modèle utilisé au sein du prédicteur de Smith. Plus précisément, un modèle incompatible conduit à une prédiction incorrecte, ce qui peut résulter en une diminution des performances, ou même une perte de la stabilité. On peut montrer que le prédicteur de Smith a une certaine tolérance à certaines incohérences du modèle, mais il reste encore vulnérable [44, 77, 95].

Dans les cas de la téléopération et de l'haptique, le modèle change complètement entre mouvement libre et restreint. Au chapitre 2, l'analyse de la stabilité du prédicteur de Smith utilisé en haptique montre que de bons résultats peuvent être obtenus pour différents types de distributions des retards. Le système est stable en mouvement libre et également en état stationnaire en mouvement restreint, mais le problème vient de la phase de transition. La difficulté principale est liée au modèle considéré dans le prédicteur de Smith, qui n'est plus cohérent avec le modèle de la réalité virtuelle en présence de contacts. Malheureusement, la forme de base du prédicteur de Smith peut compenser les retards uniquement en mouvement libre, et en mouvement restreint le modèle n'est plus valable, dès lors que l'environnement virtuel n'est pas inclus dans le modèle du prédicteur, qui va induire une sensation d'impact faible et, dans certains cas, un comportement oscillant. La figure 8 illustre le comportement du système lors du passage de mouvement libre à mouvement restreint en utilisant la forme de base du prédicteur de Smith.

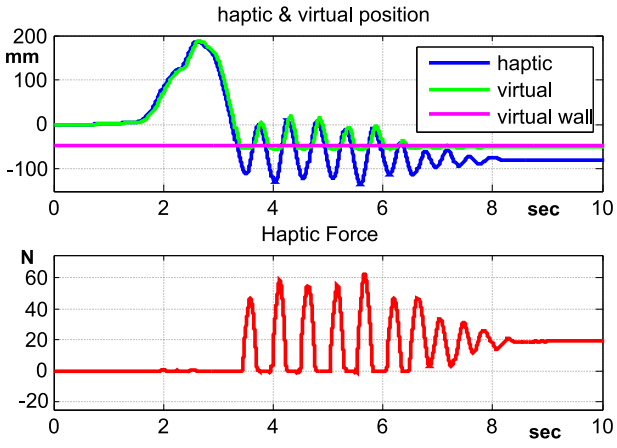


FIGURE 8: Comportement oscillant du système lors du passage de mouvement libre au mouvement contraint, en utilisant la structure classique du prédicteur de Smith.

Avec ces considérations, la solution intuitive est de trouver un moyen d'estimer la force de l'environnement et de l'inclure dans le prédicteur de Smith. Sur la base des informations reçues de l'environnement virtuel, il est possible de prédire le moment de l'impact et de maintenir la cohérence du modèle prédit, obtenant ainsi un système précis. L'idée principale de la solution proposée est d'introduire dans le prédicteur de Smith une estimation de la force de l'environnement F_{es} en utilisant la distance entre l'objet virtuel piloté et d'autres objets dans la scène. La figure 9 présente cette approche.

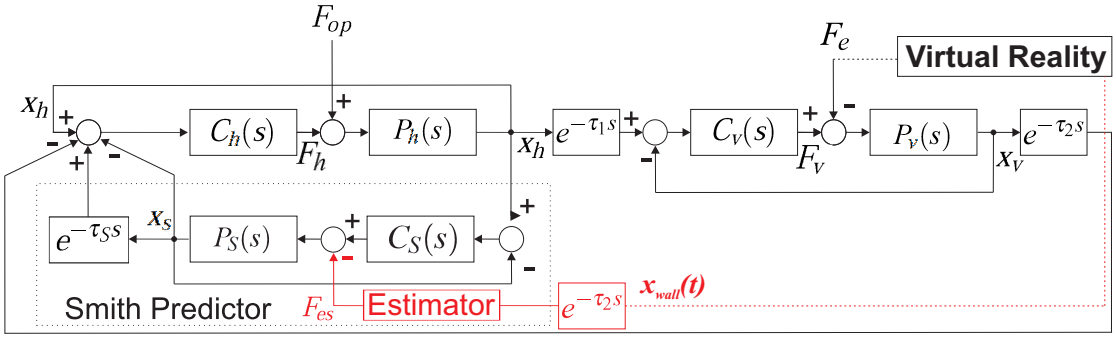


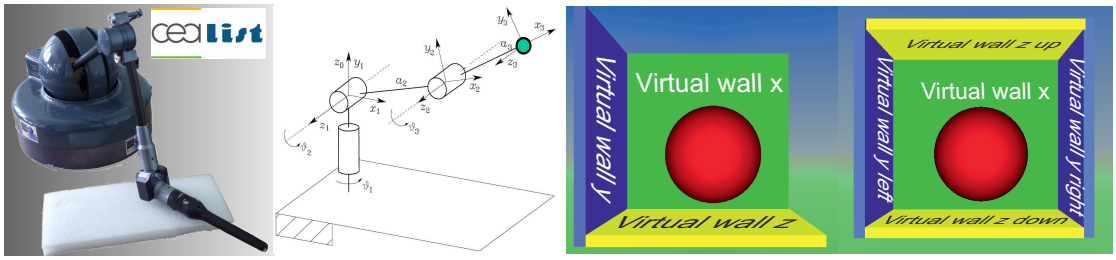
FIGURE 9: Schème du contrôle pour haptique avec prédicteur de Smith et retour en position.

La méthode proposée ici constitue une solution fiable pour la mise à jour du modèle utilisé dans le prédicteur, en prenant en compte les informations sur la distance jusqu'à collisions, disponibles dans la réalité virtuelle. Puisque le problème majeur du système de commande par prédicteur de Smith était l'incohérence du modèle, avec les informations reçues de l'environnement virtuel, il devient possible de prédire le moment de l'impact et d'avoir un modèle cohérent dans la prédiction, obtenant un système précis.

Pour tester cette méthode, une plateforme expérimentale à trois degrés de liberté est utilisée. L'objectif principal de cette plateforme est de garantir la maitrise des retards de communication et du temps de traitement de la réalité virtuelle. Pour cela, tous les algorithmes de commande (pour l'interface haptique/l'objet virtuel) et les simulations d'environnements virtuels sont exécutés sur le même ordinateur.

L'interface haptique, figures 10.a et 10.b, se compose de trois moteurs et trois codeurs optiques en quadrature, avec 1000 pts/tour (avec un réducteur de 1/10). Les lois de commande et la simulation virtuelle sont calculées en temps réel (système d'exploitation RTAI Linux) avec un temps d'échantillonnage de 1 ms. Les figures 10.c et 10.d illustrent les deux scènes virtuelles (environnement simple / boîte virtuelle) qui seront considérées pour les expériences, ainsi que l'objet virtuel. L'objet virtuel est modélisé comme une masse sphérique (égale à la masse de l'interface haptique) ($M_h = M_v$).

Afin de réaliser les essais de façon efficace et d'obtenir une analyse claire, différents scénarios, pour différents types de retards, sont proposés. Toutes les expériences sont réalisées par un



a.Interface Haptique. b. Géométrie du robot. c. Environnement *simple*. d. Boîte virtuelle.

FIGURE 10: Systéme haptique.

opérateur humain, ce qui explique pourquoi les conditions ne sont pas exactement les mêmes dans toutes les expériences. Les scénarios sont proposés à partir des situations simples ou complexes suivantes :

- mouvement libre (mouvements aléatoires sur chaque axe),
- mouvement restreint (contacts avec les murs sur chaque axe),
- contacts avec des objets en mouvement (mouvement sinusoïdal des murs virtuels),
- mouvement libre et restreint à l'intérieur d'une boîte virtuelle (mouvements aléatoires avec ou sans contacts sur chaque axe),
- mouvement libre et restreint à l'intérieur d'une boîte virtuelle avec des murs en mouvement sinusoïdal.
- contacts avec des objets en mouvement sous incertitudes aléatoires appliquées sur le retour en position,
- mouvement libre et restreint et transitions dans une boîte virtuelle avec des murs en mouvement sinusoïdal, sous incertitudes aléatoires appliquées sur le retour en position.

Les deux premiers scénarios rappellent les fonctionnalités de base des systèmes haptiques, en incluant cependant la méthode de commande proposée pour différents types de retards. Le troisième scénario est destiné à fournir une situation plus intéressante, lorsque la scène virtuelle n'est plus statique. Les quatrième et cinquième scénarios proposent une expérience plus difficile, puisque le niveau de complexité de l'environnement a été augmenté et des modifications rapides du modèle sont nécessaires. Le but des deux derniers scénarios est de tester la méthode proposée dans ce que nous considérons comme le cas le plus difficile du point de vue d'une perturbation, puisque l'estimation du modèle est fortement perturbée.

Sur la base de l'analyse effectuée pour le prédicteur de Smith pour les systèmes haptiques, présentée au chapitre 2, cinq types de retards sont envisagés :

- retards fixes et connus,
- retards incertains : incertitudes constantes,
- retards incertains : distribution uniforme,
- retards incertains : distribution gamma avec gap,
- retards incertains : distribution normale.

Le réglage du correcteur doit être réalisé en tenant compte du cas idéal, qui fournit les meilleures performances possibles en termes de perception de l'utilisateur final, ainsi que des limites de stabilité de chaque cas. Ainsi, l'idée de base est de trouver les gains les plus proches du cas idéal, tout en respectant le domaine de stabilité pour chaque cas.

Les exemples ci-dessous expliquent la démarche proposée. La figure 11 présente les résultats expérimentaux obtenus en mouvement libre avec des retards distribués selon une loi uniforme.

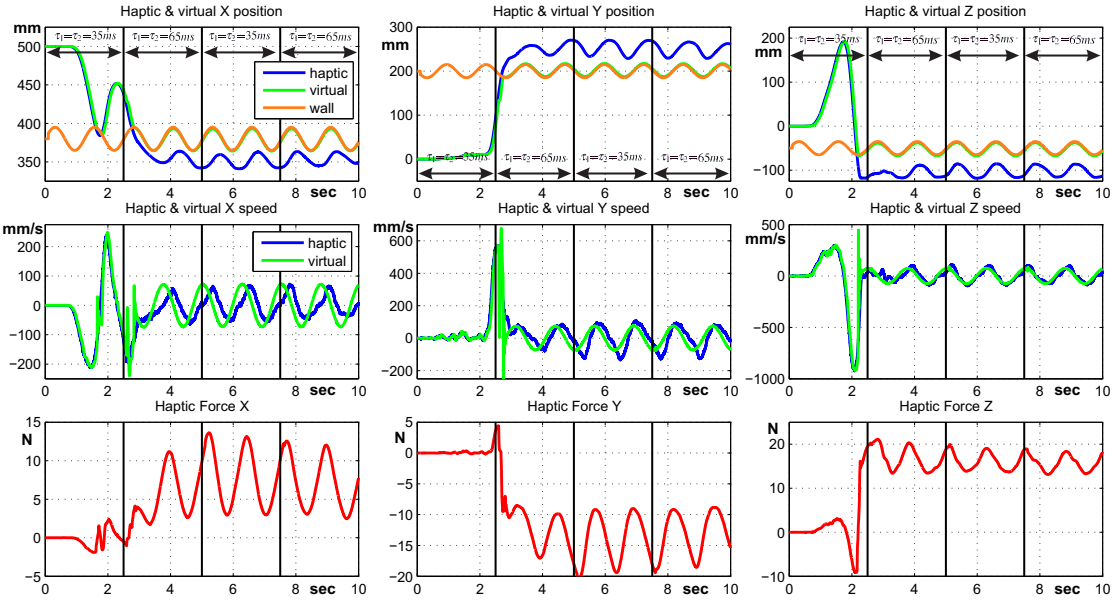


FIGURE 11: Mouvement libre avec retards distribués selon une loi uniforme.

On peut noter que, lorsque le retard utilisé dans le prédicteur de Smith est plus grand que le retard du système, il apparaît une avance de phase, résultant en une compensation de masse et de frottement. Ce phénomène peut être remarqué figure 11 correspondant à l'intervalle $\tau_1 = \tau_2 = 35ms$, alors que la force et la vitesse agissent dans le même sens, tandis que pour l'autre intervalle $\tau_1 = \tau_2 = 65ms$ l'effort et la vitesse sont en sens opposé. Les performances globales respectent les exigences souhaitées, l'effet de la viscosité est faible ($< 5N$, à l'exception de pics résultant des changements de sens à vitesses élevées) et le système est agréable à manipuler.

Le deuxième exemple choisi présente les résultats obtenus en cas de contacts avec des objets mobiles en présence de retards distribués selon une loi gamma avec gap, voir figure 12.

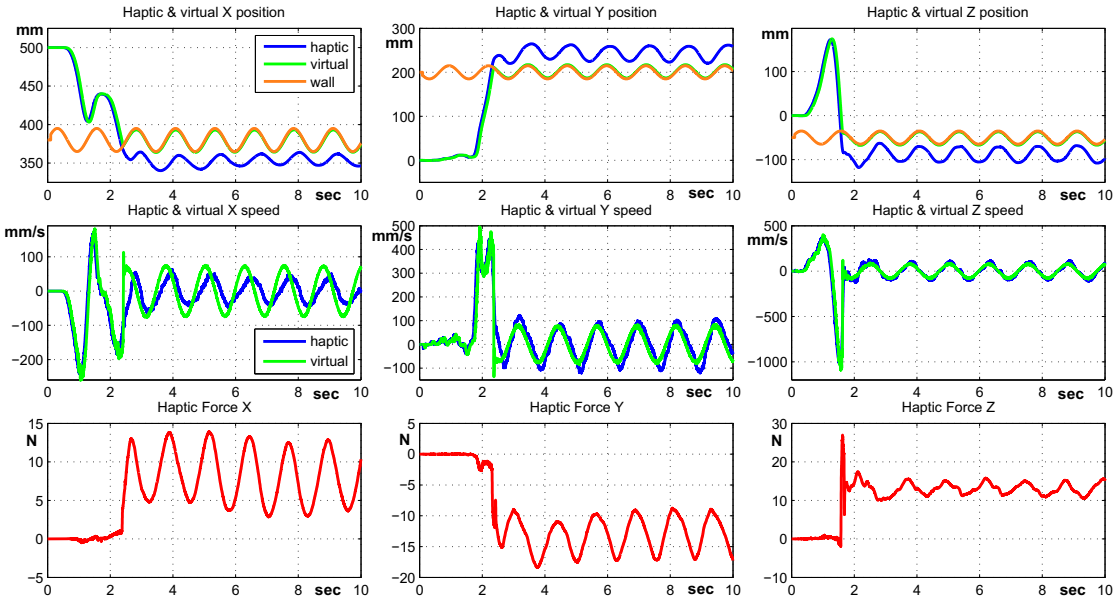


FIGURE 12: Contacts avec des objets mobiles en présence de retards distribués selon une loi gamma avec gap.

Dans ce scénario, l'objectif principal est de tester la capacité de la méthode à reproduire les déplacements des objets virtuels via l'interface haptique. Plus précisément, lorsque la force est maintenue, les oscillations suivent les mouvements exacts de l'objet virtuel, dans ce cas, la trajectoire sinusoïdale (voir figure 12, sur tous les axes à partir de $\approx 2\ s$, jusqu'à la fin). Le moment d'impact est reproduit correctement et d'une manière rigide, en fournissant à l'opérateur humain une manipulation agréable et réaliste.

En utilisant une boîte virtuelle pour l'environnement, le dernier exemple propose un scenario plus complexe et plus proche de la réalité, en présence de retards distribués selon une loi gamma avec gap, voir figure 13.

Ce scénario met en évidence les capacités de la méthode à fonctionner correctement dans des environnements dynamiques et complexes, avec commutations rapides de mouvements libres à restreints et vice-versa. Un deuxième objectif est de tester les impacts avec des objets mobiles, ainsi que la capacité de transmettre à l'utilisateur final les déplacements virtuels dans des domaines limités dans toutes les directions. Plus précisément, la force est maintenue sur les axes Y et Z à partir de $\approx 3,5$ à 7 secondes (la figure 13), et on peut remarquer que la force suit les oscillations des murs virtuels, et donc fournit à l'opérateur humain les déplacements sur les deux axes Y et Z. Les impacts sont correctement assurés, ainsi que les transitions entre mouvements.

Les résultats expérimentaux montrent que l'utilisation de cette nouvelle approche augmente les performances du système, lequel devient plus précis avec un niveau de transparence garanti dans toutes les situations considérées.

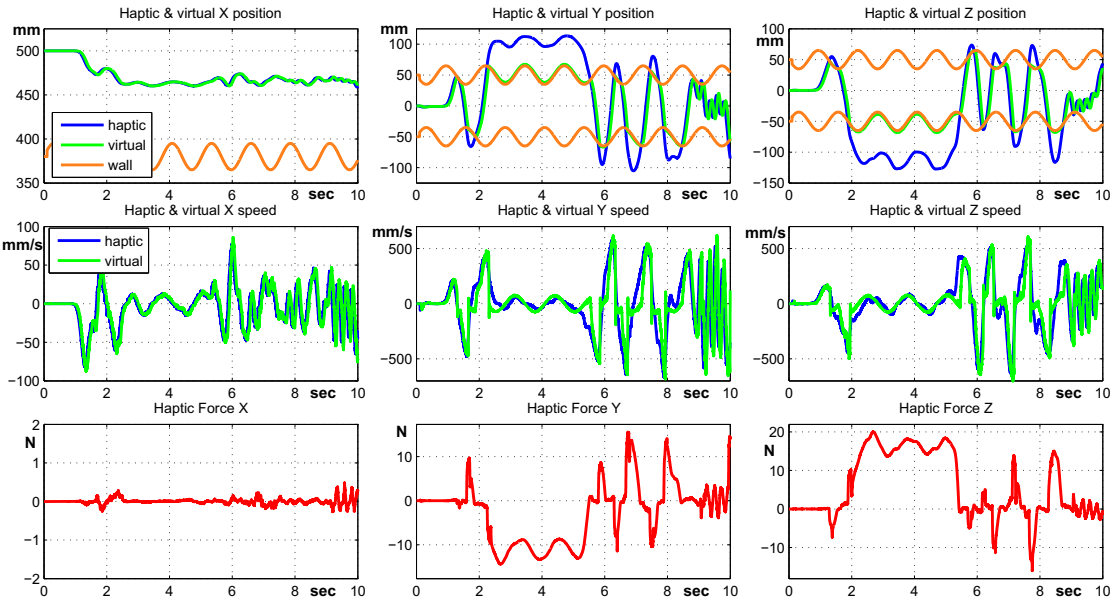


FIGURE 13: Mouvements aléatoire à l'intérieur d'une boîte virtuelle avec retards distribués selon une loi gamma avec gap.

En conclusion générale, dans le cas de retards incertains, une attention particulière est requise pour ce qui concerne le comportement et la forme du retard, afin de le modéliser de la façon la plus proche possible de la réalité, ce qui permettra ensuite de choisir plus efficacement les paramètres du correcteur. Le choix des gains du correcteur peut être plus ou moins vatse selon le type de distribution. Ainsi, le plus grand choix de paramètres, par rapport au cas idéal, est obtenu pour la distribution normale, tandis que pour la distribution uniforme, le choix est le plus restrictif (zone de stabilité plus pénalisante). Une solution intermédiaire est représentée par la distribution gamma avec gap, qui peut être appliquée si la forme du retard respecte les conditions nécessaires (moins restrictive que la distribution normale, mais plus précise que la distribution uniforme).

Chapitre 4

Le quatrième chapitre propose une approche PD avec séquencement de gains déterminés en fonction de la distance jusqu'à une éventuelle collision. La plupart des méthodes proposées dans la littérature fournissent des garanties supplémentaires de stabilité assez similaires, mais uniquement en termes de perception de l'utilisateur final (transparence). Afin d'améliorer les performances par rapport à l'utilisateur final, des valeurs différentes pour les gains du régulateur doivent être utilisées. Comme le régulateur PD classique représente la base de toutes les méthodes, celui-ci a été choisi comme structure de départ pour la mise en œuvre de la stratégie à gain variable. Le premier chapitre a souligné que des performances intéressantes en mouvement libre

peuvent être obtenues en utilisant des gains faibles, tandis que pour de bonnes performances en mouvement restreint, des gains élevés sont nécessaires. L'idée de base est alors d'utiliser un gain faible en mouvement libre afin de diminuer l'effet de viscosité et un gain plus fort pour le mouvement restreint pour obtenir l'impact raide souhaité. Le changement entre les deux gains est réalisé en fonction de la distance entre l'objet virtuel manipulé et les autres objets virtuels présents dans la scène.

Les travaux se sont donc orientés vers une approche par séquencement des gains. Beaucoup de notions différentes peuvent se retrouver sous le terme de séquencement des gains. L'idée principale est de passer ou de mélanger les valeurs de gains des régulateurs ou des modèles en fonction de différentes conditions de fonctionnement [97]. Dans la littérature, de nombreuses techniques de mise au point peuvent être trouvées [18, 21].

Lorsque la méthode de séquencement des gains est utilisée, la décision la plus difficile à prendre est le moment de la commutation et la manière de commuter. Dans la plupart des cas, toutes les stratégies reviennent à l'approche LPV (paramètre linéaire variant), pour plus de détails, voir [26, 34, 108]. En d'autres termes, sur la base de la valeur d'un paramètre, les correcteurs et/ou les modèles sont modifiés en temps réel.

La stratégie ci-dessous étend aux systèmes haptiques les concepts de base de la commande par séquencement des gains. En effet, l'environnement virtuel fournit des informations supplémentaires qui peuvent être utilisées dans l'algorithme de commande. Plus précisément, comme dans le cas du prédicteur de Smith avec retour en position, la distance entre l'objet virtuel piloté et un autre objet virtuel de la scène est utilisée pour améliorer les performances du système. Le principe est de passer du gain utilisé en mouvement libre au gain utilisé en mouvements restreints et vice-versa. Le changement doit être totalement transparent pour l'opérateur humain. Le changement de gain est réalisé en deux étapes :

- dans un premier temps, le gain est changé du côté virtuel,
- dans un deuxième temps, après un retard τ_2, le gain est mis à jour du côté haptique.

La figure 14 présente l'approche.

Dans le cas de mouvement libre, un gain K_p faible est utilisé afin de garantir un effet de viscosité réduit, sans perdre la coordination en position et vitesse entre l'interface haptique et l'objet virtuel contrôlé. Pour le mouvement restreint, un gain K_p important est considéré afin de conférer une réponse ferme à l'opérateur humain.

Comme déjà mentionné, le gain faible - K_{p_min} doit être choisi aussi petit que possible afin d'obtenir l'erreur de suivi souhaitée et de respecter la zone de stabilité en présence des retards (fixes ou aléatoires). Ensuite, le gain fort - K_{p_max} doit être choisi aussi grand que possible en

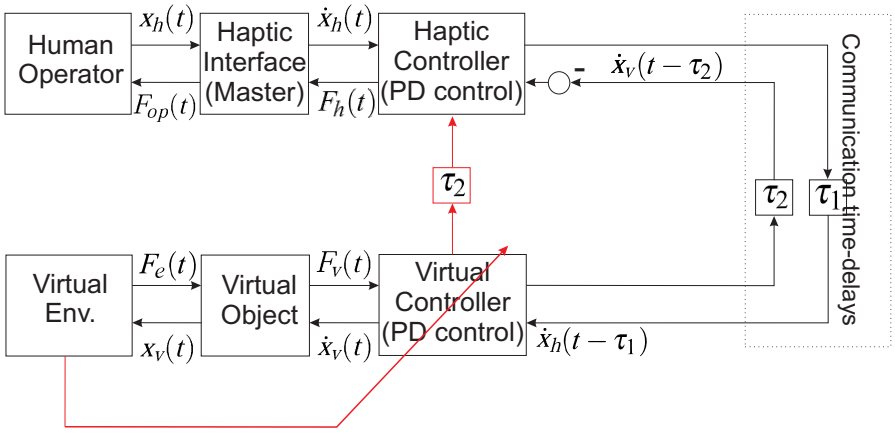


FIGURE 14: Système de contrôle pour PD avec séquencement de gain en fonction de la distance entre l'objet virtuel et l'objet le plus proche de collision de la scène virtuelle.

respectant la contrainte de stabilité. Si nécessaire, afin de remplir les conditions de stabilité, le gain K_d pourrait également être commuté entre deux valeurs correspondant aux mouvements libre et restreint respectivement.

La partie la plus difficile de cette solution est la partie liée à la commutation, plusieurs conditions doivent être remplies simultanément :

- Lors du passage du gain faible au gain fort, une transition transparente doit être atteinte qui fournit également la réponse raide souhaitée.
- Lors du passage du gain fort au gain faible, l'effet de collage doit être minimisé. Cet effet est la conséquence de la manipulation en mouvement libre utilisant le gain fort jusqu'à ce que la mise à jour soit réalisée.
- En cas de changement de direction pendant la phase de transition, le système doit être capable de réagir de manière souple et transparente.

La figure 15 présente l'organigramme de l'algorithme.

Sur la base de ces impératifs, le paramètre de commutation a été réglé expérimentalement afin d'obtenir les meilleures performances dans toutes les situations. Les gains sont mis à jour sur chaque axe indépendamment. Par exemple, lors d'une collision sur l'axe X, un gain élevé sera implanté uniquement pour cet axe, les gains faible seront maintenus pour les autres axes.

Cette approche - PD avec séquencement de gain, est testée expérimentalement et analysée pour différents scénarios et pour différents types de retards (fixes ou incertains). Comme dans le cas du prédicteur de Smith avec retour en position, toutes les expériences sont réalisées par un opérateur humain, ce qui explique pourquoi les conditions ne sont pas exactement les mêmes dans toutes les expériences.

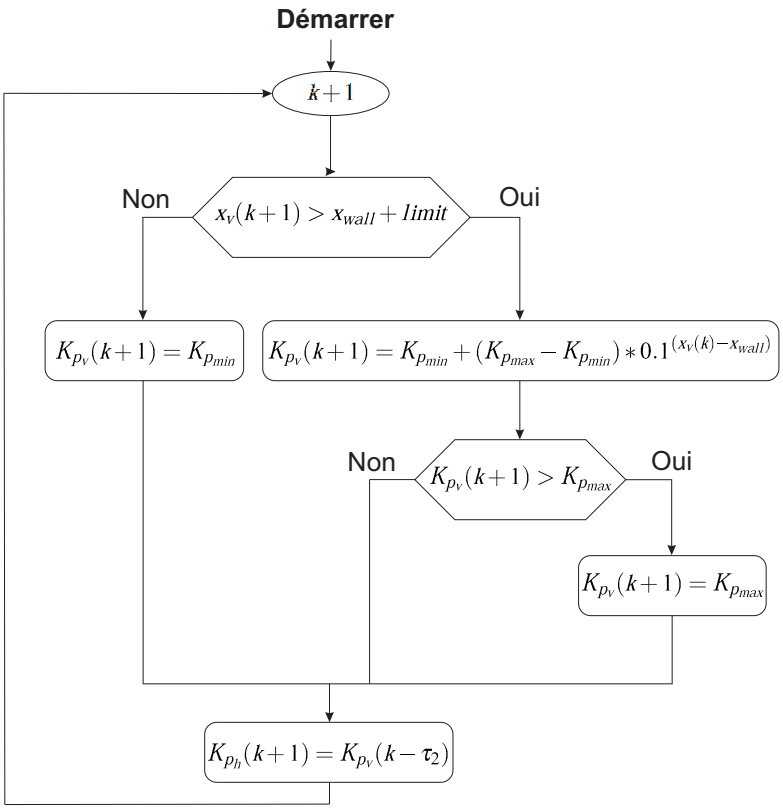


FIGURE 15: Organigramme de l'algorithme.

Dans le même esprit que dans le chapitre 3, afin de créer un moyen efficace de tester la méthode proposée, plusieurs scénarios ont été définis :

- mouvement libre (mouvements aléatoires sur chaque axe),
- mouvement restreint maintenu (contacts continus sur chaque axe pendant plus de 5 s) et retour en mouvement libre,
- mouvements aléatoires à l'intérieur d'une boîte virtuelle (mouvements aléatoires avec ou sans contacts sur chaque axe)

Les deux premiers scénarios reprennent les fonctionnalités de base des systèmes haptiques, en utilisant la méthode de commande proposée pour différents types de retards. Le dernier scénario propose une expérience plus réaliste, puisque le niveau de complexité de l'environnement est augmenté et des changements rapides de gain sont nécessaires. Le but est de tester la méthode proposée pour des changements rapides entre mouvements libres et restreints dans des espaces étroits.

En se basant sur l'analyse de la commande PD effectuée pour les systèmes haptiques, présentée dans le deuxième chapitre, et compte tenu des types de retards analysés, trois cas de retards sont envisagés pour chaque scénario :

- retards constants,
- retards incertains : distribution uniforme,
- retards incertains : distribution gamma avec gap.

Le premier exemple considéré montre les mouvements libres dans le cas de retards constants, figure 16.

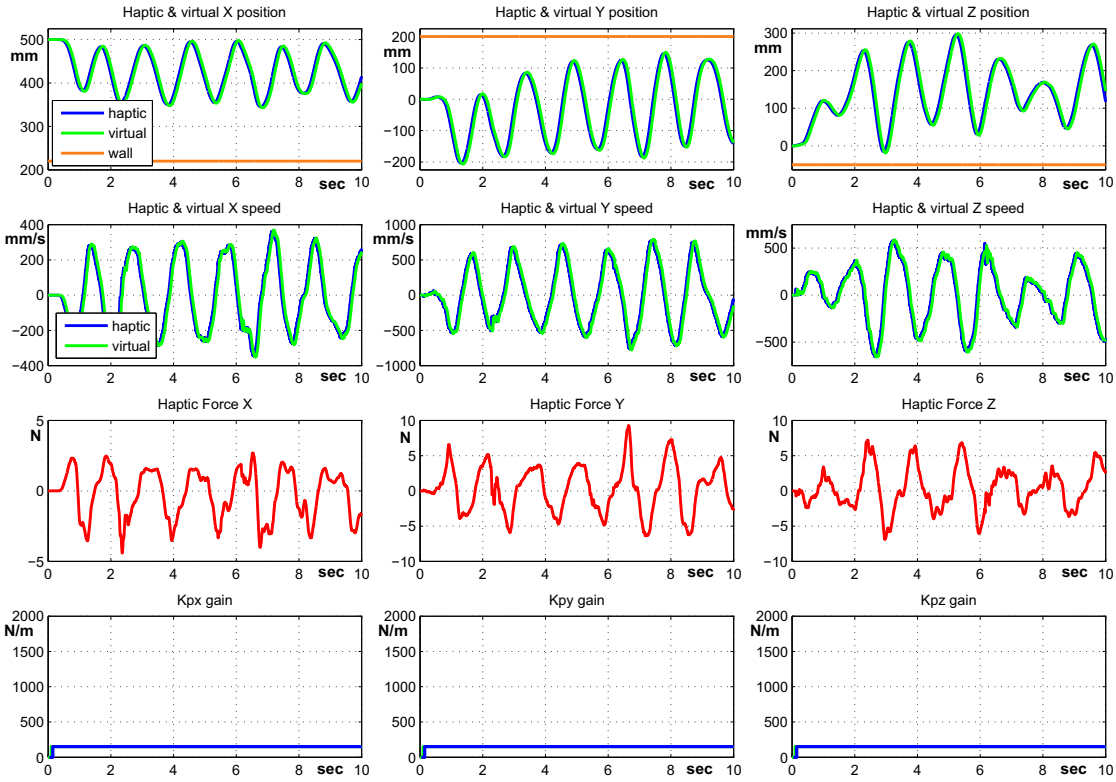


FIGURE 16: Mouvements libres dans le cas de retards constants.

On peut remarquer que l'effet de la viscosité est faible (<5 N, à l'exception des pics dus aux changements de direction à vitesses élevées), offrant une manipulation agréable. Comme il n'y a pas de changement de mouvement, le gain faible est maintenu pendant toute la durée de l'expérience.

Ensuite, afin de tester la méthode dans le cas de mouvements restreints, l'exemple de la figure 17 introduit cette situation en présence de retards distribués selon une loi uniforme.

Dans cet exemple, les gains K_p sont commutés en fonction du type de mouvement - libre ou restreint. Comme la figure le montre, le sentiment d'impact est correctement reproduit (la force augmente rapidement), et l'utilisateur peut ressentir les contacts d'une manière rigide. En revenant de mouvement restreint à libre, l'effet de collage se fait ressentir pendant une période

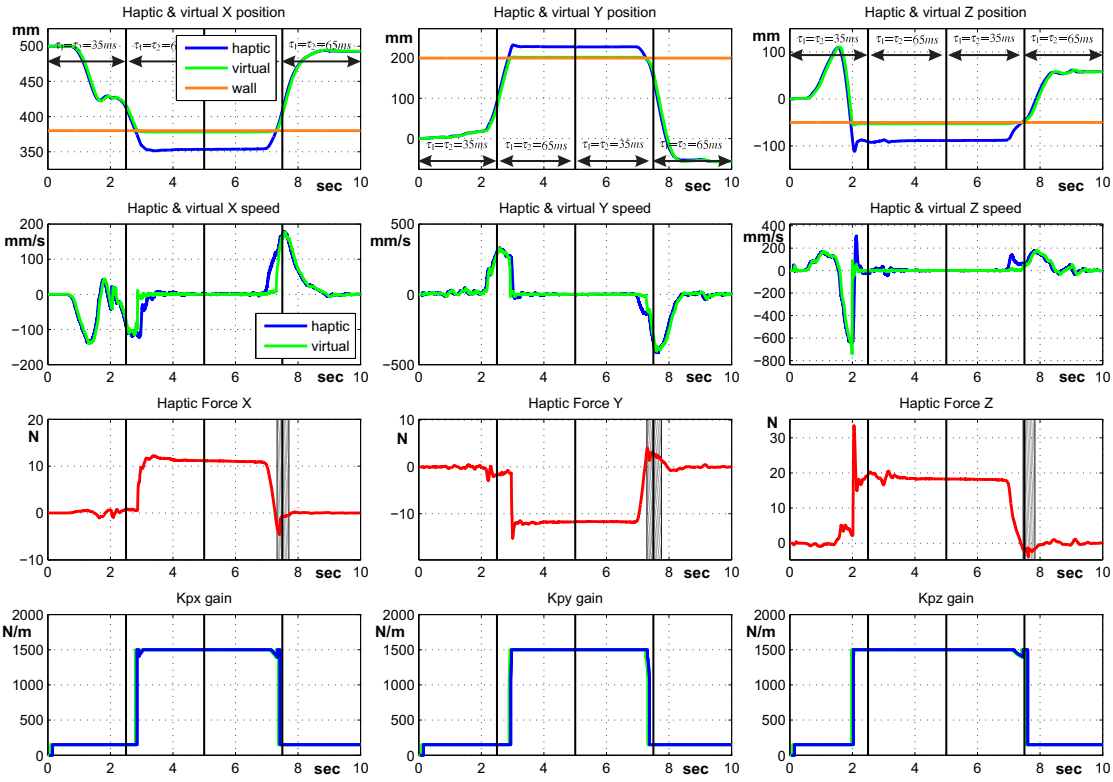


FIGURE 17: Mouvements restreints en présence de retards distribués selon une loi uniforme.

de temps limitée (zone ombrée des graphiques concernant la force sur chaque axe), sans inconvénients particuliers. Le comportement global du système offre un sentiment réaliste, ainsi qu'une manipulation agréable.

Le dernier exemple propose des mouvements aléatoires (avec ou sans contacts) à l'intérieur d'une boîte virtuelle en présence de retards distribués selon une loi gamma avec gap, figure 18.

Le but de ce scénario est de tester la méthode proposée en configuration de commutations rapides (entre mouvements libres et restreints). Les situations de mouvement restreint peuvent être vues sur les graphiques par le dépassement des limites des murs virtuels sur chaque axe. Comme on peut le constater, il existe de nombreux changements sur les gains K_p, et en particulier, des changements au dernier moment, qui ne sont pas parfaitement gérés. Plus précisément, en raison de ces commutations rapides, les effets ressentis par l'opérateur humain ne sont pas très clairs. De temps à autre surviennent des augmentations non naturelles de l'effet de viscosité, tandis que dans d'autres cas l'impact est ressenti avec un manque de raideur.

En résumé, comme il existe deux situations de fonctionnement, et implicitement deux réglages correspondants, l'approche par séquencement des gains semble être appropriée pour une telle situation. La méthode proposée améliore les performances globales du système pour des situations impliquant des transitions claires (de mouvement libre à mouvement restreint ou vice versa

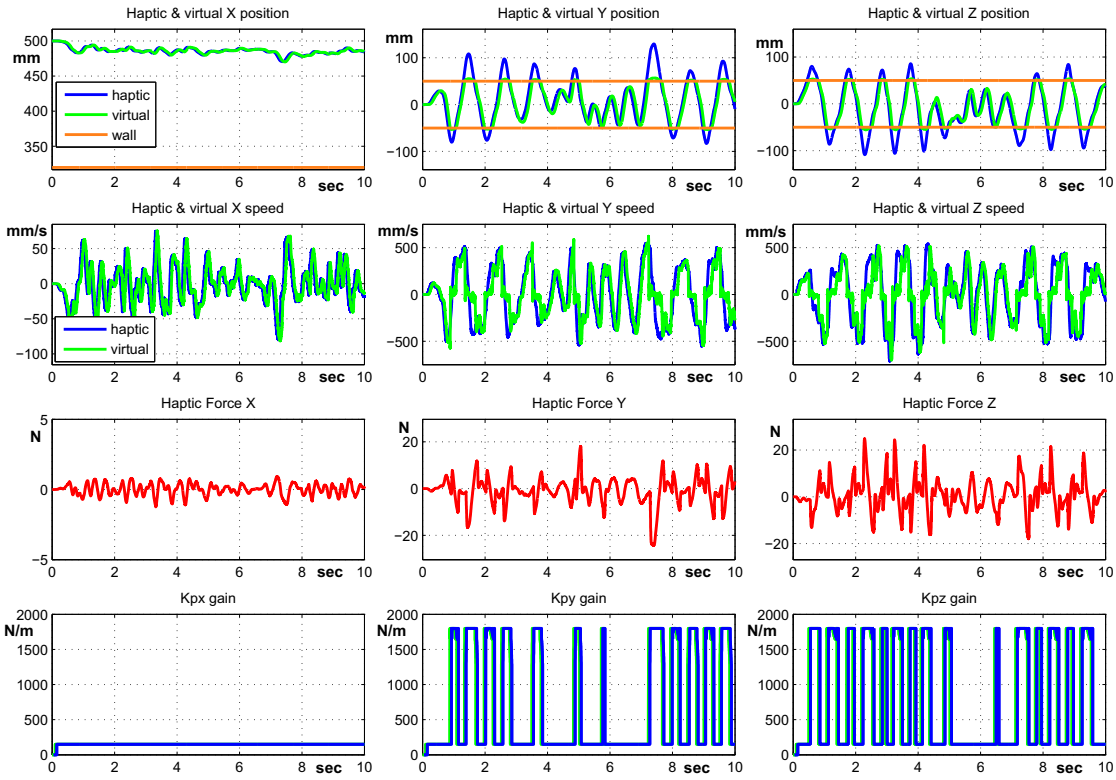


FIGURE 18: Mouvements aléatoires à l'intérieur d'une boîte virtuelle en présence de retards distribués selon une loi gamma avec gap.

à basses fréquences). Les expériences réalisées, pour un retard global de 100 *ms*, soulignent les améliorations, ainsi que les limites de la méthode dans les cas de commutations rapides.

En général, la méthode fonctionne correctement pour des retards de communication relativement petits ($< 20ms$), alors que pour des retards plus importants (> 25 *ms*), il apparaît une perte de performances, en particulier lors de changements rapides entre les deux modes de fonctionnement. Une solution possible pour l'amélioration de la méthode serait la mise en œuvre d'un algorithme de gain plus complexe, basé sur la vitesse du mouvement et les valeurs des retards. Plus précisément, pour une vitesse importante, la limite de commutation doit être augmentée afin de modifier rapidement le gain, tandis que pour une vitesse faible la limite doit être réduite afin que le processus de mise à jour soit terminé des deux côtés au moment de l'impact. Ceci pourrait être formalisé également comme un problème d'optimisation pour trouver la limite optimale en fonction de la vitesse et de la valeur du retard, afin de terminer le processus de mise à jour à l'instant de l'impact. Un autre point qui peut être ajouté, afin de diminuer l'effet de collage qui apparaît lors du passage de mouvement restreint à mouvement libre, est de prendre en compte le sens du déplacement, afin de diminuer le gain côté haptique avant la réception des informations de la commande virtuelle. Une stratégie dynamique de changement pour chaque axe en fonction de la vitesse, de la valeur du retard et du sens peut réduire sensiblement les limites de la méthode.

Conclusions et perspectives

L'objectif principal de cette thèse est d'apporter de nouvelles solutions afin d'améliorer les performances en termes de perception de l'utilisateur final pour les systèmes haptiques affectés par des retards de communication. Comme mentionné précédemment, les principaux problèmes des systèmes haptiques en présence de retards sont d'une part l'effet de viscosité en mouvement libre, et d'autre part la réponse non rigide en cas de contacts.

La première étape de ce travail a consisté à analyser et tester expérimentalement les méthodes de commande les plus couramment utilisées en haptique. Les résultats expérimentaux obtenus sur une plateforme haptique à un degré de liberté ont révélé que les principaux objectifs du système haptique (viscosité faible en mouvement libre et réponse raide en cas de contacts) ne peuvent pas être assurés simultanément par les méthodes présentées. Les résultats ont montré que de bonnes performances peuvent être obtenues soit en mouvement libre, soit en mouvement restreint, avec une perte de performances dans l'autre cas.

Ensuite, les outils théoriques d'analyse de stabilité des systèmes à retards dans différentes configurations, ainsi que les contraintes physiques et les conseils de réglage d'un point de vue pratique ont été présentés. Puisque le régulateur PD est la structure la plus couramment utilisée en haptique et en téléopération, une étude complète a été menée pour ce qui concerne la stabilité, ainsi que la fragilité des régulateurs PD pour les systèmes affectés par des retards. L'étude a été faite pour des retards fixes, ainsi que pour des incertitudes sur leurs valeurs, modélisées par les distributions uniforme et gamma avec gap. Il faut noter que, dans le cas de retards incertains, une attention particulière doit être portée à la formalisation du comportement du retard afin de choisir ensuite les gains du régulateur d'une manière plus efficace. Les résultats spécifiques pour le cas haptique, ainsi que quelques exemples généraux, ont été présentés à l'aide d'une représentation géométrique, dans le but de donner un aperçu clair sur la région de stabilité dans l'espace des paramètres du régulateur PD. Cette approche géométrique a permis de donner un traitement unitaire dans le cadre de cette thèse.

L'utilisation du prédicteur de Smith a été analysée dans deux directions distinctes : les régions de stabilité dans l'espace des gains du régulateur et l'espace des retards (retard du système et retard du prédicteur). Dans le premier cas, pour les retards fixes ou incertains, une région admissible pour la stabilité des gains du correcteur ($K_p,\ K_d$) a été élaborée. Pour le deuxième cas, une fois mis en place les paramètres de réglage du régulateur PD ($K_p,\ K_d$), une étude de la dépendance entre la variation du retard du système et du retard du prédicteur de Smith a été proposée. Une analyse spécifique de la commande se fondant sur le prédicteur de Smith dans le cas de retards incertains a été présentée en détail à partir du cas où les incertitudes sont fixées, et puis pour des distributions uniforme, gamma avec gap, et enfin, en utilisant une notion de stabilité moyenne, pour le cas de la distribution normale, le tout validé par des exemples du domaine de l'haptique.

Sur la base de l'analyse effectuée au premier chapitre et par rapport aux résultats théoriques du deuxième chapitre, deux nouvelles approches ont été proposées et testées sur un système haptique à trois degrés de liberté afin d'améliorer les performances en termes de perception de l'utilisateur final :

- La première méthode - prédicteur de Smith avec retour en position, utilise les informations provenant de l'environnement virtuel sur la distance jusqu'à une éventuelle collision afin de fournir un modèle plus précis au prédicteur. Il est important de noter que le prédicteur de Smith classique compense les retards uniquement en mouvement libre, tandis que pour le mouvement restreint le modèle n'est plus exact, engendrant des oscillations. Cette méthode a été testée à partir de situations simples (mouvement libre et restreint) puis plus complexes dans lesquelles des impacts avec des objets en mouvement et des boîtes virtuelles ont été examinées pour des retards fixes et incertains. Dans tous les cas, le prédicteur de Smith avec retour en position a assuré les performances souhaitées.

- La deuxième méthode proposée - PD avec séquencement des gains, s'envisage naturellement dès lors que de bonnes performances peuvent être obtenues dans un cas, en utilisant les gains adéquats, mais en perdant complètement les performances de l'autre cas. Comme mentionné précédemment, la méthode fonctionne correctement pour des retards relativement petits (< 20 *ms*), alors que pour des valeurs plus importantes (> 25 *ms*), il apparaît une perte de performances, en particulier lors de changements rapides.

  Parmi les deux méthodes proposées sous leur forme actuelle, les meilleurs résultats sont obtenus pour le prédicteur de Smith avec retour en position, car les performances du régulateur PD avec séquencement de gain sont fortement liées à la vitesse et à la valeur du retard. Un autre aspect est représenté par la perte de performances lors de commutations rapides entre mouvements libres et restreints. La raison qui peut expliquer la différence entre les deux méthodes est que le prédicteur de Smith avec retour en position agit comme un prédicteur (comme son nom l'indique), tandis que le PD avec séquencement de gain agit sur la base des actions présentes sans aucune anticipation.

Des perspectives permettant d'améliorer encore la commande à base de prédicteur de Smith avec retour en position seraient l'ajout d'un observateur de passivité ou d'une méthode de modulation de consigne, afin de fournir une garantie supplémentaire de stabilité en cas d'oscillations rapides de l'environnement virtuel ou de perturbations violentes sur le retour en position.

De plus, cette nouvelle stratégie pourrait être étendues aux systèmes haptiques à six degrés de liberté et des environnements virtuels plus complexes, soit concaves ou avec des formes géométriques non régulières. Les perspectives d'amélioration de la méthode PD avec séquencement de gain sont liées à l'utilisation d'un algorithme de changement de gains dynamique, basé sur la vitesse du mouvement et les valeurs des retards.

En termes d'applications, la prochaine étape pourrait être la mise en œuvre des algorithmes de commande pour des applications aux systèmes collaboratifs. Plus précisément, dans cette perspective, le système haptique a été conçu pour un seul opérateur humain, tandis que dans les développements futurs, les algorithmes seront mises en œuvre pour les systèmes avec plusieurs utilisateurs qui interagissent dans le même environnement virtuel. Certaines idées peuvent être trouvées dans [36] dans le cadre d'une l'étude d'un cas simple de prédicteur de Smith.

De plus, les algorithmes présentés ici peuvent être également mis en œuvre pour des applications de supervision utilisées dans des systèmes de téléopération, qui sont également affectés par des retards induisant des problèmes similaires à ceux abordés dans cette thèse. Les applications de supervision représentent une aide virtuelle pour l'opérateur humain dans la réalisation des tâches difficiles [43, 56].

Les plages de variations temporelles des retards analysés ici sont similaires à celles rencontrées lors d'applications via Internet, et ainsi, de futures applications seront testées directement sur Internet. Afin d'améliorer les algorithmes, des problématiques de pertes de paquets et des situations de dysfonctionnement du réseau seront prises en compte et mises en œuvre dans les stratégies de commande. De telles approches sont nécessaires lors de l'utilisation via Internet comme moyen de communication.

Abbreviations and acronyms

CAD	Computer-aided design
2D/3D	Two/Three-Dimensional space
PID(P, PI, PD)	Proportional-Integral-Derivative (controller)
1/3-DOF	One/Three Degree(s) of freedom
SISO	Single-Input and Single-Output
atan2	Computes the principal value of the argument function applied to the complex number $x + iy$.
VE	Virtual environment
VR	Virtual reality

General Introduction

General Context. Teleoperation & Haptic Systems

In the last decades, many new technologies have been developed and introduced in the industrial environment. This is the case of teleoperation systems and also haptic systems. According to [19], a teleoperation system enables human interaction with inaccessible environments to direct human contact due to their localization or dangerous circumstances. Roughly speaking, the goal of *teleoperation systems* is to replace the direct human manipulation. In this sense, these systems are used for working in harmful environments (like radioactivity or dust [88]), for high precision operations (like surgeries [182]), as well as for spatial operations which are made remotely (as discussed by [171]). Telesurgery and space telerobotics are two examples of more recent teleoperation applications involving long distance communication between master and slave units [28, 61, 167]. Recent interest in robot-assisted surgery [60, 139] pointed out many advantages, as, for example, minimal invasiveness [45], enhanced accuracy and dexterity [31, 169], and increased safety and reliability [147, 181]. Telesurgery takes this one step further by its potentials for providing access to expert medical care for a larger group of patients more effectively and cost efficiently. Haptic feedback has been shown to improve task performance during both surgical and space teleoperation applications [81, 187].

According to [53], the interest in teleoperated systems and the technology being developed for teleoperation is increasing. New technology such as telework, tele-assistance or e-learning has a great number of common objectives which favors the development of more advanced interfaces and new techniques for work and communication. Such actions are all extending the possibilities of remote presence. Essential development regarding the interface of teleoperation is concentrated on haptic and visual feedback [2, 82]. Other interesting points to consider are voice and gesture recognitions that establish dialogues [117, 148]. As a consequence of this, a great effort is made to improve communications between operator and environment, especially when using public networks such as Internet.

Figure 19 presents the teleoperation basic principle.

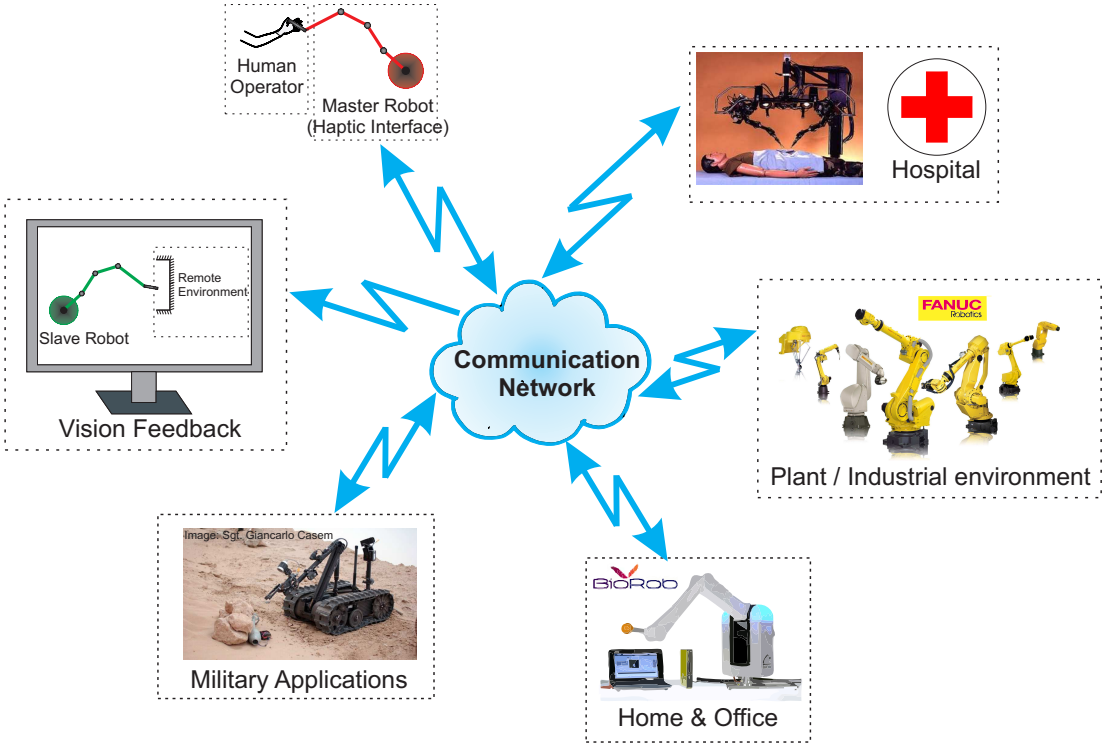


FIGURE 19: Teleoperation systems

Roughly speaking, a teleoperation system is composed of two robots, geographically differently disposed, working on the principle of master/slave. Generally, the human operator imposes a force and/or a position to the master robot which will transmit the command to the slave robot through the communication network. Depending on the situations encountered by the slave robot, the master should also receive information related to these situations via the feedback force (a complete survey on teleoperation systems can be found in [76]). Resuming, the basic idea of a teleoperated system is to enable the human operator to interact with remote environments providing the feeling of tele-presence[3].

From the control point of view, according to [186], the subject is very challenging, since the haptic control loop (where motion and force data are exchanged between the master and the slave manipulator) is closed over a communication network, e.g. the Internet. The communication network introduces unreliabilities such as (varying) time-delays and packet loss, which are not only degrading the human haptic perception of the remote environment, but can destabilize the overall system. In recent years, control approaches based on the passivity framework and the scattering transformation have been developed in order to stabilize the teleoperation system in the presence of such communication unreliabilities, see, for instance, [5, 73] and the references therein.

[3]the feeling of tele-presence is the feeling of the human end user of being present and acting directly in the environment, without using additional devices.

Numerous researchers have contributed to this field over the last decades. Since Ferrell defined the problem of bilateral kinesthetic teleoperation with *"delayed force feedback"* [51, 52], various strategies for stabilization of closed loop teleoperation with communication latency were suggested [5, 6, 130, 132, 140]. In addition, several approaches were proposed for achieving a high degree of transparency [96] with respect to exchange of force and position information in bilateral kinesthetic teleoperation, see, for instance, [3, 10, 66, 68, 78]. It is worth mentioning that the authors of [195] address the problem of the position/force tracking in teleoperation systems and proposes a haptic proxy control scheme, assuming that communication delays are both time-varying and asymmetric. Furthermore, Zhang *et al.* are proposing several solutions based on H_∞ control design for time-varying delays and polytopic-type uncertainties [197], for dealing with the problem of delay-dependent robust control for time-varying delay teleoperation system with norm-bounded and time-varying model uncertainties [198], and also the corresponding discretization problem in the context of teleoperation is addressed [196].

Haptic systems represent an *extension* of the teleoperation systems. A haptic system has only the master robot, which is usually called haptic interface. The second robot is replaced by a virtual robot/object which is working in a virtual environment. The principle is the same, the user should *feel* the contacts encountered by the virtual object/robot via the feedback force [177].

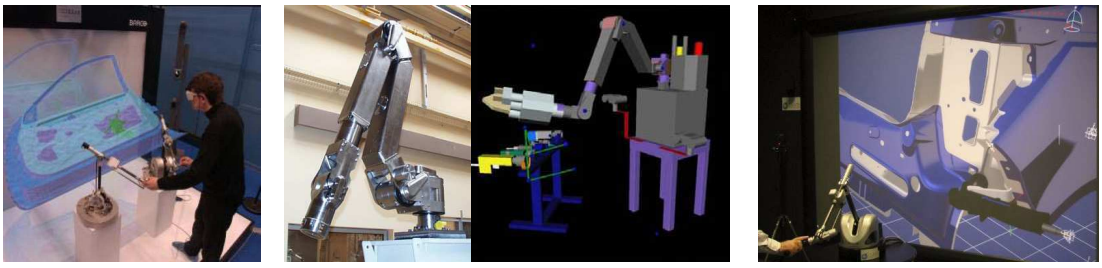
Haptics research is looking to recreate the touch *feeling* in a detailed way for users in virtual reality [93]. Ideally, interacting with a virtual or remote environment would be just as simple and vivid as using a hand tool or your own fingers. According to [93], the technology available today cannot yet meet this ambitious goal, so systems must be streamlined to contain only information that is the most important for the task at hand.

In order to provide feedback forces, it is always necessary to have a proper virtual model that a user can interact with. In graphics rendering, figures are rendered on a screen to give visual feedback to a user. Similarly, in haptics, computed interactive force between a user and a virtual model is rendered to the user via the haptic device. Such virtual models can be constructed from mathematical functions (implicit and parametric objects) or a collection of geometric primitives [87, 142]. According to [93], the most common way of interacting with a polygonal object is to use a virtual proxy, which always remains outside of the object in contact situations while trying to move closer to the user whenever possible [94]. This needs collision detection algorithms that monitor and regulate the movement of the proxy. Unlike visual feedback, haptic signals need to be updated at a high rate of 1 kHz or above in order to provide and preserve the *real* sensations to the user. To overcome the computational problems, haptic computations are usually separated from graphics and other slower computations. More details on such an approach can be found in [143].

According to [64], one of the most challenging problem in haptics is to create a control system which is simultaneously *stable* (i.e., does not exhibit vibration or divergent behavior) and gives

high fidelity under any operating conditions and for any virtual environment parameters. A *classic engineering "trade-off"* must be made since realism of the haptic interface (for example, in terms of stiffness of "hard" objects) must often be reduced in order to guarantee totally stable operation. Initial efforts to solve this problem introduced the "virtual coupling" between the virtual environment and the haptic device [40, 201]. The virtual coupling is a virtual mechanical system containing a combination of series and parallel elements interposed between the haptic interface and the virtual environment to limit the maximum or minimum impedance presented by the virtual environment in such a way as to guarantee stability. .

Virtual environments have become very popular and are used in many domains, like prototyping (figure 20.a example of prototyping using haptic interfaces and virtual environment [99]), trainings for different devices and assistance in completing difficult tasks (figure 20.b virtual environment used for task assistance/supervision [43, 56]), virtual assembling (figure 20.c - example of virtual assembly [7, 176]). Numerous early attempts of collaborative simulations were bent towards multi-user virtual environment (VE) applications. Thus, to cite only a few, SIMNET [30] and NPSNET [110] were real-time VE prototypes for military training. They used dead-reckoning[4] method to give the same scenes (or state synchronization) of the simulated environment to the participants as much as possible. Some multi-user virtual reality systems have incorporated *haptic interfaces* to allow human's direct mechanical interactions. Basdogan *et al.* [22] developed a collaborative game using haptic interaction, and Hudson *et al.*[79] established a nanoManipulator system with atomic force microscopes. They used semi-optimistic concurrency control[5] to improve responsiveness at the client side. However, since there was no treatment for state synchronization, the two users suffered from de-synchronized visualization, and as a result the success rate of the collaborative task was fairly limited. The simulation was implemented in peer-to-peer network configuration to maximize responsiveness.



a. Virtual Prototyping. b. Virtual Assistance/Supervision. c. Virtual Assembly.

FIGURE 20: Examples of Virtual Environments Applications

[4]Method used in navigation, representing the process of calculating the current position by using a previously determined position, based on known or estimated speeds over elapsed time, and course.

[5]Concurrency control guaranties correct results for concurrent operations, while getting the results as fast as possible. Semi-optimistic - the operations are block only in some situations (rules violation), while for the rest of the cases, the optimistic strategy is applied (delaying rules checking until the operation ends) [23].

A major control problem of such systems is the presence of *time-delays*. Large delays are widely considered since several years, aiming at *natural* teleoperation/haptics over the Internet. The presence of time-delays may induce instabilities and complex behaviors, and such topics have been largely treated in the literature (see, for instance, [57, 98, 129, 199] and the references therein). Excellent overview of some existing results, as well as some open problems can be found in [153]. Systems with distributed delays are present in many scientific fields such as economy [86], population dynamics [42], traffic control [83], network/Internet-based systems [188, 193].

Time-delay often arises in feedback control systems due to the acquisition of response and excitation data, information transmission, on-line data processing, computation and application of control forces. In spite of the efforts to minimize time delays, they cannot be totally eliminated, even with today's advanced technology, due to physical limits. The information delay is often negligible but, for some cases, it may still be crucial. There exists an abundant literature on control for time-delay systems (see, for instance, [90, 91, 138, 151, 168] and the references therein).

Similar to the teleoperation, also in haptics, the problem of time-delays and their influence on the system's performances remains open. Speaking about haptic systems over the Internet, which are nowadays much more frequently used, the problems encountered in haptic control are very similar. There exist two time-delay sources: the communication channel and the processing time for the virtual reality environment. For complex virtual environments, the processing time can increase substantially and can introduce unwanted effects and unexpected behaviors [47].

In figure 21, the general scheme of a network-based haptic system is presented.

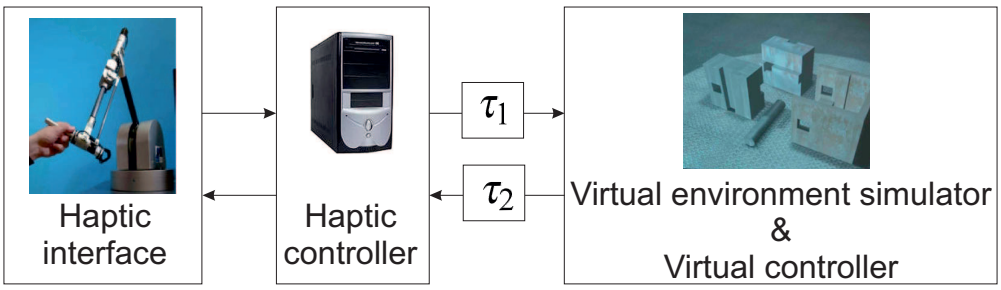


FIGURE 21: General Scheme of a Haptic System

An ideal haptic/teleoperation system must have:

- position tracking error as small as possible between the haptic interface and the virtual object (master and slave for teleoperation systems);

- high degree of transparency, i.e. in free motion, the force feedback felt at the haptic interface end must be as small as possible and in case of hard contact, a stiff response is desired.

More precisely, in free motion the delay effect can be felt by the viscosity phenomenon (high force feedback felt at the haptic interface end). Next, in the case of a hard contact with the environment, the impact effect will not be stiff, or the most unwanted situation is to loose the system stability due to the delays. Therefore, the delays must be taken into account and included in the control laws. However, a trade-off between stability, position tracking error and transparency must be always made.

The first step to solve this problem was to *port* the existing algorithms from teleoperation into the haptics context. More precisely, methods like Proportional Derivative (PD) with local dissipation [101], PD with passivity observer [14, 159, 160], PD with passive set-point modulation [100], wave scattering transform [130] and Smith predictor [36], have been appropriately *adapted* from teleoperation to haptics framework.

Since there exists a *large* number of algorithms and methods, in order to evaluate their performances, comparative studies for teleoperation systems as well as for haptic systems can be found in the literature, like [112, 154] or [104, 164], respectively. In our opinion, the second step will be to use the additional available data from the virtual environment, i.e. the information from the virtual environment about the distance between the objects, possible collisions and many other details regarding the system can be extracted in order to improve the control algorithms.

A reliable and simple method in controlling time-delay systems remains the *Smith predictor* [174]. Such an approach was quite *appealing* in various applications. More precisely, a Smith predictor-based control scheme for motion synchronization in virtual environments under large time-delays is presented in [35]. In teleoperation, among many solutions proposed for overcoming the problem of time-delay, good results are also obtained using the Smith predictor, see [173, 189] for further details. Good results are also obtained for wireless networks control systems, see, for instance, [46]. The examples may continue, the central idea remains the basic Smith predictor, which has been modified/adapted depending on the needs or capabilities of the corresponding systems. A detailed study on the variations of the Smith predictor and their behaviors can be found in [137].

Motivation and Objectives

The global objective is to propose a collaborative haptic platform with remote users affected by important communication time-delays (≈ 50 *ms* forward and backward delays for an average

Internet transmission between two locations in Europe). In present, haptic platforms with *local* users are highly developed. In our opinion, the next step is to consider a single user application, affected by important time-delays. Resuming, the context of this work consists of a haptic system dedicated to a single user, affected by communication time-delays between the controller of the haptic interface and the virtual reality simulator.

Even if there are many studies and solutions in the delay case, the characterization of the effect induced by time-delays on the system's dynamics still remains open. Among the approaches proposed in the literature, there exist a few solutions which are proposing additional guarantees on stability without taking into account the transparency [14, 100, 101, 159, 160].

In teleoperation as well as in haptics, the time-delays decrease the performances of the system in terms of end-user perception [72, 146]. More precisely, the goal of such systems is to provide to the end-user a *perfect* telepresence sensation, i.e. the feeling that the user is directly interacting with the environment without using any additional devices. As mentioned previously, the effects of time-delays can be felt especially by the viscosity effect in free motion and the *soft* response in case of hard contacts. A solution to this problem may be the decreasing or increasing the controller gains, but this functions like a water mattress, increasing the performances in free motion, using *small* gains will lead on a decrease of performances in case of hard contacts and vice versa.

Since this work is focused on haptic systems, and since the first step was to *port* the algorithms from teleoperation into haptics, the second *natural* step is to explore the new opportunities offered by these systems. More precisely, haptic systems offer more clear and predictable environmental interactions, information that can be easily capitalized in order to improve the system's performances.

The main objective is to find new control solutions for preventing the unwanted situations linked to time-delays and also to increase the overall performances for general cases. Starting from the *ideal* situation, when the system is free-of delays (communication and/or computation delays), the performances and guarantees of the system under time-delays will be brought as close as possible to the *ideal* case. Starting from the control solutions applied in teleoperation, the objective is to improve the performances using the additional information available in a haptic system. Also, some specific control schemes and algorithms may be revealed, based on the particularities of the haptic systems with respect to teleoperation ones.

Thesis Plan

Chapter 1 is devoted to an appropriate analysis and to an experimental comparison between the most common control structures used in teleoperation and/or haptics. More precisely, methods as classic Proportional Derivative (PD), PD with local dissipation [101], PD with passivity observer [14, 159, 160], PD with passive set-point modulation [100], wave scattering transform [130] and Smith predictor [36], were appropriately *adapted* from teleoperation to haptics framework and further improved. The goal is to present an extended comparison of the most popular control algorithms used in teleoperation systems, applied to haptic systems. The analyzed methods are in their *basic* form, no specific and/or non-general modifications proposed in the literature were considered.

To the best of the author's knowledge, comparative studies for teleoperation systems can be found in the literature (see, for example, [112, 154]). For instance, for the haptic systems, at least one paper [164] compares only three methods (classic PD, PD with passivity observer and wave scattering) in the context of collaborative haptic over the Internet.

An experimental approach, using one degree of freedom (1-DOF) haptic platform will point out the advantages and inconveniences of each method from a practical point of view. Between the haptic interface controller and the virtual environment controller, the position will be transmitted in both directions, concept corresponding to position-position architecture. To ensure a full control of the communication delays and processing time, all the control algorithms (for haptic interface/virtual object) and virtual environment simulations will be run on the same computer.

The **second chapter** introduces some *theoretical tools* needed in analyzing the stability of the delayed systems in different situations as well as the physical limitations of the experimental platforms considered. First, a short overview of the PID controllers, as well as the time-delay and haptic cases are briefly presented and discussed.

Next, the Proportional-Derivative (PD) controllers under time-delays used in haptics will be analyzed and discussed. More precisely, by using some classical tools, a complete stability analysis of a haptic system under constant time-delays will be presented. Next, one focuses on analyzing the fragility of PD controllers in the presence of I/O time-delays. In order to facilitate the methods, some examples are presented in order to highlight the main ideas. Finally, to complete the presentation, uncertain time-delays are considered. More precisely, uniform distribution and gamma distribution with gap will replace the constant time-delays, and under these circumstances, a stability analysis will be performed. Using a geometrical approach, some examples are presented in order to illustrate the stability results for this case.

Finally, a specific analysis for the Smith predictor control is proposed. As it was previously mentioned, the Smith predictor represents a classic approach for controlling time-delays system. The closed-loop stability analysis is performed from two different points of view:

- Characterizing stability regions in controller's gain parameters (related to the so-called D-decomposition method due to Neimark [127]),
- Characterizing stability regions in delay parameters-space (extending the so-called D-decomposition method [116]).

For the first approach (stability regions in controller's gain parameters), characterizing the time-delay is initially considered to be constant and then multiple delay uncertainties will be analyzed:

- Constant uncertainty,
- Uniform distributed time-delays,
- Gamma distributed time-delays with gap,
- Normal distributed time-delays.

In **the third chapter**, the use of Smith predictor-based control will be addressed and a specific solution is proposed and discussed. The use of Smith predictor provides good results in the free motion case. However, in restricted motion, the results are no longer valid since contact forces must be added in the dynamics, i.e. the model used for prediction is no longer accurate. The main idea of the proposed solution is to introduce into the Smith predictor the environmental forces by using the distances between the controlled virtual object and other objects from the scene. Based on the information received by the virtual environment, it is possible to predict the impact moment and to *update* the predictor's model resulting in an accurate system.

In order to validate the proposed approach, a 3-DOF platform will be used. Finally, the analysis of the experimental results and the concluding remarks end the chapter.

Chapter 4 presents another solution for increasing the system performance - PD with gain scheduling. In order to complete the existing approaches, this chapter proposes a gain-scheduling PD control approach depending on the distance until a possible collision. To the best of the author's knowledge, the majority of the methods proposed in the literature provide more an additional guarantee of the stability, but in terms of end user perception (transparency), there are not important differences. In order to improve the performance with respect to the end user, different values for the controller gains should be used. Since the *classic* PD control is the *core* of all methods, this one was selected as the starting structure for variable gain implementation. The basic idea is to use a *small* gain in free motion in order to obtain a low viscosity movement and

a *high* gain in restricted motion for achieving the desired stiff impact by an appropiate switching from one control strategy to another. Such a switch will depend on the distance between the virtual manipulated object and the other virtual objects present in the scene. In this sense, a complete study, as well as a detailed method description are proposed.

The validation of the suggested method will be made by using a 3-DOF haptic system. The result's analysis and some conclusions close this chapter.

The overall conclusions as well as some perspectives of this work are presented in the last chapter.

In terms of contributions, a complete comparative study of the most commonly used control methods for haptics was proposed and experimentally tested on a 1-DOF haptic platform. The stability and fragility of the PD controllers for haptics under fixed and distributed time delays have been also considered. Next, the stability and fragility were analyzed in the case of Smith predictor-based control scheme for fixed and distributed time delays. Finally, in order to increase the system's performances, two new approaches - *Smith Predictor with distance feedback* and *PD with Gain Scheduling depending on the Distance* were proposed and experimentally tested on a 3-DOF experimental platform.

Various Appendix complete the thesis presentation. Thus, Appendix A presents an extension of the experimental results from Chapter 1 for the most common control algorithms except the Smith predictor, for a constant time-delay of 10*ms* (instead of 50*ms* case introduced and discussed in Chapter 1).

Next, Appendix B presents a method of analyzing the *"trade-off"* between stability and transparency based on 4 channel control scheme used in teleoperation [96, 194]. The goal in bilateral teleoperation is that the master and slave robots should track the same position and the forces acting on the robots should be reflected. This is called *kinesthetic coupling* between the master and the slave systems, and means that the teleoperator is transparent. To attain such goals, the 4 channel teleoperation architecture has been proposed independently by [5] and [194]. Generally, for bilateral teleoperation systems, it is very important to have a high level of transparency. Furthermore, to the best of the author's knowledge, for end-users, this represents the most important characteristic. Next, in order to obtain performance for closed-loop systems, the stability must be carefully taken into account. The analysis proposed here handles both concepts in the corresponding controller's parameter space.

Appendix C introduces the detailed mathematical development of the stability conditions used in Subsection 2.3.1. In Appendix D the obtained results from Section 2.3.1 are further verified on PD-like control scheme proposed by [136]. Finally, Appendix E completes the fragility analysis proposed in Section 2.3.2.

List of publications related to the PhD

A list of the publications submitted/accepted to various journals and conferences is provided below.

Submitted journal papers (1)

- **Bogdan Liacu**, Ahmet Taha Koru, Hitay Ozbay, Silviu-Iulian Niculescu and Claude Andriot. Optimizing Low-Order Controllers for Haptic Systems under Delayed Feedback. *Submitted to Control Engineering Practice, 2012*[6]. (first submission: 20.02.2012, second submission: 26.09.2012, third submission: to be done by 26.11.2012)

Submitted book chapters (1)

- **Bogdan Liacu**, Irinel-Constantin Morarescu, Silviu-Iulian Niculescu, Claude Andriot, Didier Dumur, Frederic Colledani, and Patrick Boucher. Smith Predictor-based Control with Distance Feedback for Haptic Systems under Distributed Time-Delays. *Chapter in Low Complexity Controllers for Time Delay Systems (A. Seuret, C. Bonnet, H. Mounier and H. Ozbay) - to be submitted by the end of November 2012*[7].

Published Conference papers (9)

- **Bogdan Liacu**, Irinel-Constantin Morarescu, Silviu Niculescu, Claude Andriot, Didier Dumur, Frédéric Colledani, and Patrick Boucher. Some Remarks on Smith Predictor-based Control with Distance Feedback for a 3-DOF Haptic System with Distributed Delays. *IFAC Joint conference, Grenoble, France, Feb. 2013.*
- **Bogdan Liacu**, Irinel-Constantin Morarescu, Silviu Niculescu, Claude Andriot, Didier Dumur, Patrick Boucher, and Frédéric Colledani. Proportional-Derivative (PD) Controllers for Haptics subject to Distributed Time-Delays: A geometrical approach. *12th International Conference on Control, Automation and Systems (ICCAS), Jeju, Korea, Oct. 2012.*
- **Bogdan Liacu**, Claude Andriot, Didier Dumur, Silviu-Iulian Niculescu, Frédéric Colledani, and Patrick Boucher. PD Control with Gain-Scheduling depending on the Distance for a 3-DOF Network Based Haptic System. *Multi-Conference on Systems and Control (MSC), Dubrovnik, Croatia, Oct. 2012.*

[6] This publication represents an extended version of the article presented at *10th IFAC Workshop on Time Delay Systems (TDS), Boston, USA, Jun. 2012.*

[7] This publication represents an extended version of the article submitted at *Submitted at IFAC Joint conference (5th Symposium on System Structure and Control, 11th Workshop on Time-Delay Systems, 6th Workshop on Fractional Differentiation and Its Applications), Grenoble, France, Feb. 2013.*

- **Bogdan Liacu**, Irinel-Constantin Morarescu, Silviu Niculescu, Claude Andriot, Didier Dumur, Frédéric Colledani, and Patrick Boucher. Control of a Haptic System using Smith Predictor. *43rd International Symposium on Robotics (ISR), Taiwan, Aug 2012*.

- **Bogdan Liacu**, Claude Andriot, Didier Dumur, Frédéric Colledani, Silviu-Iulian Niculescu, and Patrick Boucher. Experimental Comparative Study of Control Architectures for Haptic Interfaces including Communication Delays. *Mediterranean Conference on Control and Automation, Barcelona, Spain, Jul. 2012.*

- **Bogdan Liacu**, Ahmet Taha Koru, Hitay Ozbay, Silviu-Iulian Niculescu, and Claude Andriot. Low Order Controller Design for Haptic Systems under Delayed Feedback. *10th IFAC Workshop on Time Delay Systems (TDS), Boston, USA, Jun. 2012*.

- **Bogdan Liacu**, Irinel-Constantin Morarescu, Claude Andriot, Silviu Niculescu, Didier Dumur, Patrick Boucher, and Frédéric Colledani. Some Remarks on the Fragility of Smith Predictors used in Haptics. *11th International Conference on Control, Automation and Systems (ICCAS), Gyeonggi-do, Korea, Oct. 2011*.

- **Bogdan Liacu**, Cesar Mendez-Barrios, Silviu-Iulian Niculescu, and Sorin Olaru. Some Remarks on the Fragility of PD Controllers for SISO Systems with I/O Delays. *14th International Conference on System Theory and Control (SINTES14), Sinaia, Romania, Oct. 2010.*

- **Bogdan Liacu**, Cesar Mendez-Barrios, Silviu-Iulian Niculescu, and Sorin Olaru. Some Remarks on the Fragility of Transparency and Stability in General 4-channel Architecture for Bilateral Teleoperation with Delay. *10th International Conference on Control Automation and Systems (ICCAS), Gyeonggi-do, Korea, Oct. 2010*.

Communications without acts (2)

- **Bogdan Liacu**, Claude Andriot, Didier Dumur, Frédéric Colledani, Silviu-Iulian Niculescu, and Patrick Boucher. Control Architectures for Haptic Interfaces including Communication Delays: An Experimental Comparative Study. *1st "DelSys" Workshop, Supélec/L2S, Paris, France, November 20-22, 2012.*

- **Bogdan Liacu**, Frédéric Colledani, Experimental Comparative Study of Control Architectures for Haptic Interfaces Including Communication Delays, *French-Israeli Workshop on Delays & Robustness, Technion-IIT, Haifa, Israel, April 3-5, 2011*.

Chapter 1

Methods and control algorithms used in haptics and teleoperation. A state of the art

1.1 Introduction

The aim of this chapter is to present a comparative study of existing control algorithms for haptic interfaces and virtual environments subject to communication delays. More precisely, methods like Proportional Derivative (PD) with local dissipation [101], PD with passivity observer [14, 159, 160], PD with passive set-point modulation [100], wave scattering transform [130] and Smith predictor [36], introduced in haptics directly from teleoperation will be discussed in the sequel.

To the best of the author's knowledge, the methods in discussion are the most commonly used in the context of haptics. In the teleoperation framework, [112, 154] proposed comparative studies, while for haptics [164] compares only three methods (classic PD, PD with passivity observer and wave scattering) in the case of collaborative haptic over the Internet.

In this chapter, based on some experimental approach, the main characteristics of each method will be pointed out and discused. As mentioned in the introduction, a position-position architecture will be used, denoting the transmission of the position between haptic and virtual controllers. The main objective is to highlight the "+" and "-" of each method with respect to time-delays from the point of view of position tracking error and transparency degree. Furthermore, the results will be to analyzed and compared for each method. For this study, some simple one degree of freedom (1-DOF) haptic platform will be used. To ensure a full control

of the communication delays and processing time, all the control algorithms (for haptic interface/virtual object) and virtual environment simulations will be run on the same computer.

1.2 Haptic & teleoperation systems' schemes

Based on the chronological order of development (first, the teleoperation systems and later the haptic systems), the first scheme presented - Figure 1.1, corresponds to the teleoperation systems.

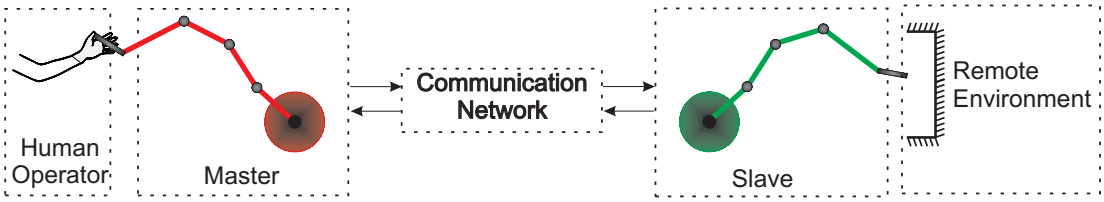


FIGURE 1.1: Teleoperation system scheme

The goal of such a system is to provide an appropriate remote access to the end user in harmful and/or geographically different environments. A teleoperation system is composed by two robots, located in two different places, working on the principle of master/slave. Generally, the human operator imposes a force and/or a position to the master robot that will transmit the command to the slave robot throughout the communication network. Depending on the situations encountered by the slave robot, the master should also receive them via the feedback force (a complete survey on teleoperation systems can be found in [76]).

In Figure 1.2, the general scheme of a network-based haptic system is presented.

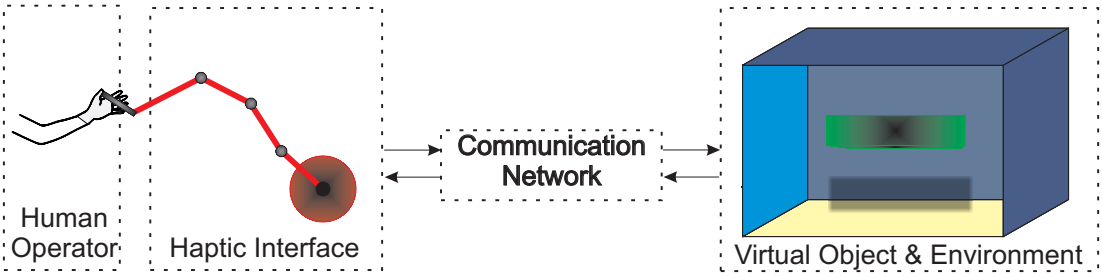


FIGURE 1.2: Haptic system scheme

The main difference between teleoperated and haptic systems is that the slave robot and remote environment are replaced by a virtual robot and a virtual environment, respectively. The principle is the same, the user should *feel* the contacts encountered by the virtual object/robot via the feedback force. As mentioned in the introduction, haptics research seeks to recreate the complex sense of touch for users in virtual reality. Ideally, interacting with a virtual environment

should be as *natural* as a direct interaction with a *real* environment without using any additional systems.

In both cases (*teleoperation and haptics*), the block scheme which is behind the presented systems is basically the same (see Figure 1.3 for the control block scheme).

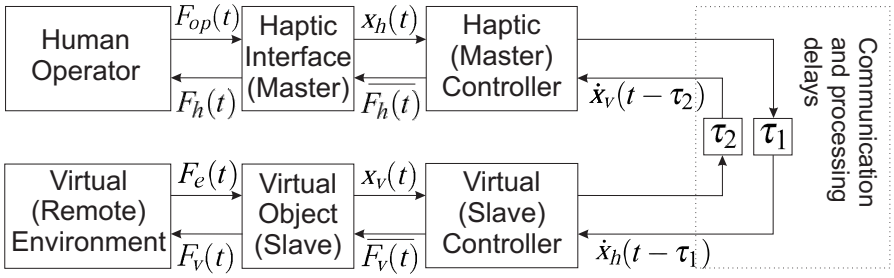


FIGURE 1.3: General block scheme for Haptic/Teleoperation Systems

Here, $\overline{F_{h/v}}$ represents the calculated command corresponding to a specific force. This value represents the command of the master/slave robots that must be *converted* by the DC motors.

As it can be seen from the block scheme from Figure 1.3, the human operator interacts with the system through the master robot (or haptic interface in the case of haptics). Depending on the operator force and based on the information received from the encoders, the position and velocity are determined. The haptic controller calculates the command, i.e. the feedback force for the master robot (haptic interface) based on the information of positions and velocities from the master and slave robots (haptic interface and virtual environment). On the slave (virtual) side, the algorithm is the same, but in reverse order. The command is calculated by the slave/virtual controller based on the information of positions and velocities from the master and slave robots (haptic interface and virtual environment). Next, the command is converted into force by the slave robot/virtual object, force which acts directly with the environment. Further on, based on the action of the slave robot/virtual object, the environment reacts through the environmental force and based on, the position and velocity are determined.

Next, the control scheme is presented in Figure 1.4.

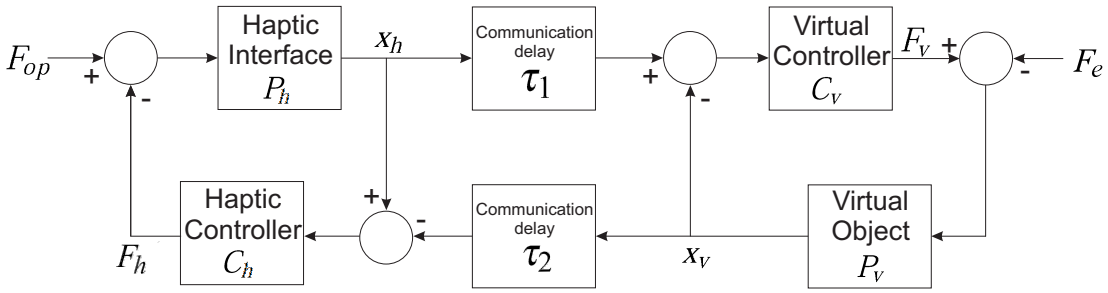


FIGURE 1.4: General control scheme for Haptic/Teleoperation Systems

Starting from this point, the *classical* dynamic (nonlinear) equations of motion for two similar robots in the framework of haptic/teleoperation systems [136] (obtained from the equality of forces, accelerations, frictions and based on the fundamental principle of dynamics) is given by:

$$M_h(x_h)\ddot{x}_h(t) + B_h(x_h,\dot{x}_h)\dot{x}_h = -F_h(t) + F_{op}(t), \tag{1.1}$$

$$M_v(x_v)\ddot{x}_v(t) + B_v(x_v,\dot{x}_v)\dot{x}_v = F_v(t) - F_e(t), \tag{1.2}$$

where x_h, x_v are the haptic interface/virtual object positions, F_{op}, F_e are the human operator/environmental forces, F_h, F_v are the force control signals, M_h, M_v are the symmetric and positive-definite inertia matrices, and B_h, B_v are the Coriolis matrices of the haptic interface and virtual object systems, respectively. The *main* idea is to use two PD controllers, one to control the haptic interface (master robot) and another one for the virtual object (slave robot). The controller equations are there given as follows:

$$F_h(t) = \underbrace{K_{d_h}(\dot{x}_h(t) - \dot{x}_v(t-\tau_2))}_{\text{delayed D-action}} + \underbrace{K_{p_h}(x_h(t) - x_v(t-\tau_2))}_{\text{delayed P-action}}, \tag{1.3}$$

$$F_v(t) = \underbrace{K_{d_v}(\dot{x}_h(t-\tau_1) - \dot{x}_v(t))}_{\text{delayed D-action}} + \underbrace{K_{p_v}(x_h(t-\tau_1) - x_v(t))}_{\text{delayed P-action}}, \tag{1.4}$$

where τ_1, τ_2 are the forward and backward finite constant (or variable) delays and $K_{p_h}, K_{d_h}, K_{p_v}, K_{d_v}$ are the PD control gains for the haptic and virtual controllers respectively.

1.3 Performance Criteria

In order to characterize the system's performance, some criteria need to be defined.

Before introducing the performance criteria, it is worth mentioning that in haptics, as in teleoperation, there exist two functioning cases (situations):

- free motion - when there are no contacts and only the coordination between the haptic interface and the virtual object (master and slave) must be ensured.
- restricted motion - when the virtual object (slave robot) has contacts with the environment, and this phenomenon must be accurately transmitted to the end user via the haptic interface (master robot).

The first criterion considered for the performances evaluation, is the *stability* one. The system's stability must be guaranteed in all situations. The stability condition also guarantees a bounded error for position and velocity. Furthermore, under the assumption that the user is no longer moving the haptic interface (the velocity is zero and the position is constant), in order to achieve

an asymptotically convergence to zero of the virtual object velocity and to ensure position coordination (between the haptic interface and virtual object or master/slave), the system must be stable [38, 136]. Excepting the two cases (free and restricted motion), when the behavior of the system is *clearly* defined in terms of motion, there exist also the transition phases, which sometime can induce many unwanted (and/or unexpected) situations and/or behaviors. More precisely, the system can be stable in free and also in restricted motion, but it may have transition problems, which may cause long time oscillation, and sometimes even instability.

The second criterion is defined as being the *tracking error* between the haptic interface and the virtual object (master and slave). Similarly to the previous criterion, each case (free and restricted motion) must be carefully analyzed. More precisely, in free motion, the desired behavior is to obtain a *perfect* position and velocity tracking between the haptic interface and the virtual object (or master/slave in the teleoperation framework) [11]. Since, in most of the cases, the systems are affected by time-delays, it is worth mentioning that frequent direction changes at high velocities may induce important tracking errors. In this sense, the time response of the used actuators (generally DC motors) is usually limited, or/and other components of the system (except the communication network) may have slow responses. With these considerations, sometimes at high frequencies, the error may increase due to slow response of the system's components. Figure 1.5 illustrates an example of frequent direction changes at high velocities, showing large errors during fast transitions. Next, in restricted motion, the tracking error should be maintained as small as possible. It is important to point out the idea that the feedback force is calculated based on this error, i.e. the force is proportional to the tracking error. Under this circumstance, it is impossible to have zero tracking error, even in cases when there are no communication time-delays (ideal cases) [185]. Figure 1.6 illustrates the dependency between the tracking error and feedback force. An exception is made when PID controllers are used, but, since the "I" gain introduces destabilizing effects, is almost never used for such systems.

The third and last performance criterion is the *transparency level* of the system. Transparency can be defined as the telepresence sense between the operator and the environment (see, for instance [76]). Speaking of both types of systems - haptic and teleoperation, the transparency characterizes the end user sensation of acting directly with the virtual reality or the remote environment without any additional device. There exist two major problems linked to transparency. The first one is that in free motion, the force feedback should be zero, which is impossible, because the system functioning is based on this error [80]. This force should be minimized as much as possible. Once this force is not felt (or is negligible) by the human operator, it is said that the system is transparent in free motion. The second case is related to restricted motion. An important test, in order to establish the level of transparency, is the contact of the virtual object (or slave robot) with other objects. Depending on the speed at the impact moment, the human operator should be able to feel the corresponding impact as if he was directly having contact

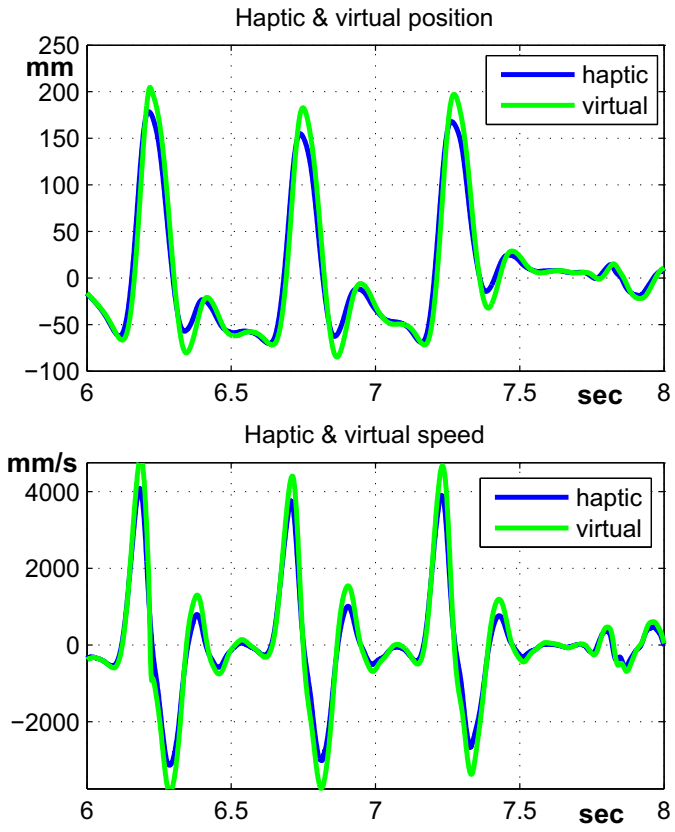


FIGURE 1.5: Frequent changes at high velocities.

with the respective object [154]. Generally, for the benchmarks, stiff walls will be used. An accurate impact feeling is provided by a fast rising force feedback, see Figure 1.7a. A progressive rising force creates a soft impact, resulting in a loss of transparency and an overall performance degradation, see Figure 1.7b.

Summarizing these criteria, the following steps are obtained:

- Guarantee the stability in free and restricted motions;
- Minimize the tracking error in free and restricted motions;
- Provide the maximum level of transparency.

These criteria can be presented in a three-layers representation, as depicted in Figure 1.8.

The layer representation highlights the idea that the objective must be achieved one by one in a *specific* order. More precisely, in the author's opinion, the stability must be guaranteed before tracking error and transparency.

This problem may also be seen from the tuning parameters point of view. Guaranteeing the stability will define an allowable range for the tuning parameters. Next, the allowable range

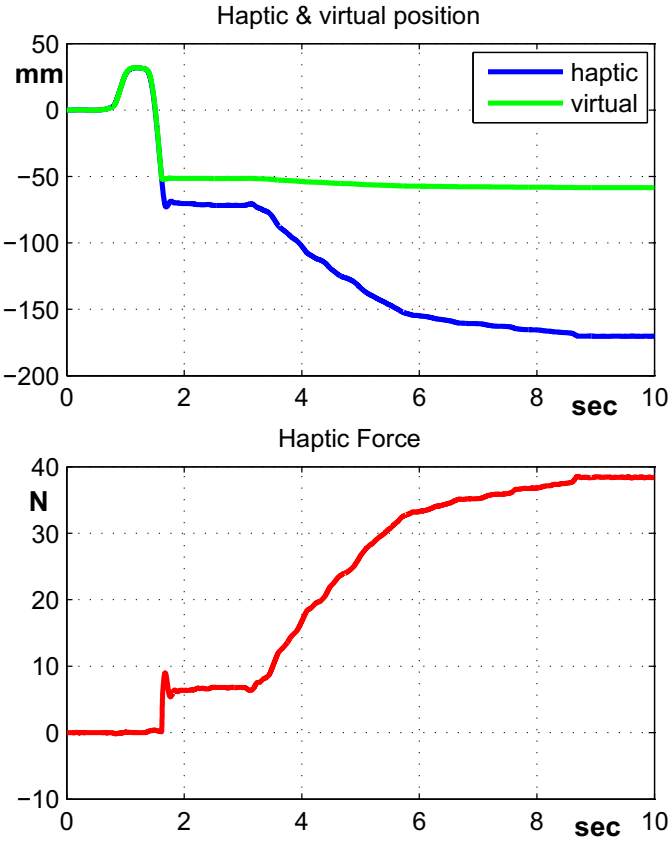
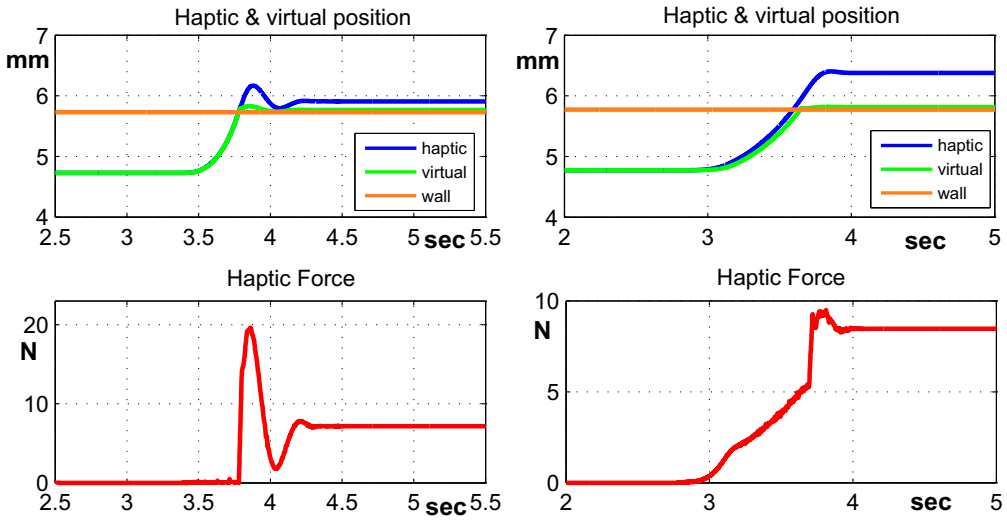


FIGURE 1.6: Dependency between the tracking error and the feedback force.



a. Stiff force

b. Soft force

FIGURE 1.7: Force rising - stiff and soft feeling.

which provides a *reasonable* tracking error should be included in the stability range. Finally the corresponding bounds, guarantying the minimum desired level of transparency should also be less than or equal to the previous defined zone. Figure 1.9 summarizes this idea.

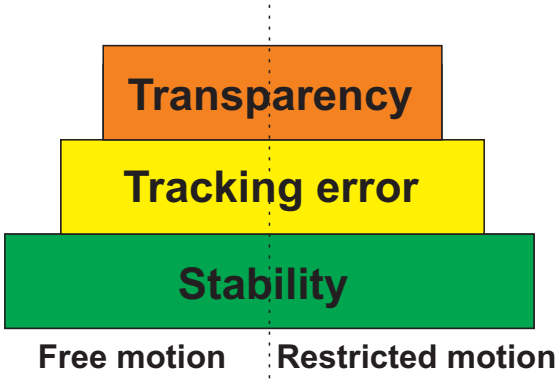


FIGURE 1.8: Criteria's layer.

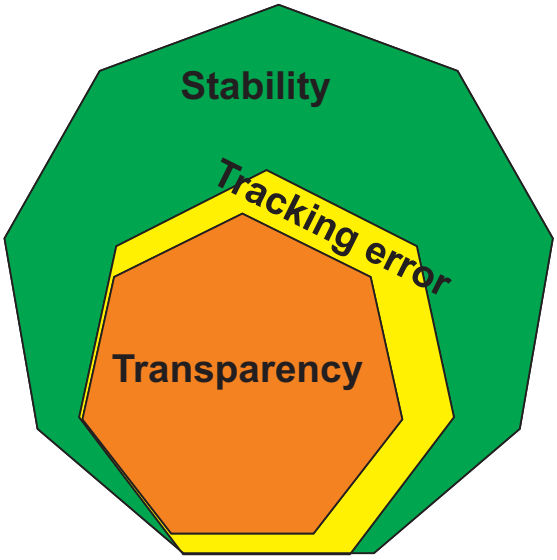


FIGURE 1.9: Parameters' ranges.

1.4 Commonly used algorithms (state of the art)

It is well known that the presence of time-delays in haptics and teleoperation affects the stability, as well as the transparency of such systems. More precisely, in free motion the delay effect can be felt by the viscosity phenomenon (high force feedback felt at the haptic interface end), in the case of a hard contact with the environment, the impact effect will not be stiff, or the most unwanted situation is to loose the system stability due to the delays. The delays must be taken into account and included in the control laws. However, a trade-off between stability, position tracking error and transparency must be always made. Generally, there are a few *major* approaches, on which most of the solutions are based on.

In present, among the literature, there can be found many methods and algorithms for teleoperated and haptic system. To the best of the author's knowledge the main algorithms are:

- *Classic* Proportional Derivative control [136]

- Proportional Derivative control with local dissipation [101]
- Proportional Derivative control with passivity observer [14, 159, 160]
- Proportional Derivative control with passive set-point modulation [100]
- Wave-Scattering transform [130]
- Proportional Derivative control with Smith predictor [36]

In this section these algorithms will be presented, analyzed and discussed.

1.4.1 Classic Proportional-Derivative (PD)

According to the literature [39, 156, 180], Proportional-Derivative (PD) controllers are largely used in teleoperation systems as well as in haptics. For such systems, a very fast response is required and, in most of the cases, they are affected by large communication delays.

In the case of classic PD controller, equations (1.3)-(1.4) remain unchanged. In terms of *transparency*, it is required to have F_h (feedback force) as small as possible in free motion, while in restricted motion as high as possible. In our opinion, there is a contradiction, since it is impossible to obtain the desired performances in both cases with the same gains, and thus, always a compromise must be made in order to guarantee *reasonable* performances in both cases.

1.4.2 Proportional-Derivative (PD) control with local dissipation

The main idea of this method is to include an additional term *watching* on stability. This term is a local dissipation acting in order to maintain the passivity of the system.

A system is said to be passive if and only if:

$$\int_0^t F(\tau)\dot{x}(\tau)d\tau + E(0) \geq 0, \quad \forall t > 0, \tag{1.5}$$

where $\dot{x}$ and F are the variables denoting velocity and force respectively, and $E(0)$ is the energy stored initially in the system at $t = 0$. Passivity is also a sufficient condition for stability [130].

The method was proposed by [101] based on the controller passivity concept, Lyapunov-Krasovskii technique for the delayed systems, and Parseval's identity.

In order to achieve coordination between the haptic interface and the virtual object, bilateral force reflection[1], and energetic passivity[2] of the closed-loop system, the haptic interface and the virtual object control forces $F_h(t), F_v(t)$ from equations (1.3) and (1.4) are designed as follows:

$$F_h(t) = \underbrace{K_{d_h}(\dot{x}_h(t) - \dot{x}_v(t-\tau_2))}_{\text{delayed D-action}} \underbrace{+(-K_{diss} + P_e)\dot{x}_h(t)}_{\text{dissipation}} \underbrace{+K_{p_h}(x_h(t) - x_v(t-\tau_2))}_{\text{delayed P-action}}, \tag{1.6}$$

$$F_v(t) = \underbrace{K_{d_v}(\dot{x}_h(t-\tau_1) - \dot{x}_v(t))}_{\text{delayed D-action}} \underbrace{-(-K_{diss} + P_e)\dot{x}_v(t)}_{\text{dissipation}} \underbrace{+K_{p_v}(x_h(t-\tau_1) - x_v(t))}_{\text{delayed P-action}}, \tag{1.7}$$

where K_{diss} ($K_{diss} = 0.05K_p$) is the dissipation gain to *passify* the delayed D-control action and P_e is an additional damping ensuring coordination between the haptic interface and the virtual object (for more details, see [101]). Resuming, the main idea of this method is to add an additional gain in order to guarantee the system's passivity.

1.4.3 Proportional-Derivative (PD) control with passivity observer

This method was patented in 2006 by Hannaford et *al.* [65]. The provided method is used for stabilizing a haptic interface of computer controlled virtual-reality or teleoperation systems comprising a robot manipulator. According to the authors, 'stabilizing' means to reduce the sense of vibration in haptic interface.

The energy of the network elements is observed using the Passivity Observer (PO) introduced in [159, 160]. For the one-port element:

$$E_{obs}(n) = \Delta T \sum_{k=0}^{n} (F(k)\dot{x}(k)), \tag{1.8}$$

where $F(k)$ and $\dot{x}(k)$ denote force and velocity of the port conjugate pair, $E_{obs}(n)$ is the real time observed energy with the sign convention represented as in Figure 1.10, and ΔT is the sample time of the system.

For a two-port network, equation 1.8 becomes:

$$E_{obs}(n) = \Delta T \sum_{k=0}^{n} (F_1(k)\dot{x}_1(k) - F_2(k)\dot{x}_2(k)), \tag{1.9}$$

where $F_1, \dot{x}_1, F_2$ and $\dot{x}_2$ are the conjugate pairs on both ports respectively with the sign convention represented as in Figure 1.10.

[1] The ability of the system to provide the same force at both ends based on the human and environmental forces as well as on the calculated commands.

[2] Guaranteeing the system passivity from the energetic point of view, i.e. the energy must be equal to or greater then 0.

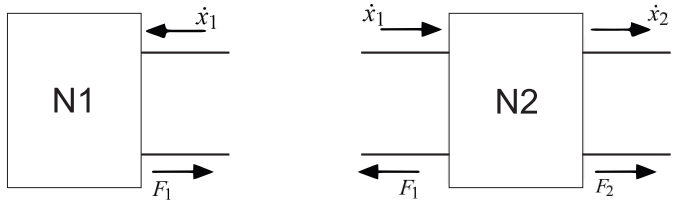


FIGURE 1.10: One-port network element Two-port network elements

Based on the passivity observer, it was further build the passivity controller (PC). In the sequel the one port network case will be detailed. There are two possible configurations:

- series connection (see Figure 1.11.a):

$$\dot{x}_1 = \dot{x}_1' - \frac{F_1}{\alpha}, \tag{1.10}$$

- parallel connection (see Figure 1.11.b):

$$F_1 = F_1' + \alpha \dot{x}_1, \tag{1.11}$$

where α is defined as follows:

$$\alpha(t) = \begin{cases} \frac{-E_{obs}(t)}{\Delta T \dot{x}_1(t)^2} \text{ if} & E_{obs} < 0 \\ 0 & E_{obs} \geq 0 \end{cases} \tag{1.12}$$

for the series case, and for the parallel case:

$$\alpha(t) = \begin{cases} \frac{\Delta T F_1(t)^2}{-E_{obs}(t)} \text{ if} & E_{obs} < 0 \\ 0 & E_{obs} \geq 0 \end{cases} \tag{1.13}$$

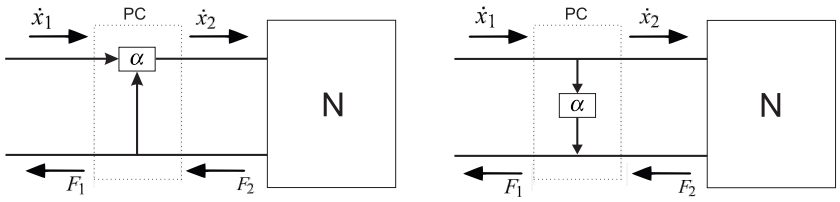


FIGURE 1.11: a. Series connection b. Parallel connection

The energy of the system based on Figure 1.12, is observed using the Passivity Observer (PO) introduced in [159, 160], as follows:

$$E_{obs}(t) = \Delta T \sum_{k=0}^{n} (F(k)\dot{x}(k)), \tag{1.14}$$

where $E_{obs}(t)$[3] is the real time observed energy with the sign convention represented as in Figure 1.12, and ΔT is the sample time of the system.

Among the two possible configurations, only the series connection one is considered (see Figure 1.12), due to the missing of force sensor on the experimental platform considered in Section 1.5.1:

$$F_h(t) = F'_h(t) + \alpha_h(t)\dot{x}_h(t), \tag{1.15}$$

where $\alpha_h(t)$ is defined similarly to equation 1.12.

In this case, equations (1.3) and (1.4) become:

$$F_h(t) = \underbrace{K_{d_h}(\dot{x}_h(t) - \dot{x}_v(t-\tau_2))}_{\text{delayed D-action}} \underbrace{+K_{p_h}(x_h(t) - x_v(t-\tau_2))}_{\text{delayed P-action}} \underbrace{+\alpha_h(t)\dot{x}_h(t)}_{\text{PC}}, \tag{1.16}$$

$$F_v(t) = \underbrace{K_{d_v}(\dot{x}_h(t-\tau_1) - \dot{x}_v(t))}_{\text{delayed D-action}} \underbrace{+K_{p_v}(x_h(t-\tau_1) - x_v(t))}_{\text{delayed P-action}} \underbrace{-\alpha_v(t)\dot{x}_v(t)}_{\text{PC}}, \tag{1.17}$$

where α_h is the corresponding coefficient for the haptic interface previously defined and α_v is the virtual object coefficient, similarly defined.

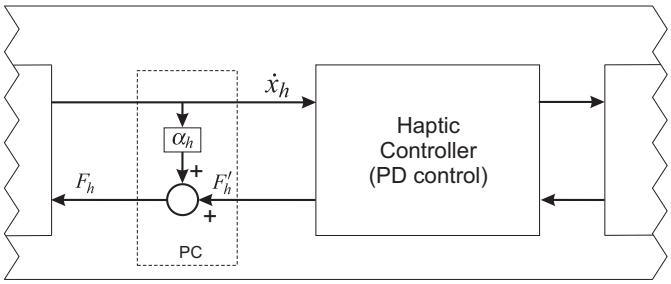


FIGURE 1.12: Passivity Controller

In the sequel, the most common problem of this solution are discussed. According to [64], a potential problem which may occur is that the forces required to dissipate the generated energy may exceed the actuator limits. Generally this phenomenon appears for small velocities. A possible solution is to limit the value of α [64]. Another solution could be the limitation of the force generated by the PC, or both in the same time. For instance, [70] proposed an exact computation of the maximum value for α:

$$\alpha(t) \leq \frac{m}{(1+d)\Delta T} = \alpha_{max}, \tag{1.18}$$

where m is the mass of the system, ΔT is the sampling period and $d \in \mathbb{N}$ is the number of round trip[4] delay sampling periods.

[3] Notation in discrete time is, $t : n\Delta T$, where ΔT is sampling time.

[4] Round trip is defined as being the sum of forward and backward delays ($\tau_1 + \tau_2$, in our case, according to Figure 1.3)

Further on, due to fast sign changes of the velocity at small values will result in a noisy system response. A detailed study and solution for removing the noisy behavior are proposed in [158]. The solution suggests a method to ignore the produced energy from the velocity sign change and another one to maintain the PC force when the velocity is equal to zero.

Another problem observed by [14] is the accumulation of extra energy due to time delays. This phenomenon makes the system vulnerable to instability because the amount of time needed to have a negative energy is often very large and so the reaction will be too slow. In order to solve this problem, the same paper proposes an energy resetting strategy driven by the following two conditions:

$$f < f_{th}, \quad t > t_{th},$$

where f, t are the current force and time, and f_{th}, t_{th} are the force and time thresholds. The values for the thresholds are chosen by observing the characteristics of the delayed channel (for more details, see [14]).

One of the latest paper [13] presents the problem of position drift. By using the PC, some energy is dissipated and lost. The accumulation of these losses results in a position difference between the two ends. The proposed solution is to generate energy in order to actively compensate the drift as allowed by the passiveness of the communication channel.

The PC has several desirable properties for applications including haptic interface control. The PO and PC can both be implemented with simple software in existing haptic interface systems. Energy storage elements in the system do not have to be modeled, only dissipation. Dissipation in the elements outside the PO needs to be identified for optimum performance [64].

1.4.4 Proportional-Derivative (PD) control with Passive Set-Position Modulation

Based on the same theory as the previous method (Eq. 1.5), as reported in [100], this approach has big tolerance to package loss[5] and variable time-delays. More precisely, when a data package is lost, the previous set position is maintained. In order to enforce passivity, the control action is restricted before its application, policy that does not possess any singularity in the result.

According to [100], in our case, the passive set-position modulation can be expressed as follows:

$$\begin{aligned} &\min_{\bar{x}_h(t)} \| x_h(t) - \bar{x}_h(t) \| \\ &\text{s.t. } E_h(t) = E_h(t-1) + K_{d_h}\dot{x}_h(t-1)^2 - \tfrac{1}{2}K_{p_h}(-\bar{x}_h(t) + \\ &\quad + \bar{x}_h(t-1))(2x_h(t) - \bar{x}_h(t) - \bar{x}_h(t-1)) \geq 0 \end{aligned} \tag{1.19}$$

[5] Due to the network imperfections, especially when the Internet® is used, the data at given moment may be lost.

$$\begin{aligned}&\min_{\bar{x}_v(t)} \parallel x_v(t) - \bar{x}_v(t) \parallel \\ &\text{s.t. } E_v(t) = E_v(t-1) + K_{d_v}\dot{x}_v(t-1)^2 - \tfrac{1}{2}K_{p_v}(-\bar{x}_v(t)+ \\ &\quad +\bar{x}_v(t-1))(2x_v(t) - \bar{x}_v(t) - \bar{x}_v(t-1)) \geq 0\end{aligned} \tag{1.20}$$

where $\bar{x}_h(t)$ and $\bar{x}_v(t)$ represent the modulated set positions for haptic interface and virtual object and E_h, E_v are the accumulated energies on the haptic and virtual side respectively. With these considerations equations (1.3) and (1.4) rewrite as follows:

$$F_h(t) = \underbrace{K_{d_h}(\dot{\bar{x}}_h(t) - \dot{x}_v(t-\tau_2))}_{\text{delayed D-action}} \underbrace{+K_{p_h}(\bar{x}_h(t) - x_v(t-\tau_2))}_{\text{delayed P-action}}, \tag{1.21}$$

$$F_v(t) = \underbrace{K_{d_v}(\dot{x}_h(t-\bar{\tau}_1) - \dot{\bar{x}}_v(t))}_{\text{delayed D-action}} \underbrace{+K_{p_v}(x_h(t-\tau_1) - \bar{x}_v(t))}_{\text{delayed P-action}}. \tag{1.22}$$

As mentioned in [100], this method is flexible, local/decentralized, does not involve often problematic numerical integration/differentiation, and can be easily extended to nonlinear robots. Due to optimization-based procedure, this modulation strategy is also free from the incidental diversion/wave-reflection problem of the scattering/wave-based approaches when the packet-loss is substantial (especially during the hard-contact task) and also from the noisy-behavior/sudden impulsive-force problems of the time-domain passivity control when the robot's velocity is slow.

1.4.5 Wave-Scattering Transform

Also based on the theory of passivity, wave variables present a modification/extension which creates robustness to arbitrary time delays [133]. According to [134], the transmission of wave variable provides an efficient and *simple* implementation while guarantying also the system stability for unknown time-delays. In order to achieve the desired performances, wave variables provide an alternative information encoding scheme to the standard power variables. The method is applicable to nonlinear systems and can handle unknown models and large uncertainties, thus suited for interaction with real physical environments.

The basic wave transformation relates velocity, force, right and left moving waves [133]. In our case, see Figure 1.13, $\dot{x}_h$ and $\dot{x}_v$ will be converted into u_m and v_s.

The wave variables (u_m, v_s) can be computed from the standard power variables as follows:

$$u_m(t) = \frac{b\dot{x}_h(t) + \dot{x}_{v_d}(t)}{\sqrt{2b}}, \qquad v_s(t) = \frac{\dot{x}_v(t) - b\dot{x}_{h_d}(t)}{\sqrt{2b}}, \tag{1.23}$$

where b is the characteristic wave impedance and may be a positive constant or a symmetric positive defined matrix and $\dot{x}_{h_d}$, $\dot{x}_{v_d}$ are the delayed outputs after applying the wave scattering

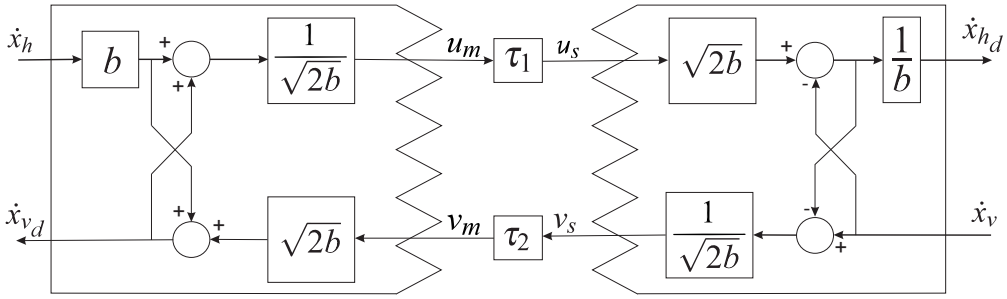


FIGURE 1.13: Wave variables scheme.

transform. The characteristic wave impedance b also assumes the role of a tuning parameter, which can trade off the velocity of motion against the level of forces, and influences many other characteristics. More precisely, increasing the wave impedance will place a larger weight on the velocity compared to the force, making the system appear more damped. When the wave impedance is decreased, force levels are lower, motion is easier and the system appears less damped, for more details, see [133].

This transformation is bijective, so that it is always unique and invertible. No information is lost or gained by encoding the variables in this way. In practice, the wave transformations provide an interface between power and wave variables.

In this case equations (1.3) and (1.4) rewrite as follows:

$$F_h(t) = \underbrace{K_{d_h}(\dot{x}_h(t) - \dot{x}_{v_d}(t))}_{\text{delayed D-action}} \underbrace{+K_{p_h}(x_h(t) - x_{v_d}(t))}_{\text{delayed P-action}}, \tag{1.24}$$

$$F_v(t) = \underbrace{K_{d_v}(\dot{x}_{h_d}(t) - \dot{x}_v(t))}_{\text{delayed D-action}} \underbrace{+K_{p_v}(x_{h_d}(t) - x_v(t))}_{\text{delayed P-action}}, \tag{1.25}$$

where:

$$\begin{aligned} \dot{x}_{h_d}(t) &= \tfrac{1}{b}(-\dot{x}_v(t) + \sqrt{2b}u_m(t-\tau_1)), \\ \dot{x}_{v_d}(t) &= b\dot{x}_h(t) + \sqrt{2b}v_s(t-\tau_2). \end{aligned} \tag{1.26}$$

It is well known that this basic wave variables scheme introduces wave-based reflections (see for instance [134]). In the literature many solutions can be found for reducing the reflections, like [135].

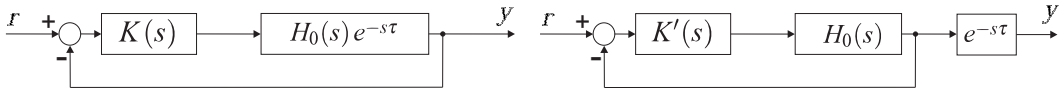
In terms of transparency, the wave variables method add an additional term b/τ_2 in steady state and an additional inertia $b\tau_2$ during the motions at the haptic end and similar at the virtual end - b/τ_2, $b\tau_2$ (for more details, see [131]).

Wave variables provide an alternative information encoding scheme to the standard power variables. The required transformations are extremely simple and preserve all information.

1.4.6 Smith Predictor

A common method used in controlling time-delay systems is based on Smith predictor [174]. The basic idea is to take advantage of some interconnection transformation by taking the delay out of the loop and constructing a controller for an equivalent simpler scheme. If the idea is appealing, it is important to note that it works only under some constrains on the system's dynamics and/or on delays. A Smith predictor-based control scheme for motion synchronization in virtual environments under large time-delays is presented in [35]. In teleoperation, among many solutions proposed for overcoming the problem of time-delay, good results are also obtained using the Smith predictor, see [12, 173, 189] for further details. Good results are also obtained for wireless networks control systems, see for instance [46]. The examples may continue, the central idea remains the basic Smith predictor, which has been modified/adapted depending on the needs or capabilities of the systems. It is worth mentioning that different modifications of the Smith predictor were proposed in the literature. For example [113] proposes a modification that involves the design of extra compensators in the two feedback paths in the Smith predictor structure in order to reduce the effect of load disturbances (this modification was also discussed by [55]). In [141], a solution involving the feedforward of a disturbance signal acting on the process for improving the regulatory performances is proposed. Another modification suggested by [15], is to use a decoupled set point response from the load response in order to avoid the steady state error, since the conventional Smith predictor may not be used for processes modeled by an integrator and time-delay. A complete study on the variations of the Smith predictor can be found in [137].

Considering a relatively simple SISO configuration - Figure 1.14.a, affected by time-delays, the goal is to obtain a system having the delay out-of-the-loop, as shown in Figure 1.14.b.



a. System affected by time-delays b. Out-of-the-loop delay system

FIGURE 1.14: Smith predictor basic scheme.

The transfer functions of the systems represented in Figures 1.14.a-b are given by:

$$H_{ry}(s) = \frac{K(s)H_0(s)e^{-s\tau}}{1+K(s)H_0(s)e^{-s\tau}}, \qquad H'_{ry}(s) = \frac{K'(s)H_0(s)}{1+K'(s)H_0(s)}e^{-s\tau}.$$

Considering $H \equiv H'$ that is the interconnection transformation not appealing the transfer function, K takes the form:

$$K(s) = \frac{K'(s)}{1+K'(s)H_0(s)(e^{-s\tau}-1)}. \tag{1.27}$$

The Smith predictor control system (Figure 1.15) can predict the object's response and compensate the time delays resulting in an improvement of the dynamic characteristic [174]. More precisely, when dealing with time-delay systems, the use of Smith predictor will pull out the delay from the loop. However the basic form of Smith predictor is not working if $H_0(s)$ is not stable, or for uncertain (not perfectly known) time-delays τ (for more details about the Smith predictor limitations, see, for instance, [113]).

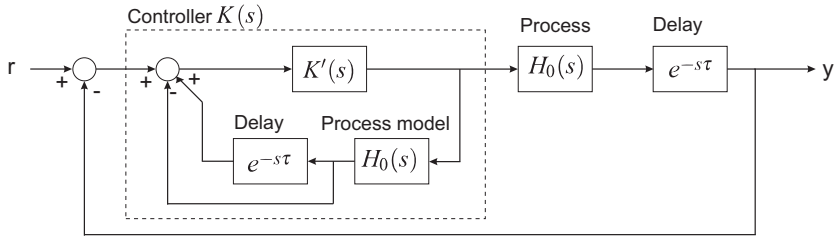


FIGURE 1.15: Smith predictor basic scheme.

In the haptic framework, the Smith predictor will be used just on the haptic interface side in order to compensate the delay effects on transparency. This solution was chosen in order to avoid additional uncertainties linked to the second predictor. The entire virtual part was considered to represent the delayed process, and completely included in the predictor. Figure 1.16 presents the Smith predictor control scheme for the haptic system (based on Figure 1.3).

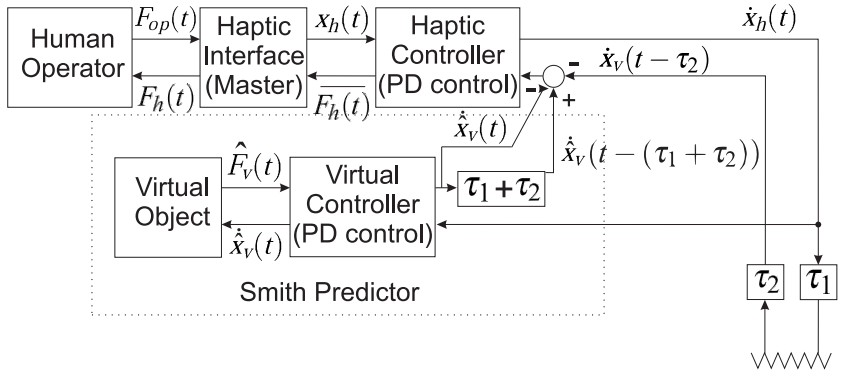


FIGURE 1.16: Smith predictor scheme for haptic interface.

In this case, equations (1.3) and (1.4) become:

$$F_h(t) = \underbrace{K_{d_h}(\dot{x}_h(t)-\dot{x}_v(t-\tau_2)+\dot{\hat{x}}_v(t-(\tau_1+\tau_2))-\dot{\hat{x}}_v(t)}_{\text{delayed D-action}} \\ \underbrace{+K_{p_h}(x_h(t)-x_v(t-\tau_2)+\hat{x}_v(t-(\tau_1+\tau_2))-\hat{x}_v(t))}_{\text{delayed P-action}}, \tag{1.28}$$

$$F_v(t) = \underbrace{K_{d_v}(\dot{x}_h(t-\tau_1)-\dot{x}_v(t))}_{\text{delayed D-action}} \underbrace{+K_{p_v}(x_h(t-\tau_1)-x_v(t))}_{\text{delayed P-action}}, \tag{1.29}$$

where $\dot{\hat{x}}_v$, $\hat{x}_v$ represent the estimated velocity and position for virtual object.

Based on the control scheme presented in Figure 1.3, the new control scheme including the Smith predictor is presented in Figure 1.17.

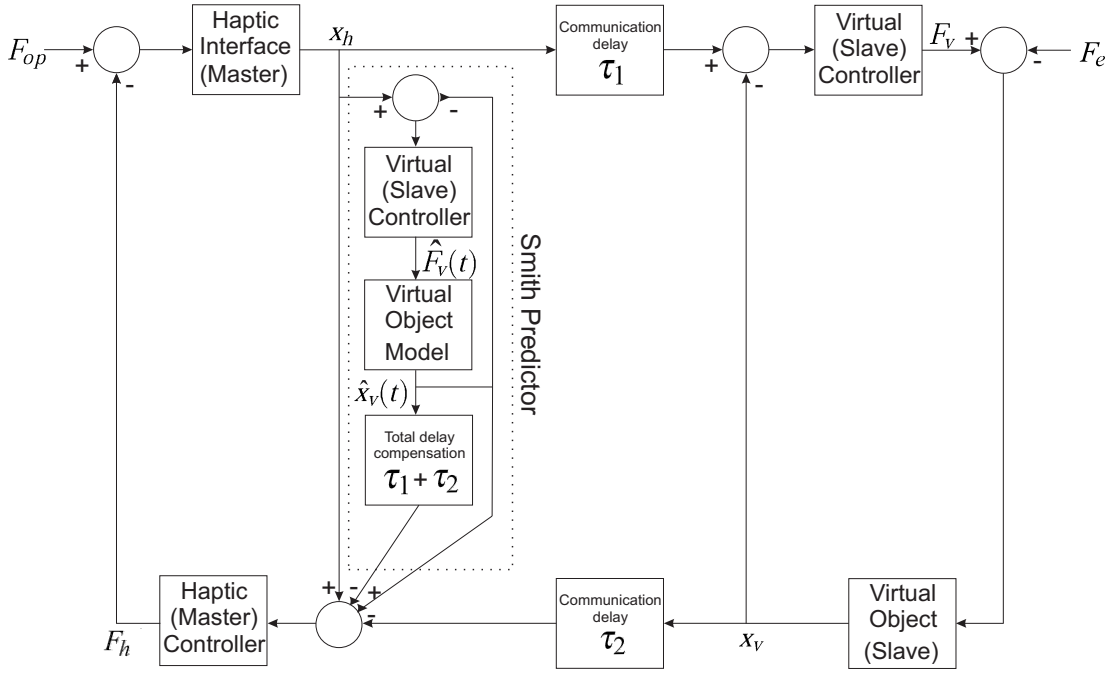


FIGURE 1.17: Control scheme for haptic interface including the Smith predictor.

According to [36], motion synchronization and invariant local dynamics for distributed interconnected systems can be achieved by using Smith predictor.

Classic Smith predictor needs accurate identity between the compensation model and actual object. When the model has obvious errors, the control quality of Smith control system will become bad, even worse than without predictor, as discussed by [179].

1.5 Experimental comparative study of the most common algorithms

This section describes the 1-DOF experimental platform and presents in a comparative manner the experimental results obtained for the algorithms presented in the previous section.

1.5.1 Presentation of the 1-DOF haptic experimental platform

As previously mentioned, in order to ensure a full control of the communication delays and processing time, all the control algorithms (for haptic interface/virtual object) and virtual environment simulations will be run on the same computer.

The haptic interface, Figure 1.18, consists of one direct-drive motor and an optical quadrature encoder with 2000 pts/rev (with a gear ratio of 1/10). The controllers and the virtual simulation are running in real time mode (on RTAI Linux) with a sampling time of 1 ms.

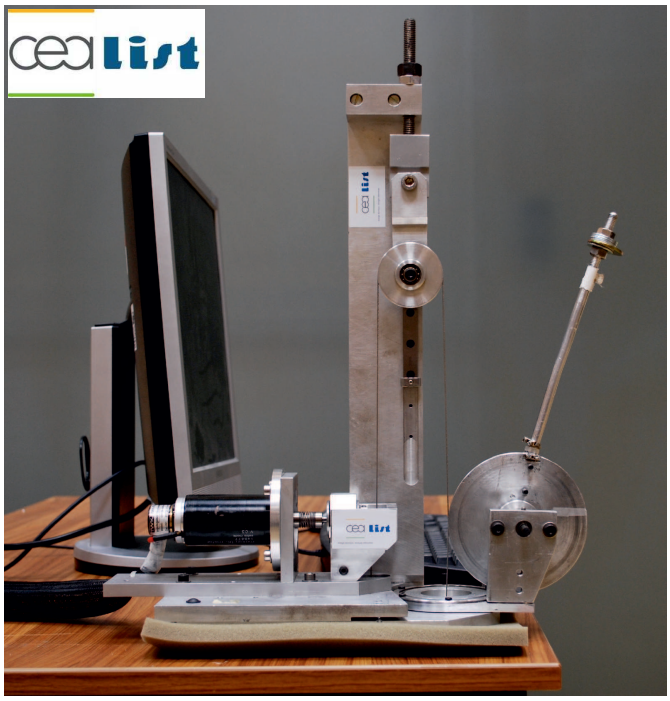


FIGURE 1.18: Experimental Platform

The virtual object is modeled to be similar to the haptic interface ($M_h = M_v$, $C_h = C_v$). The virtual wall which results in force environment F_e is defined by the following equation:

$$F_e = K_{wall}(x_v - x_{wall}) + B_{wall}\dot{x}_v, \tag{1.30}$$

where $K_{wall} = 20000\ N/m$ and $B_{wall} = 10\ Ns/m$ represent the stiffness and the damping used to compute the virtual force environment. Here, x_{wall} defines is the virtual wall position and x_v, $\dot{x}_v$ are the virtual object position and velocity.

1.5.2 Results' Analysis

1.5.2.1 Comparison Criteria

Before detailing our analysis, the comparison criteria are briefly defined. Each method presented in the previous section is discussed, using the following criteria with their corresponding scenarios:

- first, position tracking error in the case of restricted motion and constant delay, using an optimal tuning in order to have a minimal position tracking error,

- second, viscosity effect in the case of free motion and constant delay, using an optimal tuning in order to have a minimal position tracking error,
- third, position tracking error in the case of restricted motion and constant delay, using an optimal tuning in order to have a minimal viscosity effect,
- finally, viscosity effect in the case of free motion and constant delay, using an optimal tuning in order to have a minimal viscosity effect.

1.5.2.2 Results

Figure 1.19 presents the *ideal*[6] system behavior in free and restricted motions.

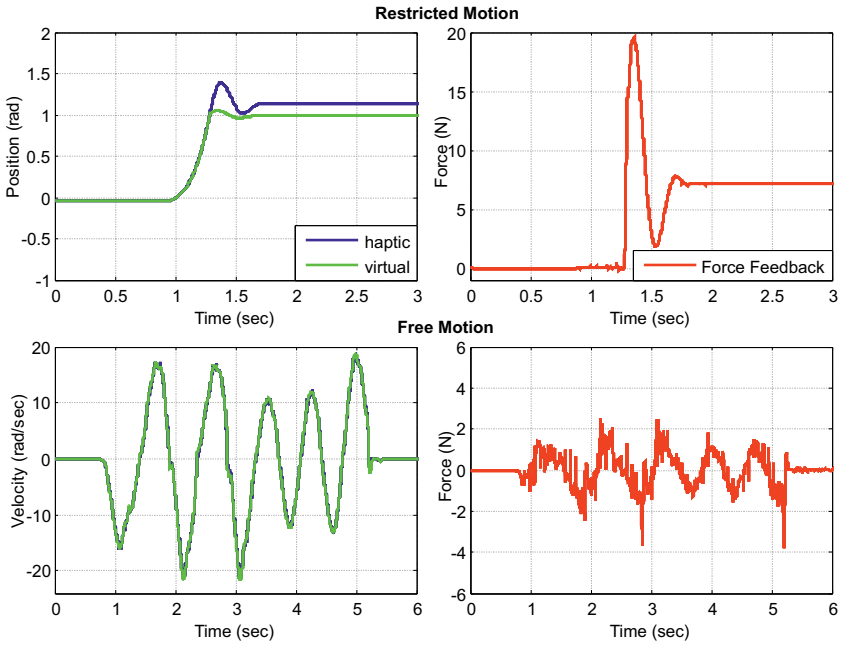


FIGURE 1.19: Ideal Haptic System Behavior.

As mentioned before, an *ideal* haptic system must have a *small* position tracking error in restricted motion and an *insignificant* force feedback (low viscosity i.e. high degree of transparency) in free motion.

In the sequel, the analysis is made for two different constant time-delay values :

$$\begin{cases} \tau_1 = \tau_2 = 10\ ms, \\ \tau_1 = \tau_2 = 50\ ms. \end{cases}.$$

[6] ideal conditions are considered to be the conditions in which the system is free of communication time-delays

However, the major interest of this thesis is to understand the system behavior for *large* time-delays, characteristic of Internet/long distance communication delays. Therefore, since the interesting case is represented by the second one ($\tau_1 = \tau_2 = 50ms$), only these results are given in Figure 1.19, the results corresponding to the first one ($\tau_1 = \tau_2 = 10ms$) are presented in Appendix A.

For the next experiments, a fixed delay $\tau_1 = \tau_2 = 50\ ms$ and a constant human operator force ($F_h = 7\ N$) for the restricted motion cases were considered.

The delays related to the virtual reality simulations were not taken into account, since the simulations run at the same frequency. The basic idea is to set up an optimal tuning for restricted motion in order to have a minimal position tracking error (close to the *ideal* case) and then to analyze it for free motion case. Next, the gains will be set up in order to obtain a reduced viscosity (near to the *ideal* case) and then the restricted motion case will be analyzed. Based on the *ideal* behavior, the gains were tuned for the two cases as follows:

- best performance for position tracking error in the case of restricted motion, Figures 1.20 and 1.21 (free and restricted motion), with:
  - for the first four methods[7]:

$$K_p = 1500\ N/m, \qquad K_d = 80\ Ns/m,$$

  - for the wave-scattering method:

$$K_p = 1000\ N/m, \qquad K_d = 70\ Ns/m, \qquad b = 0.2,$$

  - for the Smith predictor method:

$$K_p = 950\ N/m, \qquad K_d = 150\ Ns/m,$$

- best performance for viscosity in the case of free motion, Figures 1.22 and 1.23 (free and restricted motion), with:
  - for the first four methods:

$$K_p = 200\ N/m, \qquad K_d = 15\ Ns/m,$$

[7] classic PD, PD with local dissipation, PD with passivity observer and PD with set-point modulation

– for the wave-scattering method:

$$K_p = 180\ N/m, \qquad K_d = 15\ Ns/m, \qquad b = 0.3,$$

– for the Smith predictor method:

$$K_p = 250\ N/m, \qquad K_d = 90\ Ns/m.$$

Table 1.1 summarizes the controller gains' values for each method and case.

Method	**Optimal - Restricted motion**		**Optimal - Free motion**	
	$K_p\ (N/m)$	$K_d\ (Ns/m)$	$K_p\ (N/m)$	$K_d\ (Ns/m)$
Classic Proportional-Derivative (PD)	1500	80	200	15
Proportional-Derivative control with local dissipation				
Proportional-Derivative control with passivity observer				
Proportional-Derivative control with Passive Set-Position Modulation				
Wave-Scattering Transform	1000	70	180	15
Smith Predictor	950	150	250	90

TABLE 1.1: Controller gains' values.

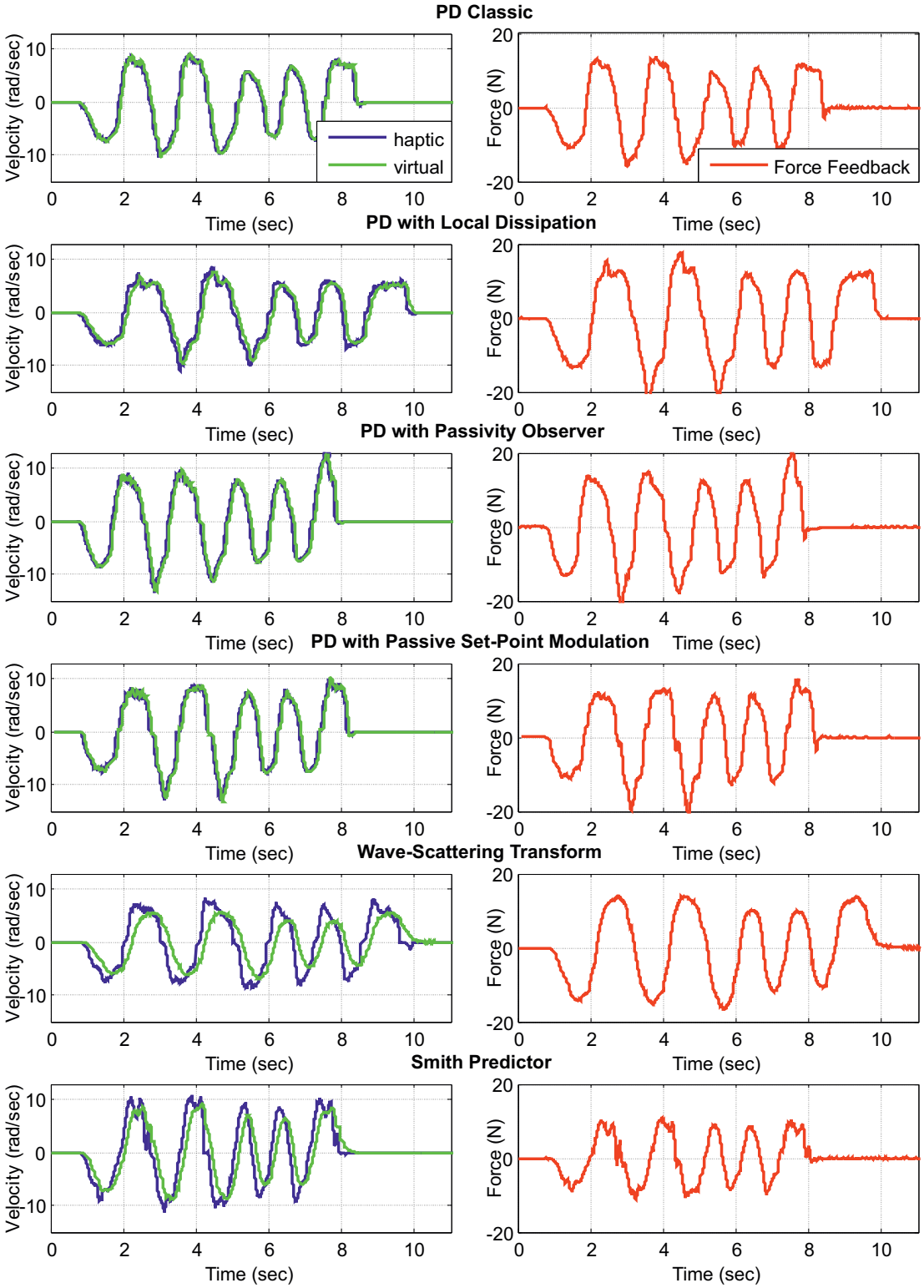


FIGURE 1.20: Optimal error tracking - free motion, 50 ms delay.

In Figure 1.24 the maximum position tracking error and the average force feedback (measured at a speed of 8 rad/sec) is presented for each method and each case.

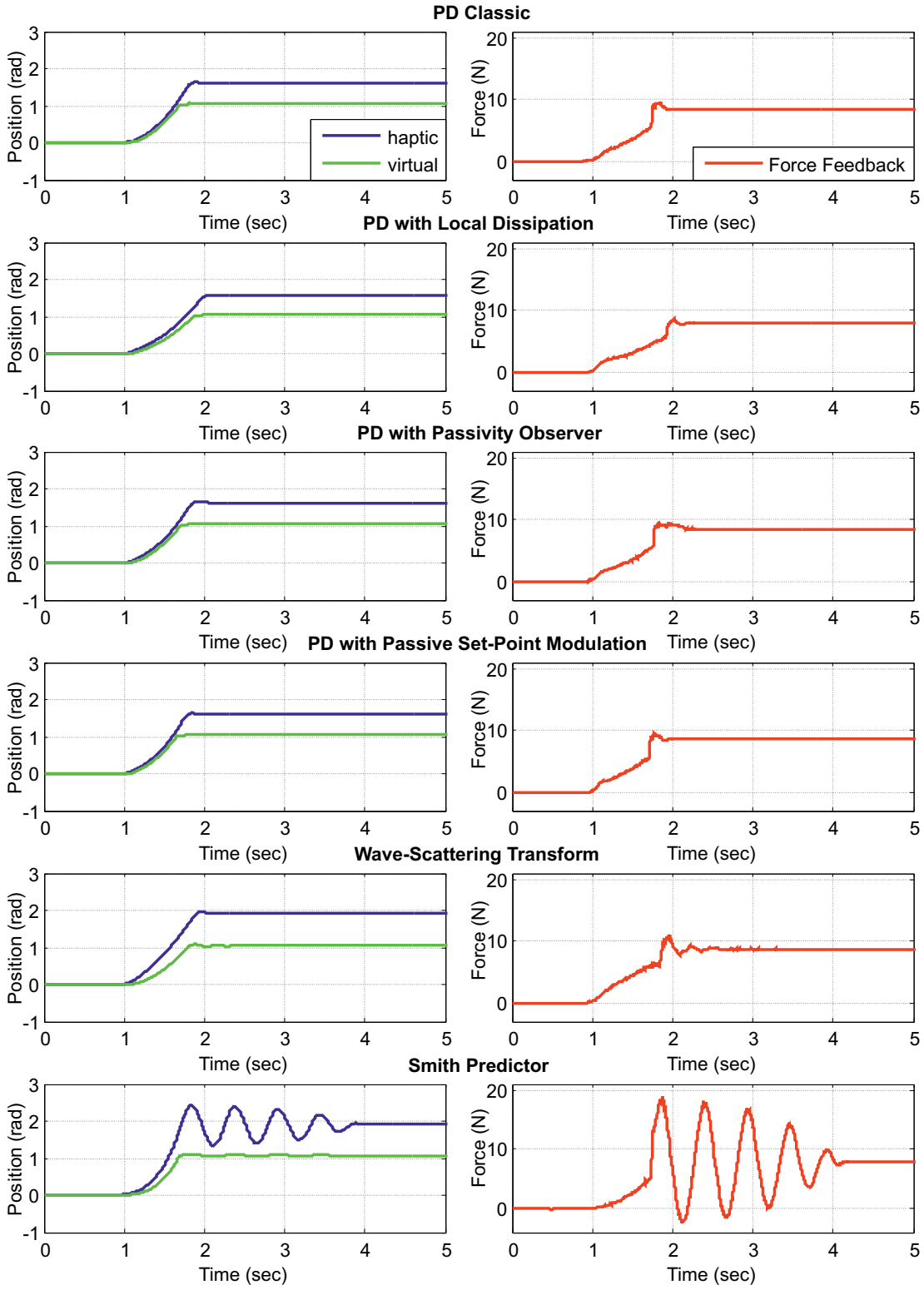


FIGURE 1.21: Optimal error tracking - restricted motion, 50 ms delay.

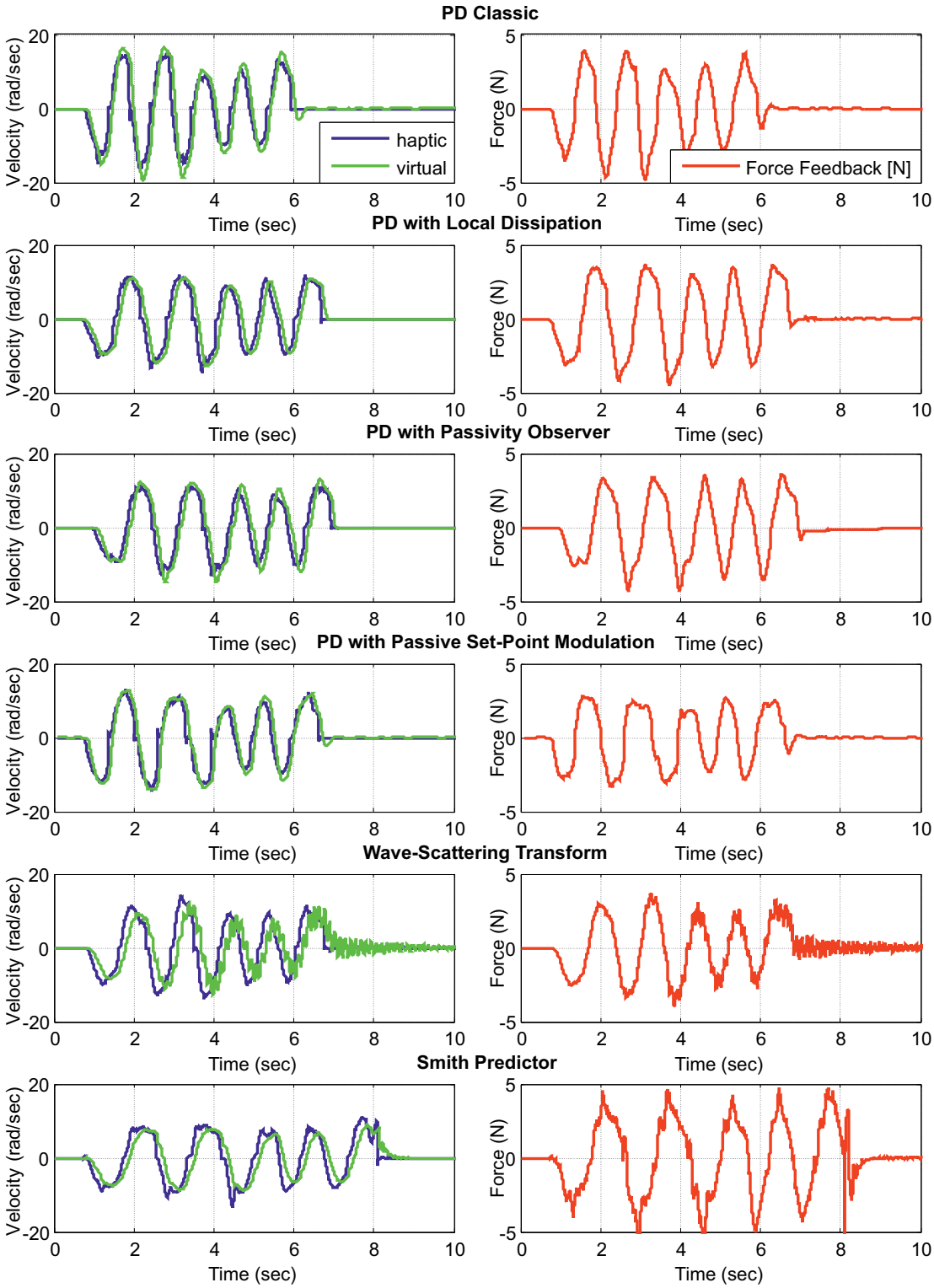


FIGURE 1.22: Optimal viscosity effect - free motion, 50 ms delay.

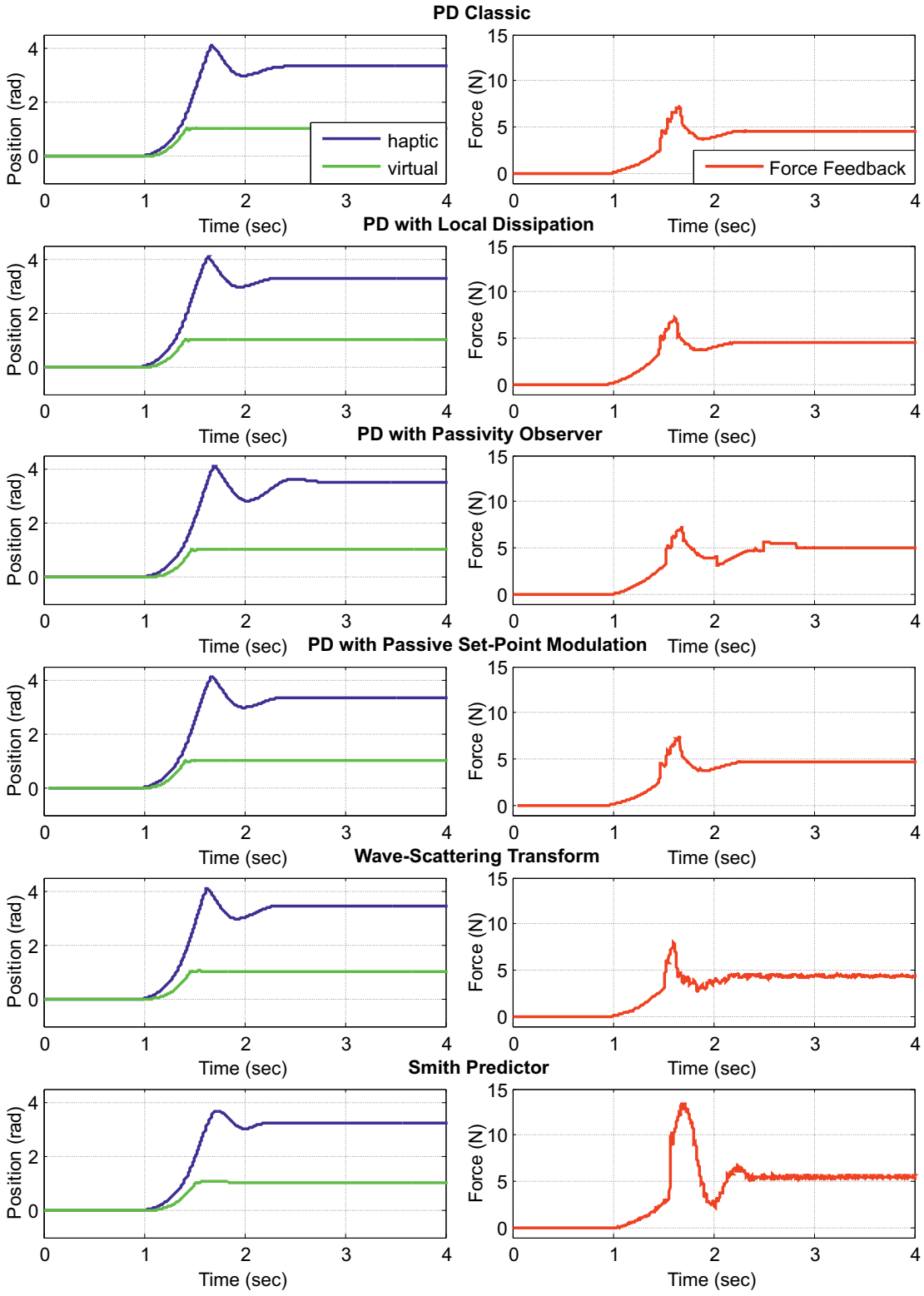


FIGURE 1.23: Optimal viscosity effect - restricted motion, 50 ms delay.

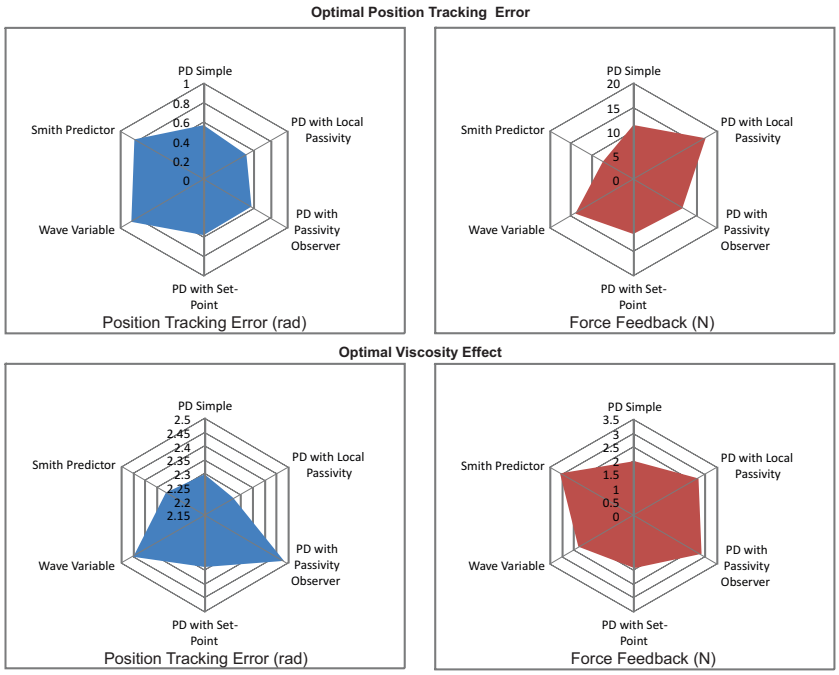


FIGURE 1.24: Position tracking error and force feedback performance in the case of 50 ms delay.

According to the experimental results, the differences between the methods do not appear as being significant. It is obvious, that for the first five methods the stability is guaranteed, furthermore the passivity observer and the set-point methods provide an additional theoretical guarantee. The Smith predictor method appear to be more sensitive in terms of stability. On the other hand, none of these methods can provide high degree of transparency and small position tracking error simultaneously. The compromise between transparency and position tracking error is obvious. The presence of time delays deteriorates the system's performances including "disturbing" effects as for example viscosity, which is directly connected to time delays.

For the **first case** study - optimal tracking error, the best method in terms of position tracking error is the PD control with local dissipation with an error of 0.512 rad, but in free motion the viscosity effect is significant (17.2 N). In our opinion, the best method, from the transparency point of view, is represented by the Smith Predictor, with a force feedback of 7.5 N. It is important to mention that in terms of position tracking error the result is not completely satisfactory (0.832 rad).

The *best* "trade-off" between transparency and position tracking error is obtained by the PD control with set-point method with a position tracking error equal to 0.569 rad and a force feedback of 11.2 N.

For the **second case** study - optimal viscosity effect, which is more significant in the case of haptics (the interest to have a low viscosity effect in free motion and a stiff response in case

of hard contact appears to be more important than to have small position tracking error), the best performance, i.e. the smallest force feedback, is obtained for the PD control with set-point modulation method (1.9 N) and a position tracking error of 2.336 rad. The smallest position tracking error is guaranteed by the PD control with local dissipation (2.271 rad) like in the first case.

The *best* "trade-off" between transparency and position tracking error is assured by the classic PD control with a position tracking error equal to 2.298 rad and a force feedback of 2 N.

It is worth mentioning that some of the methods (wave variables in the second case for free and restricted motion, PD with passivity observer in first case for restricted motion and Smith predictor in the second case for restricted motion) induce significant noise on the responses.

Generally speaking, even if one is able to see small differences between the methods, the optimal tuning for one case will deteriorate the performance for the other one.

1.6 Conclusions

In this chapter the most common control methods used in teleoperation systems and further applied to haptics have been analyzed. In haptics, similar to the teleoperation, a high degree of transparency and a small position tracking error are desired simultaneously. However, these two conditions cannot be obtained in the same time by none of the presented methods. However, based on our benchmark example, the best experimental results in the haptics context are obtained for the Smith predictor and PD control with set-point modulation method.

In [155] an algorithm to switch between optimal position tracking error and optimal viscosity is proposed with satisfactory results. Such an idea inspired us a gain scheduling method that is proposed and discussed in detail in Chapter 4.

The next chapter introduces the theoretical tools needed for analyzing the stability of the haptic systems in the presence of time-delays. More precisely, a method for analyzing the stability and transparency, a fragility analysis for PD controllers, and finally a complete analysis for the Smith predictor-based controllers used in haptics are proposed.

Chapter 2

Methods of Analysis

2.1 Introduction

As seen in the previous chapter, one of the key issues in haptic systems is the influence of communication delays on the global system performances.

Therefore, this chapter presents some of the theoretical tools needed in the analysis of the stability of the delayed systems in different configurations, as well as some physical limitations and tuning clues from a practical point of view. More precisely, the chapter starts with an *extremely* short overview of existing Proportional-Integral-Derivative (PID) control methods in the literature of delayed systems. Next, the Proportional-Derivative (PD) controllers under time-delays used in haptics will be analyzed and discussed. Further on, a specific analysis for the Smith predictor-based control is proposed. The last section introduces the limitations of a real system and practical hints for tuning the controller's gains. Some concluding remarks end the chapter.

As mentioned before, a *classical* problem in teleoperated and haptic systems is represented by the stability problem. Once this problem is solved, the second step is to provide the desired performances for the system. In haptics, as in teleoperation, the goal is to obtain the tracking position between the haptic interface and the virtual object, as well as a high degree of transparency in free and restricted motions.

Section 2.2 presents a short overview on PID controllers and the motivation of choosing the PD control for haptics as well as some stability considerations related to this controller.

The remaining chapter is organized as follows: a short overview of existing PID control methods is presented in Section 2.2. Next, in Section 2.3, the Proportional-Derivative (PD) controllers under time-delays used in haptics are particularly analyzed and discussed. First, in Subsection 2.3.1, using classical tools, the closed-loop stability analysis of practical bilateral haptic systems

coupled with a virtual environment by using a standard proportional derivative (PD) control law is addressed. The delays in the communication channels are assumed to be constant and, as it will be explain later, only the *overall delay* (the sum of the forward and backward delays) needs to be known. Continuing the PD controllers analysis, Subsection 2.3.2 presents the *fragility* analysis for single-input-single-output (SISO) systems affected by input (or output) delays. For the development of controllers for real environments, a careful analysis has to considered in order to take into account the variation of the parameters. Generally, real systems present parameter variations which often lead to an unstable behavior of the overall system. The non-fragile PD-controller choice is presented in a frequency domain setting by using some relatively simple geometric arguments. In order to prove the efficiency of the method some general illustrative examples are presented. The last subsection, 2.3.3, is devoted to the uncertain delays case, by considering some specific distribution *modeling*. Generally, the time-delay can be studied in order to determine the variation shape and based on, a distribution model is proposed, which can be further included in the control law. The delay distribution aims describing the delay variation in a stochastic manner and capturing its average behavior. Considering distributed time-delays, it appears that the accuracy of the model is improved in the proposed frame, uniform distribution and gamma distribution with gap will replace the constant time-delays. Using a geometrical representation, some examples from haptics are presented in order to illustrate the stability results in this case.

As it was previously mentioned (Chapter 1), a classical approach for time-delays system remains the Smith predictor and its variants. Section 2.4 presents a complete stability analysis of the Smith predictor-based control scheme for haptic systems. The stability analysis is proposed from two points of view: stability regions in controller's gain parameters and stability in the corresponding delay parameters-space (in the case when the communication delay assumed constant, is subject to modeling errors or uncertainty). In the first case, for fixed or predefined (distributed) time-delays, an allowable stability region for the tuning gains of PD controller ($K_p,\ K_d$) is drawn. In the second case, once the tuning parameters of the PD controller ($K_p,\ K_d$) are derived, a study of the variation dependency between the system's delay and the Smith predictor's delay is proposed. Next, based on the same principles as in the case of standard PD controllers, a fragility analysis of the controllers is proposed in the case of Smith predictor-based scheme. Subsection 2.4.6 is devoted to the analysis of Smith predictor control in the case of uncertain delays. Multiple delay uncertainties will be addressed: starting from the case when the uncertainty is fixed, then uniform distribution and gamma distribution with gap will be analyzed, and finally, using a mean stability approach, the normal distribution case will be presented. In order to point out the main ideas, illustrative examples from haptics are inserted during the presentation .

Finally, Section 2.5 presents some practical guide lines for the tuning strategies of the controllers as well as some system's limitations. Some concluding remarks end the chapter.

2.2 PID control for time-delay systems. An overview.

According to [89], the PID (or its variations P, PI or PD) controllers can be used to control the majority of the closed-loop Single Input / Single Output (SISO) systems. This controller can be easily implemented in industrial applications using pneumatic, hydraulic, mechanical or electronic/software devices. A PID controller consists of three *elements*, each one corresponding to the proportional, integral and derivative actions. The standard form of a PID controller in the Laplace domain is given by:

$$C(s) = P + I + D = K_p + \frac{K_i}{s} + K_d s, \tag{2.1}$$

where K_p, K_i and K_d are generally called the proportional, integral and derivative gains respectively. In the time domain, the PID controller is given by:

$$c(t) = K_p e(t) \; + \; K_i \int e(t)dt \; + \; K_d \frac{de(t)}{dt}, \tag{2.2}$$

where $e(t)$ represents the controller input. As mentioned in [199], the PID controller takes into account the past information through the integral action, the present information by the use of proportional gain and finally the future is roughly *predicted* by the derivative action.

Generally, a *simple* P controller can stabilize only first-order unstable systems. For such a controller, in order to decrease the steady state error large, gains have to be used. Next, the integral actions is introduced to eliminate steady state errors and to reduce the oscillations. The integral action slows the system response, while large perturbations and noises are easily tolerated. Finally, the derivative action improves the system performance wherever fast responses are needed, since the error prediction strategy improves the response speed. In terms of applications, the P controller is not frequently used, while the PI controller is often used in industrial application specially for processes where the a fast response is not required. As mentioned, the PD controller is used for fast response requirements, particularly for (fast) moving objects and other systems with high dynamics.

The tuning of the PID controllers can be done by using various approaches such as Ziegler-Nichols, analytical, optimized or auto tuning methods (see, for instance [9, 16, 17, 41]). As presented in [138, 199], since the implications of each one of the three parameters is complex, some general guidelines must be taken into account:

- The proportional term (P) provides an immediate action in the control signal based on the error. Generally, a large gain will guarantee a fast response, as well as a small static error. However, a too large gain may cause the actuators saturations. Since this a static error based method, it is impossible to eliminate completely the steady state error.

Controller type	K_p	K_i	K_d
P	$0.5K_u$		
PI	$0.45K_u$	$\frac{1}{1.2}T_u$	
PID	$0.6K_u$	$0.5T_u$	$0.125T_u$

TABLE 2.1: Ziegler-Nichols tuning.

- The integral action (I) eliminates the steady state error for step inputs as well as in case of perturbations. Due to the error accumulation from the past, the integral action slows the system, which may easy result in an overshoot response. In order to minimize the overshots, the a smaller gain should be chosen. In [16], a wind-up effect corresponding to the actuators' saturation is presented, as well as some anti-wind-up strategies.

- The derivative parameter provides a fast reaction of the system based on the predicted evolution of the error. The slow response and the overshoots cause by the integral action, as well as the stability of the system are improved by the derivative gain. It is worth mentioning that increasing too much the D-gain, will also result in an overshoot response.

Ziegler-Nichols method

A very common method used for tuning the PID controllers is represented by the so-called the Ziegler-Nichols method [200]. This method was proposed for first-order-plus-dead-time systems given by:

$$G(s) = \frac{K}{Ts+1}e^{-s\tau}, \tag{2.3}$$

where K represents the static gain of the plant, $T > 0$ is the time constant and τ is the system delay or dead time. The algorithm consists in three steps:

1. The plant is set under P-controller with a small gain for a step reference.

2. Next, the P-gain is increased until the system starts oscillating. The P-gain value corresponding to the oscillating frequency will be retained (K_u) as well as the oscillating period (T_u), known as ultimate gain and ultimate frequency respectively.

3. The last step consists in setting the controller gains depending on its structure. Table 2.1 resumes the values.

The main idea of this method is to find the proportional gain K_u for which the system is critically stable (crosses the critical point $(-1,0)$) and based on this estimation to tune-up the PID gains. Even if the resulting response is often oscillating, the obtained gains represent a good starting point and reference for further developments.

Analytical tuning based on gain and phase margins

Many circumstances require a specific gain tuning in order to achieved the desired gain and phase margins so that the system is robustly stable. The gain margin is defined as follows:

$$A_m = \frac{1}{|C(j\omega_p)G(j\omega_p)|},$$

where C, G represent the controller and the delayed plant respectively, and ω_p satisfies:

$$arg|C(j\omega_p)G(j\omega_p)| = -\pi.$$

Further on, the phase margin ϕ_m of the system is defined as:

$$\phi_m = arg|C(j\omega_g)G(j\omega_g)| + \pi,$$

where the gain crossover frequency ω_g satisfies:

$$|C(j\omega_g)G(j\omega_g)| = 1.$$

According to [75], for given gain and phase margins (A_m, ϕ_m), the K_p, K_i gains of PI controller are given by:

$$K_p = \frac{T\omega_p}{A_m K}, \qquad K_i = \left(2\omega_p - \frac{4\omega_p^2\tau}{\pi} + \frac{1}{T}\right)^{-1},$$

where:

$$\omega_p = \frac{A'_m phi_m + 0.5\pi A_m(A_m - 1)}{(A_m^2 - 1)\tau}.$$

The parametric approach

According to [24], during the 1960's and 1970's, the stability problem under large parameter uncertainty was almost completely ignored in the control literature. The work of Kharitonov from 1978, with the apparition of the so-called Kharitonov theorem may be seen as a starting point of the studies devoted to real parametric uncertainty. Furthermore, [24] mention that the first notable result following Kharitonov work was obtain by [175], and it refers to the computation of the largest stability *ball* in the polynomial coefficients space around a given point. After this more powerful results were obtained by [25, 29], including the generalized Kharitonov theorem (GTK) proved by [32].

The fundamental role in most of the results on robust stability is played by the Boundary Crossing Theorem. In order to introduce this theorem the following assumption is made:

Assumption 1. $P(\lambda, s)$ is a family of polynomials of:

1. fixed degree n (invariant degree),

2. continuous with respect to λ on some fixed interval $I = [a,b]$.

Theorem 2.1. *(Boundary Crossing Theorem) Under the Assumption 1, suppose that $P(a,s)$ has all its roots in S whereas $P(b,s)$ has at least one root in U. Then, there exists at least one ρ in $(a,b]$ such that:*

(a) $P(\rho,s)$ has all its roots in $S \cup \partial S$, and:

(b) $P(\rho,s)$ has at least one root in ∂S.

Consider now the quasipolynomial family:

$$Q(s,\lambda) = d(s,\lambda) + e^{-sT_1}n_1(s,\lambda) + e^{-sT_2}n_2(s,\lambda) + \ldots + e^{-sT_m}n_m(s,\lambda),$$

where $\lambda \in [a,b]$ and assume that $Q(s,a)$ is Hurwitz, $Q(s,b)$ is not Hurwitz, $deg[d(s,\lambda)] = n,\ \forall \lambda \in [a,b]$ and $deg[n_i(s,\lambda)] < n, \forall \lambda \in [a,b]$. In the case of quasipolynomials the Boundary Crossing Theorem becomes:

Theorem 2.2. *Under the above assumptions, there exists at least one ρ in $(a,b]$ such that $Q(0,p) = 0$ or $Q(j\omega,\rho) = 0$ for some $\omega \in [-\infty,\ +\infty]$*

In the case of PID control, the idea to define the stability regions in the controller parameters-space was already encountered in [168], where they used the Hermite-Biehler Theorem. It is worth mentioning that such a method is computationally involved even for the simplest case of time-delay systems.

In [120], a different method for deriving the stability regions in the gain parameters space for PID controllers used in SISO systems with I/O delays was proposed. The stability *crossing boundaries* represent the collection of all points for which the corresponding characteristic equation of the closed-loop system has at least one root on the imaginary axis. Based on the D-decomposition method [127] or the parametric approach [24, 168], the authors construct the crossing curves or surfaces which define appropriate regions in the corresponding parameter space having the property that for all the points of the region, the number of unstable characteristic roots takes the same value. The results are presented using a geometrical approach in the 3D space, each dimension corresponding to a one of the PID gains (proportional, integral and derivative gains). If the closed-loop system is stable for a specific set of control parameters $(K_p,\ K_i, K_d)$, then the system is stable for any point in that region. Any crossings from the stable region to a neighbor one will result in a loss of stability since at least one pair of characteristic roots will *move* from the left to the right-half plane.

There exists several analytical tests that can be applied for the stability problem mentioned above. Among them, without any loss of generality, we may cite: Pontryagin criterion [144,

145], Chebotarev-Meiman [33], Yesipovich-Svirskii ([178] for the formulation). Most of them are computationally involved and apply for single delay case. In particular, the Chebotarev criterion appears as being the most direct and natural generalization of the Routh-Hurwitz criterion for quasi-polynomials. However, its application as an analytical criterion is not effective in practice, since an infinite number of Hurwitz determinants must be explicitly computed. Furthermore, it is worth mentioning that for long delays in the system, the high dimension determinants may need to be investigated. Finally, further discussions on the parameter based approach can be found in [116]. Among the most recent results, we wish to point out the contributions of: Morărescu and Niculescu [118], Ramirez *et al.* [150] and Sipahi *et al.* [170].

The haptic case

In haptic systems, potentially conflicting requirements as a fast response and robust stability need a control analysis in order to take the right decision, since these systems are subject to frequent position changes. It is worth pointing out that such systems have error based functionality, since the entire feedback is based on the tracking error of the position and velocity. Under these circumstances, the most appropriate control solution is to use a PD-based control, since the integral action slows down the response and *tries* to obtain zero error.

In terms of tuning, it is well known that haptic systems must be tunned according to the end user perception, respecting the stability constrains. With these considerations, the most appropriate design method in this case seems to be the approach proposed in [120]. The reasons for such a choice are twofold: first, the algorithm is extremely simple to implement, and second, it allows to depict all the stability regions in the corresponding controller-parameter space. As byproduct of the analysis, the geometric approach allows a simple characterization of the controller's *fragility* (how *far* or *close* from the instability are the chosen gains positioned). Based on the same ideas as in [120], as well as [115], Section 2.3.2 proposes a specific development for analyzing the stability of the PD controllers under time-delays. As a general strategy for haptic system, in this thesis the stability regions will be computed in the first step, and in the second step, the tuning will be made in order to obtain the best results in terms of end user performances respecting the stability constrains.

2.3 Stability of Proportional Derivative (PD) controllers used in haptics under time-delays

In the case of haptic systems, except the communication channel, delays may appear as intrinsic components of the processing time for the virtual reality environment. The presence of time-delays may induce instabilities and complex behaviors. More precisely, in free motion, the delay effect can be felt by the viscosity phenomenon (high force feedback felt at the haptic interface

end) and in the case of "hard"-contacts with the environment, the force feedback is not stiff, as explained and discussed in the previous chapter.

As mentioned previously, this thesis does not focus on the estimation of the delays in the loop. As a consequence, we preferred to take into account several delay configurations: constant and distributed delays over some time-intervals but offering the possibility of some unitary treatment.

In the sequel, the closed-loop stability analysis of some class of practical bilateral haptic systems coupled with a virtual environment by using a standard proportional derivative (PD) control law is addressed. The proposed analysis points out an interesting remark: only the overall delay (the sum of the forward and backward delays) needs to be known.

The analysis starts by assuming that the communication delays are *constant* and not necessarily the forward and backward delays separately.

2.3.1 Constant time-delays

As mentioned in the previous paragraphs, the ideal haptic system should satisfy simultaneously the following conditions: the position tracking error has to be as small as possible between the haptic interface and the virtual object, and the system has to have a high degree of transparency (in the "free" motion case, the force feedback felt at the haptic interface end must be as small as possible and in the case of a "hard"-contact, a stiff response is desired).

Figure 2.1 presents the general control scheme of a haptic interface and a virtual environment including control feedback.

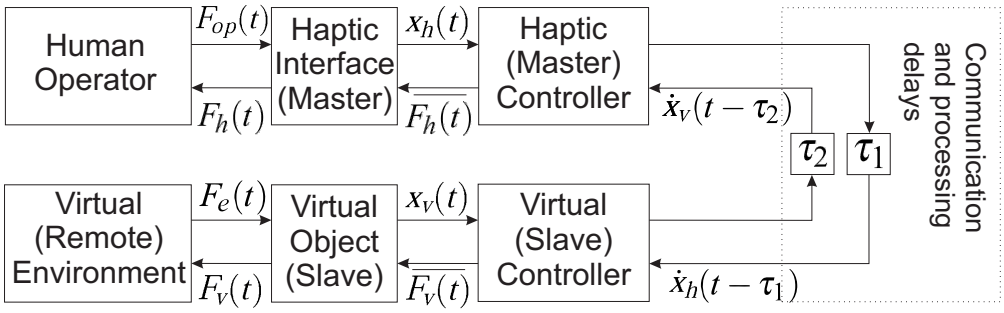


FIGURE 2.1: General PD control scheme for haptic systems.

Here, $\overline{F_{h/v}}$ represents the calculated command corresponding to a specific human operator/environmental force. This value is converted into force by the master/slave robots (or haptic interface/virtual object).

As mentioned in the previous chapter, the starting point is given by the *classical* dynamic (non-linear) equations of motion for two similar robots equations (1.1)-(1.2). The *main* idea is to use two PD controllers, one to control the haptic interface (master robot) and another one for the virtual object (slave robot). The corresponding controllers's equations (1.3)-(1.4) are then recalled:

$$F_h(t) = \underbrace{K_{d_h}(\dot{x}_h(t) - \dot{x}_v(t-\tau_2))}_{\text{delayed D-action}} + \underbrace{K_{p_h}(x_h(t) - x_v(t-\tau_2))}_{\text{delayed P-action}}, \tag{2.4}$$

$$F_v(t) = \underbrace{K_{d_v}(\dot{x}_h(t-\tau_1) - \dot{x}_v(t))}_{\text{delayed D-action}} + \underbrace{K_{p_v}(x_h(t-\tau_1) - x_v(t))}_{\text{delayed P-action}}, \tag{2.5}$$

where τ_1, τ_2 denote the forward and backward finite constant delays respectively and $K_{p_{h/v}}, K_{d_{h/v}}$ are the PD control gains.

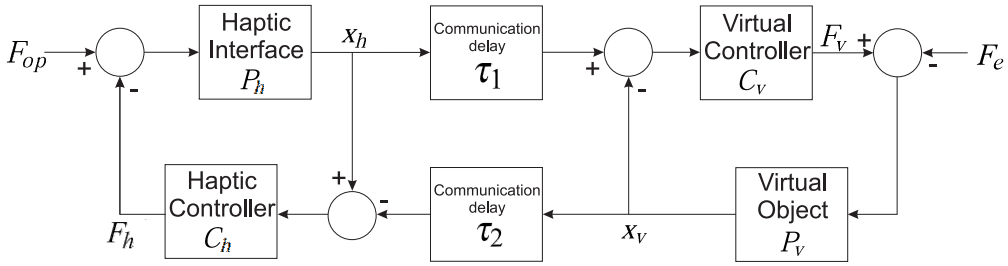


FIGURE 2.2: Bilateral Haptic System.

From Figure 2.2, the equations describing the system response can be written as follows:

$$X_h(s) = P_h(s)\left(F_{op}(s) - C_h(s)\left(X_h(s) - e^{-\tau_2 s}X_v(s)\right)\right), \tag{2.6}$$

$$X_v(s) = P_v(s)\left(-F_e(s) + C_v(s)\left(-X_v(s) + e^{-\tau_1 s}X_h(s)\right)\right), \tag{2.7}$$

where $X_{h/v}(s)$ denotes the Laplace transform of the time signal $x_{h/v}(t)$; similarly, for $F_{op}(s)$ and $F_e(s)$. The robot axis are considered to be decoupled, i.e. the matrices from (1.1)-(1.2) are considered to be diagonal. The transfer functions $P_{h/v}(s)$[1] and $C_{h/v}(s)$ are taken as follows (position available for measurement and PD structure for the control law):

$$P_h(s) = P_v(s) = \frac{1}{s(ms+b)} =: P(s), \tag{2.8}$$

$$K_{p_h} = K_{p_v} =: K_p, \quad K_{d_h} = K_{d_v} =: K_d, \tag{2.9}$$

$$C_h(s) = C_v(s) = K_p + K_d s =: C(s), \tag{2.10}$$

with m representing the mass, and b the viscous friction coefficient. The robots are modeled as linear systems since the haptic interface does not presents any particular behaviors that are not covered by the linear model, and the virtual robot is represented by an ideal case.

[1]The haptic interface as well as the virtual object are modeled to be second order spring-mass-damper systems.

As far as the internal stability analysis is concerned, the above system is equivalent to a system where the controller is PI (of the form $K_d + K_p/s$), and the process (measured) variable is represented by the velocity, i.e., process given by: $(ms+b)^{-1}$.

By rearranging (2.6) and (2.7) above, the following equation is obtained :

$$\begin{bmatrix} 1+P_h(s)C_h(s) & -P_h(s)C_h(s)e^{-\tau_2 s} \\ -P_v(s)C_v(s)e^{-\tau_1 s} & 1+P_v(s)C_v(s) \end{bmatrix} \begin{bmatrix} X_h(s) \\ X_v(s) \end{bmatrix} = \begin{bmatrix} P_h(s)F_{op}(s) \\ -P_v(s)F_e(s) \end{bmatrix}. \tag{2.11}$$

Therefore, with the process (plant) and controller definitions (2.8) and (2.10), the characteristic equation of the feedback system rewrites as follows:

$$(1+P(s)C(s))^2 - (P(s)C(s))^2 e^{-(\tau_1+\tau_2)s} = 0, \tag{2.12}$$

which is simply equivalent to:

$$\chi_1(s)\,\chi_2(s) = 0, \tag{2.13}$$

where:

$$\chi_1(s) := \left(1+P(s)C(s)+P(s)C(s)e^{-\tau s}\right),$$

$$\chi_2(s) =: \left(1+P(s)C(s)-P(s)C(s)e^{-\tau s}\right),$$

and $\tau := \frac{(\tau_1+\tau_2)}{2}$.

Remark 2.3. An analysis of equations of the form (2.13) has been given in [166] with χ_1, χ_2 scalar quasipolynomials. However, the authors did not considered a more general case study. Different approaches for handling such a control problem can be found in [161] (closed-loop stability analysis in the controller-gains parameter space), [116] and the references therein (optimal delay bound as a function of parameters). In the sequel, the stability analysis of such a feedback system will be performed, using classical tools from control theory (such as gain and phase margins).

The following result is obtained (see Appendix C for the proof):

Theorem 2.4. *The bilateral haptic system is asymptotically stable independent of the delay values (τ_1, τ_2) if the controller gains satisfy the condition:*

$$K_d \geq \frac{m}{b} K_p. \tag{2.14}$$

Furthermore, when $K_d/K_p < m/b$, there exist two cases:

(a) If $0 < mK_p - bK_d < b^2/2$, then the feedback system is stable independent of the delay values (τ_1, τ_2).

(b) If $mK_p - bK_d > b^2/2$*, then the closed-loop system is stable if and only if*

$$mK_p - bK_d < \frac{b^2}{2}(1+\omega_0^2), \tag{2.15}$$

where $\omega_0 > 0$ *is the solution of the equation:*

$$\frac{\pi - 2\left(\tan^{-1}(x) - \tan^{-1}\left(\frac{b\,K_d}{m\,K_p}x\right)\right)}{x} = \frac{(\tau_1+\tau_2)b}{2m}. \tag{2.16}$$

□

From the conditions of Theorem 2.4, the allowable range of $m\,K_p/b^2$ and K_d/b for all $b/m > 0$ can be explicitly determined. The corresponding stability region is shown for three different time delay values in Figure 2.3.

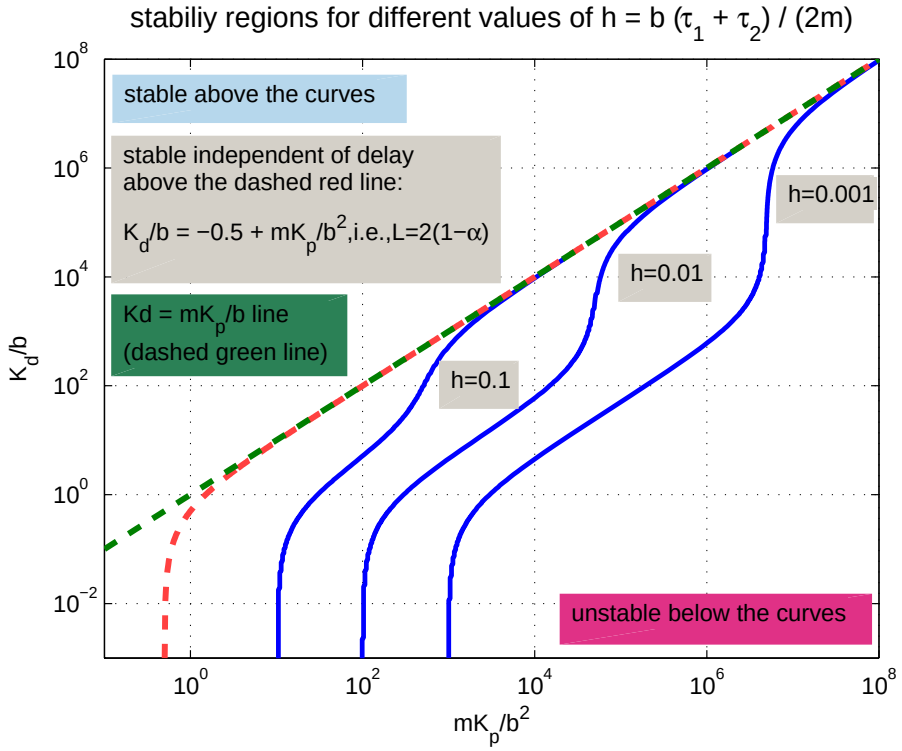


FIGURE 2.3: Allowable region of controller parameters for stability of the bilateral haptic system for $m = 1\ Kg$, $b = 0,1\ Ns/m$ and $\tau_1 = \tau_2 = 10,\ 100,\ 1000\ ms$.

Remark 2.5. In [136] a *PD-like* controller is proposed, using only the position error in the controller (instead of using both - position and velocity errors) in order to guarantee the passivity of the system. The velocity of the haptic interface/virtual object will be used with the D-gain for computing the controls for the haptic and virtual side, respectively. With this assumption, equations (2.4)-(2.5) are rewritten as follows:

$$F_h(t) = \underbrace{K_{d_h}\dot{x}_h(t)}_{\text{D-action}} \underbrace{+K_{p_h}(x_h(t) - x_v(t-\tau_2))}_{\text{delayed P-action}}, \tag{2.17}$$

$$F_v(t) = \underbrace{-K_{d_v}\dot{x}_v(t)}_{\text{D-action}} \underbrace{+K_{p_v}(x_h(t-\tau_1) - x_v(t))}_{\text{delayed P-action}}, \tag{2.18}$$

Based on the complete development from Appendix D, the stability is guaranteed if the following condition holds:

$$\frac{K_d}{K_p} > \tau \quad \Leftrightarrow \quad K_d > K_p\tau. \tag{2.19}$$

The result derived in [136]:

$$K_{d_h}\, K_{d_v} > K_{p_h} K_{p_v} \tau_1 \tau_2,$$

is exactly the same as (2.19), under the assumption (2.9) and $\tau_1 = \tau_2 = \tau$. □

2.3.2 Fragility of PD controllers

Consider now the analysis of the fragility of Proportional-Derivative (PD) controllers for SISO systems affected by input (or output) delays.

It is worth mentioning that the problem received a lot of attention in the case of systems free-of delay, see for example [85] (robustness techniques design leading to fragile controllers), [74] (non-fragile PID control design procedure), [4] (appropriate index to measure the fragility of PID controllers), but the delay case was not sufficiently addressed. A *simple method* to analyze the fragility of a given PD-controller is developed, for any SISO strictly proper system affected by input/output (constant) time-delays (see also Appendix B for different architecture). The results are presented by using a *geometrical approach*.

The proposed method consists of two steps. More precisely, the first step uses the D-decomposition method suggested by Neimark [127] (see [39, 116] for further comments) in order to derive and compute the boundaries of the stability regions. In this context, the *stability crossing curves* represent the collection of all points for which the corresponding characteristic equation of the closed-loop system has at least one root on the imaginary axis. These curves define a partition of the parameters space in several regions, each region having a constant number of unstable roots for all the parameters inside the region. By taking into account the crossing boundaries characterization in the controller parameter-space, the second step consists in deriving an algorithm to determine explicitly the *optimal non-fragile* controller. The presented algorithm allows computing explicitly the (closed-loop) stability radius in the controller parameter space. Finally, as a by product of the analysis, the maximum controller gain interval guaranteeing the closed-loop stability can be easily derived.

2.3.2.1 Preliminaries

Consider now a generic class of strictly proper SISO open-loop system with I/O delays given by the transfer function:

$$H_{yu}(s) = \frac{M(s)}{N(s)} e^{-s\tau} = c^T (sI_n - A)^{-1} g e^{-s\tau}, \tag{2.20}$$

where $(A,\ c^T,\ g)$ is a state-space representation of the open-loop system ($A \in \mathbb{R}^{n\times n}$, $c, g \in \mathbb{R}^n$). The control law is defined by a classical PD controller $K(s)$ of the form:

$$K(s) = k(1 + T_d s) = k_p + k_d s. \tag{2.21}$$

Remark 2.6. For analytical reasons, the case $k_p = 0$ will not be explicitly considered, since such a case does not have a practical counterpart, since standard derivative controllers are not applicable.

Therefore, the stability of the closed-loop systems is given by the locations of the zeros of the following meromorphic function $H : \mathbb{C} \times \mathbb{R}^2 \times \mathbb{R}_+ \mapsto \mathbb{C}$ given by:

$$H(s; k_p, k_d, \tau) = 1 + \frac{M(s)}{N(s)} (k_p + k_d s) e^{-s\tau}, \tag{2.22}$$

which has an infinite (but, countable) number of roots (see e.g., [57, 62]).

The aim of this study is to derive an appropriate PD controller $(k_p^*,\ k_d^*)$ and a positive value[2] d such that the control law (2.21) stabilizes the system (2.20) for any k_p and k_d as long as:

$$\sqrt{(k_p - k_p^*)^2 + (k_d - k_d^*)^2} < d. \tag{2.23}$$

Without any loss of generality, the following assumption is made:

Assumption 2. The polynomials $M(s)$ and $sN(s)$ in (2.22) do not have common zeros.

In the sequel, some geometric results are recalled in order to enable the computation of the stability crossing curves in the space defined by the controller's parameters $(k_p,\ k_d)$ (similar results for different types of dynamics can be found in [58] - delay parameters space and [122, 125] - some particular class of distributed time-delays). These curves represent the collection of all pairs $(k_p,\ k_d)$ for which the characteristic equation (2.22) has at least one root on the imaginary axis of the complex plane.

According to the continuity of zeros with respect to the system's parameters (see, for instance, [98] for the continuity with respect to time-delays), the number of roots in the right half plane

[2] d represents the *minimum distance* to the instability region.

(RHP) can change only when some zeros appear and cross the imaginary axis [116]. Therefore, a useful concept is the *frequency crossing set* Ω defined as the set of all real positive ω for which there exists at least a pair (k_p, k_d) such that:

$$H(j\omega; k_p, k_d, \tau) = 1 + \frac{M(j\omega)}{N(j\omega)}(k_p + k_d j\omega)e^{-j\omega\tau} = 0. \tag{2.24}$$

Only positive frequencies ω are needed to be considered, that is $\Omega \subset (0, \infty)$ since obviously,

$$H(j\omega; k_p, k_d, \tau) = 0 \iff \overline{H(-j\omega; k_p, k_d, \tau)} = 0. \tag{2.25}$$

Proposition 2.7. *For a given $\tau \in \mathbb{R}_+$ and $\omega \in \Omega$ the corresponding crossing point (k_p, k_d) is given by:*

$$k_p = -\Re\Big(\frac{N(j\omega)}{M(j\omega)}e^{j\omega\tau}\Big), \tag{2.26}$$

$$k_d = -\frac{1}{\omega}\Im\Big(\frac{N(j\omega)}{M(j\omega)}e^{j\omega\tau}\Big). \tag{2.27}$$

Remark 2.8. It is easy to see that $\forall\ \omega \in \Omega$ the following inequality holds $M(j\omega) \neq 0$. Otherwise, $N(j\omega) = 0$, that contradicts Assumption 2.

Proposition 2.9. *Let the relative degree of the system (2.20) be $\rho = 1$. Then, the closed-loop system (2.22) becomes of Neutral-Type*[3] *and,*

$$\begin{cases} k_p = \left|\frac{q_n}{p_{n-1}}\right| \\ k_p = -\left|\frac{q_n}{p_{n-1}}\right| \end{cases},$$

belong to the stability crossing boundary. Here p_{n-1} and q_n are the main coefficients of the polynomials $M(s)$ and $N(s)$, respectively:

$$M(s) = \sum_{i=1}^{n-1} p_i s^i, \quad N(s) = \sum_{i=1}^{n} q_i s^i.$$

Remark 2.10. Observe that in the case when $m = n-1$, the corresponding closed-loop system is a quasipolynomial of *neutral* type (see, for instance, [129] for further discussions on the topics).

Proposition 2.11. *[123]: Let k_d^* and $k_d^* > 0$ be given. Let $\Omega_{k_p^*, k_d^*}$ denotes the set of all frequencies $\omega > 0$ satisfying equation (2.24) for at least one pair of (k_p, k_d) in the rectangle $|\, k_p \,| \leq k_p^*, |\, k_d \,| \leq$*

[3] Neutral-Type defines a system where the denominator's degree is equal to the numerator's degree, more details can be found in [129].

k_d^*. *Then* $\Omega_{k_p^*,k_d^*}$ *consists of a finite number of intervals of finite length.*
Precisely, $\omega \in \Omega_{k_p^*,k_d^*}$ *if and only if:*

$$\left|\frac{N(j\omega)}{M(j\omega)}\right|^2 \leq (k_p^*)^2 + (k_d^*)^2\omega^2. \tag{2.28}$$

Then, when ω varies within some interval Ω_l satisfying the inequality (2.28), (2.26)-(2.27) define a continuous curve. Denote this curve by $\mathcal{T}_l$ and consider the following decompositions:

$$R_0 + jI_0 = j\frac{\partial H(s;k_p,k_d,\tau)}{\partial s}\Big|_{s=j\omega}, \tag{2.29}$$

$$R_1 + jI_1 = -\frac{\partial H(s;k_p,k_d,\tau)}{\partial k_d}\Big|_{s=j\omega}, \tag{2.30}$$

$$R_2 + jI_2 = -\frac{\partial H(s;k_p,k_d,\tau)}{\partial k_p}\Big|_{s=j\omega}. \tag{2.31}$$

The implicit function theorem [59] indicates that under appropriate assumptions, $(k_p,\ k_d)$ can be expressed locally as functions of the frequency "ω". Furthermore, the tangent of $\mathcal{T}_l$ can be expressed as follows:

$$\begin{pmatrix} \frac{dk_p}{d\omega} \\ \frac{dk_d}{d\omega} \end{pmatrix} = \begin{pmatrix} R_2 & R_1 \\ I_2 & I_1 \end{pmatrix}^{-1} \begin{pmatrix} R_0 \\ I_0 \end{pmatrix} = \frac{1}{R_2I_1 - R_1I_2} \begin{pmatrix} R_0I_1 - R_1I_0 \\ R_0I_2 - R_2I_0 \end{pmatrix}, \tag{2.32}$$

provided that

$$R_1I_2 - R_2I_1 \neq 0. \tag{2.33}$$

In order to derive the stability region of the system given by (2.22), [123] characterized the smoothness of the crossing curves and the corresponding direction of crossing.

Proposition 2.12. *The curve* $\mathcal{T}_l$ *is smooth everywhere except possibly at the point corresponding to* $s = j\omega$*, if it is a multiple solution of* (2.22).

2.3.2.2 Direction of Crossing

This subsection focuses on the characterization of the crossing direction corresponding to the curves defined by (2.26)-(2.27). Here, the notion of *positive direction* denotes the direction of the curve that corresponds to the increasing of ω. Also, *the region on the left* refers to the region on the left hand side following the positive direction of the curve.

Proposition 2.13. *Assume $\omega \in \Omega_l$, k_p, k_d satisfy (2.26) and (2.27) respectively, and ω is a simple solution of (2.25) and:*

$$H(j\omega'; k_p, k_d, \tau) \neq 0, \forall \omega' \neq \omega, \tag{2.34}$$

i.e. (k_p, k_d) is not an intersection point of two curves or different sections of a single curve. Then, as (k_p, k_d) moves from the region on the right to the region on the left of the corresponding crossing curve, a pair of solution of (2.20) crosses the imaginary axis to the right (through $s = j\omega$) if:

$$R_1 I_2 - R_2 I_1 > 0. \tag{2.35}$$

The crossing is to the left if the inequality is reversed.

Any given direction, (d_1, d_2), is to the left-hand side of the curve if its inner product with the left-hand side normal $\left(-\frac{\partial k_d}{\partial \omega}, \frac{\partial k_p}{\partial \omega}\right)$ is positive, i.e.:

$$-d_1 \frac{\partial k_d}{\partial \omega} + d_2 \frac{\partial k_p}{\partial \omega} > 0, \tag{2.36}$$

from which the following result is obtained:

Corollary 2.14. *Let ω, k_p and k_d satisfy the same condition as Proposition 2.13. Then as (k_p, k_d) crosses the curve along the direction (d_1, d_2), a pair of solutions of (2.22) crosses the imaginary axis to the right if:*

$$d_1(R_2 I_0 - R_0 I_2) + d_2(R_1 I_0 - R_0 I_1) > 0. \tag{2.37}$$

The crossing is in the opposite direction if the inequality is reversed.

All these properties will be further used to examine the fragility concept of PD controllers.

2.3.2.3 Fragility of PD controllers

Consider now the PD fragility problem, which is the problem of deriving the maximum controller parameters deviation without losing the closed-loop stability. More precisely, for a given pair of parameters (k_p^*, k_d^*), the roots of the equation:

$$N(s) + M(s)(k_p + k_d s)e^{-s\tau} = 0, \tag{2.38}$$

must be located in $\mathbb{C}_-$ (this is the case when the closed-loop system is asymptotically stable). Next, the problem is to find the maximum parameter deviation $d \in \mathbb{R}_+$ such that the roots of (2.22) stay located in $\mathbb{C}_-$ for all controllers (k_p^*, k_d^*) satisfying:

$$\sqrt{(k_p - k_p^*)^2 + (k_d - k_d^*)^2} < d. \tag{2.39}$$

This problem can be more generally reformulated as: *find the maximum parameter deviation d such that the number of unstable roots of* (2.22) *remains unchanged.*

First, let us introduce some notation:

$$\mathcal{T} = \bigcup_{l=1}^{N} \mathcal{T}_l, \qquad \mathcal{T}_l = \{(k_p, k_d) | \omega \in \Omega_l\}, \tag{2.40}$$

$$\tilde{k}(\omega) = (k_p(\omega), k_d(\omega))^T, \qquad \tilde{k}^* = (k_p^*, k_d^*)^T. \tag{2.41}$$

Let us also denote $d_{\mathcal{T}} = \min\limits_{l \in \{1,\ldots,N\}} d_l$, where:

$$d_l = min\Big\{ \sqrt{(k_p - k_p^*)^2 + (k_d - k_d^*)^2} | (k_p, k_d) \in \mathcal{T}_l \Big\}. \tag{2.42}$$

With the notation and the results above, the following proposition may be written:

Proposition 2.15. *The maximum parameter deviation from* (k_p, k_d)*, without changing the number of unstable roots of the closed-loop equation* (2.22) *can be expressed as:*

$$d = min\left\{ k_{d\infty}, |k_p^* - k_p(0)|, \min_{\omega \in \Omega_f} \left\{ \left\| \tilde{k}(\omega) - \tilde{k}^* \right\|_2 \right\} \right\}, \tag{2.43}$$

where

$$k_{d\infty} := \begin{cases} \min\left\{ \left| k_d^* - \left| \frac{q_n}{p_m} \right| \right|, \left| k_d^* + \left| \frac{q_n}{p_m} \right| \right| \right\} & \text{if} \quad m = n-1 \\ \emptyset & \text{if} \quad m < n-1 \end{cases}$$

and Ω_f is the set of roots of the function $f : \mathbb{R}_+ \mapsto \mathbb{R}$,

$$f(\omega) \triangleq \left(\tilde{k}(\omega) - \tilde{k}^* \right) \ \cdot \ \frac{d\tilde{k}(\omega)}{d\omega}, \tag{2.44}$$

where "$\cdot$" represents the dot product.

Proof: The pair (k_p^*, k_d^*) is considered as belonging to a region generated by the crossing curves. Since the number of unstable roots changes only when (k_p, k_d) get out of this region, the objective is to compute the distance between (k_p^*, k_d^*) and the boundary of the region. Furthermore,

the boundary of such a region consists of "pieces" of crossing curves and possibly the segments of the shifted axis:

$$\begin{aligned} k_p : &= k_p + \left|\frac{q_n}{p_m}\right|, \quad \text{and} \\ k_p : &= k_p - \left|\frac{q_n}{p_m}\right|, \end{aligned}$$

(for neutral-type systems) and a segment of the shifted axis: $k_d + k_p(0)$.

In order to compute the distance between (k_p^*, k_d^*) and a crossing curve, the points where the vector $(k_p - k_p^*, k_d - k_d^*)$ and the tangent to the curve are orthogonal must be identified.

In other words, the solutions of the following equation must be found:

$$f(\omega) = 0,$$

where f is defined by (2.44).

Taking into account the relation (2.32), (2.44) may be written as:

$$f(\cdot) = (k_p - k_p^*)(R_1 I_0 - R_0 I_1) - (k_d - k_d^*)(R_0 I_2 - R_2 I_0). \tag{2.45}$$

It is noteworthy that $f(\omega)$ is a polynomial function and, therefore, it will have a finite number of roots. Let us consider $\{\omega_1, ..., \omega_M\}$ the set of all the roots of $f(\omega)$ when all the pieces of crossing curves belonging to the region around (k_p^*, k_d^*) are taken into account. Since the distance from (k_p^*, k_d^*) to the $k_p(\omega)$ axis is given by $|k_d|$, one obtains:

$$d = min\left\{k_{d\infty}, |k_p^* - k_p(0)|, \min_{h=1,...,M}\left\{\left\|\tilde{k}(\omega_h) - \tilde{k}^*\right\|\right\}\right\}, \tag{2.46}$$

that is just another way to express (2.43).

The explicit computation of the maximum parameter deviation d can be summarized by the following algorithm:

Step 1: First, compute the "degenerate" points of each curve $\mathcal{T}_l$ (i.e. the roots of $R_1 I_2 - R_2 I_1 = 0$ and the multiple solutions of (2.22)).

Step 2: Second, compute the set Ω_f defined by Proposition 2.15 (i.e. the roots of equation $f(\omega) = 0$, where f is given by (2.44).

Step 3: Finally, the corresponding maximum parameter deviation d_l is defined by (2.42).

Remark 2.16. *(On the gains' optimization):* It is worth mentioning that the geometric argument above can be easily used for solving other robustness problems. Thus, for instance, if one of

the controller's parameters is fixed (prescribed), the maximum interval guaranteeing closed-loop stability with respect to the other parameter can be also explicitly computed. In particular if K_d ("derivative") is fixed, the corresponding stabilizing maximum proportional gain interval can be derived.

2.3.2.4 Some Illustrative Examples

In order to illustrate the theory of the *fragility* of the controllers, this section presents some numerical examples.

Example 2.1. *(Gantry crane)*

For this example, a gantry crane was chosen, used as slave robot in a teleoperation system as suggested by [49], having a time delay of 2 seconds:

$$g(s) = \frac{40s^2 + 2s + 400}{200s^3 + 30s^2 + 2401s + 200} e^{-2s}. \tag{2.47}$$

According to Proposition 2.7, Figure 2.4 presents the corresponding crossing curves for (2.47).

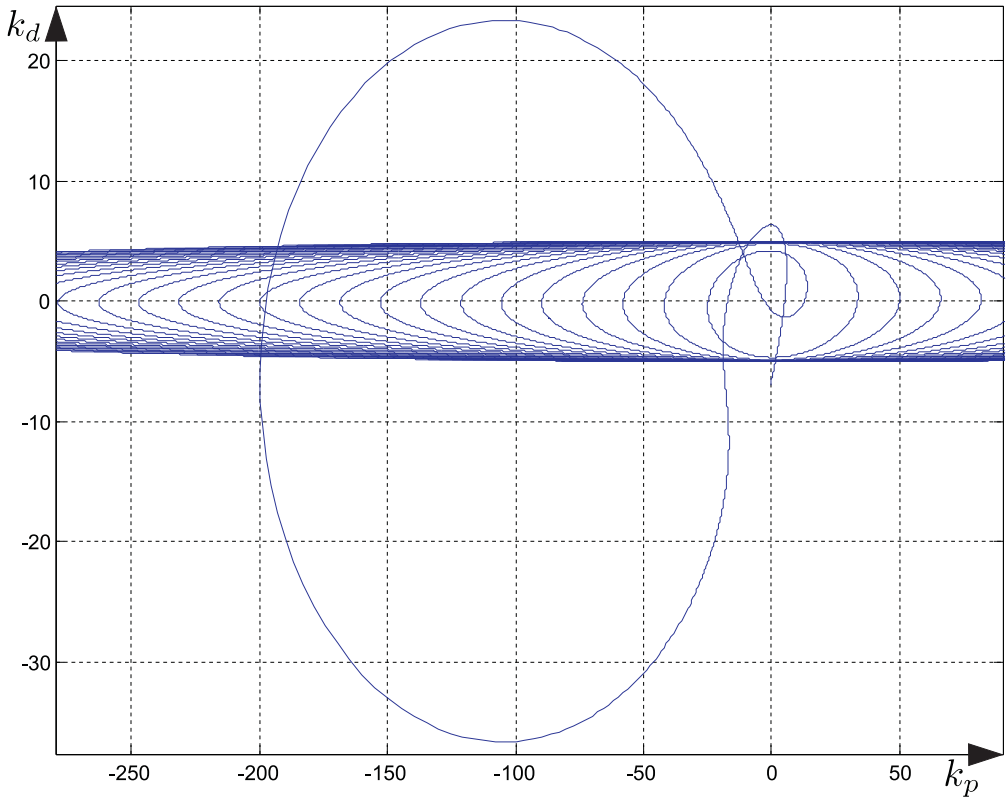


FIGURE 2.4: Corresponding crossing curves for the gantry crane example in the controllers' parameter space.

Remark 2.17. Based on the figure, the stability regions must be searched. A classical way to do it is follows: start at the origin of the parameter space. In principle, the system is sufficiently simple in order to decide the system's stability. Next, define a path connecting the origin to the point we are interesting in. In general, we try to have the simplest one: for instance starting from the origin and taking the one of the axis (coordinates) as being the first one and so on. It is worth mentioning that each intersection between such a path and the stability crossing curve will correspond to some change in the stability of the system. Once a stable point is found, the entire region where the point is located is stable. When passing from stable region to an unstable one, at least two roots cross in the right half of the plane ($\mathbb{C}_-$). Further on, the next regions may be stable or unstable. As a general rule, when passing from one region to another, a pair of two complex roots will cross either in $\in \mathbb{C}_-$, either in $\in \mathbb{C}_+$, making the system stable or instable. Two *neighbor* regions cannot be stable, since at every passing two roots must cross from $\mathbb{C}_-$ to $\mathbb{C}_+$ or viceversa.

Figure 2.5 illustrates the stability region for the system (2.47) as well as the maximum parameter deviation for the controller (k_p^*, k_d^*).

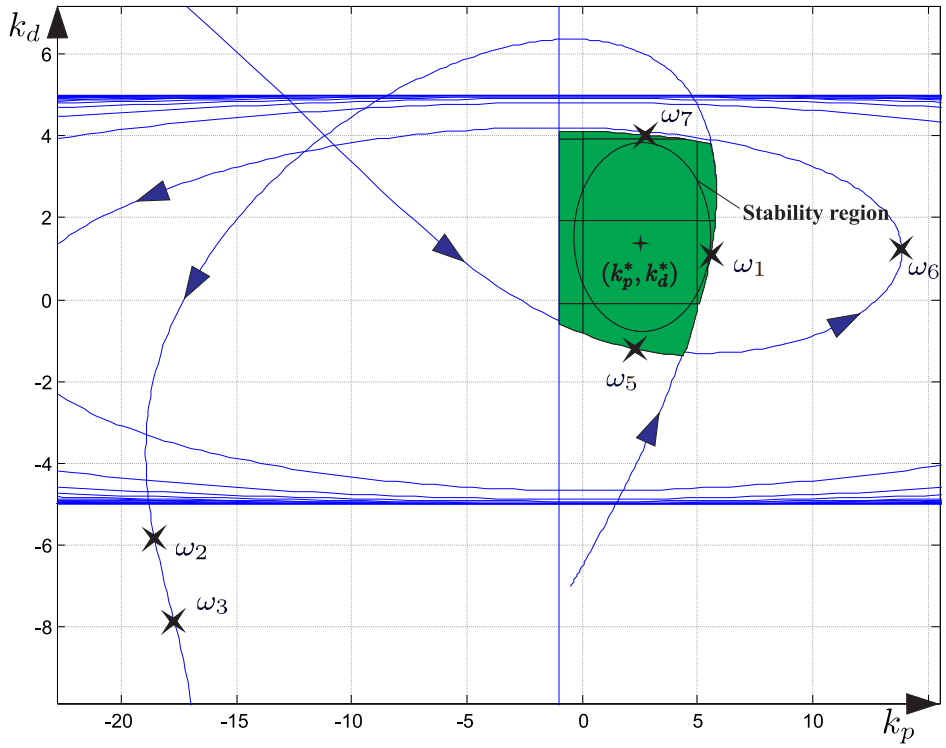


FIGURE 2.5: Stability region for the crane model (2.47); non-fragile controller given by: $k_p^* = 3.25$, $k_d^* = 1.65$, $d = 2.345980$.

In table E.1, from Appendix E the results obtained are summarized after applying the proposed algorithm in order to analyze the fragility for the controller $(k_p^*, k_d^*) = (3.25, 1.65)$.

Example 2.2. *(Sixth order non-minimal phase system): For the second example, the following process was considered [20]:*

$$g(s)=\frac{-s^4-7s^3-2s+1}{(s+1)(s+2)(s+3)(s+4)(s^2+s+1)}e^{-\frac{s}{20}}. \tag{2.48}$$

Figure 2.6 presents the stability region in k_p, k_d for (2.48).

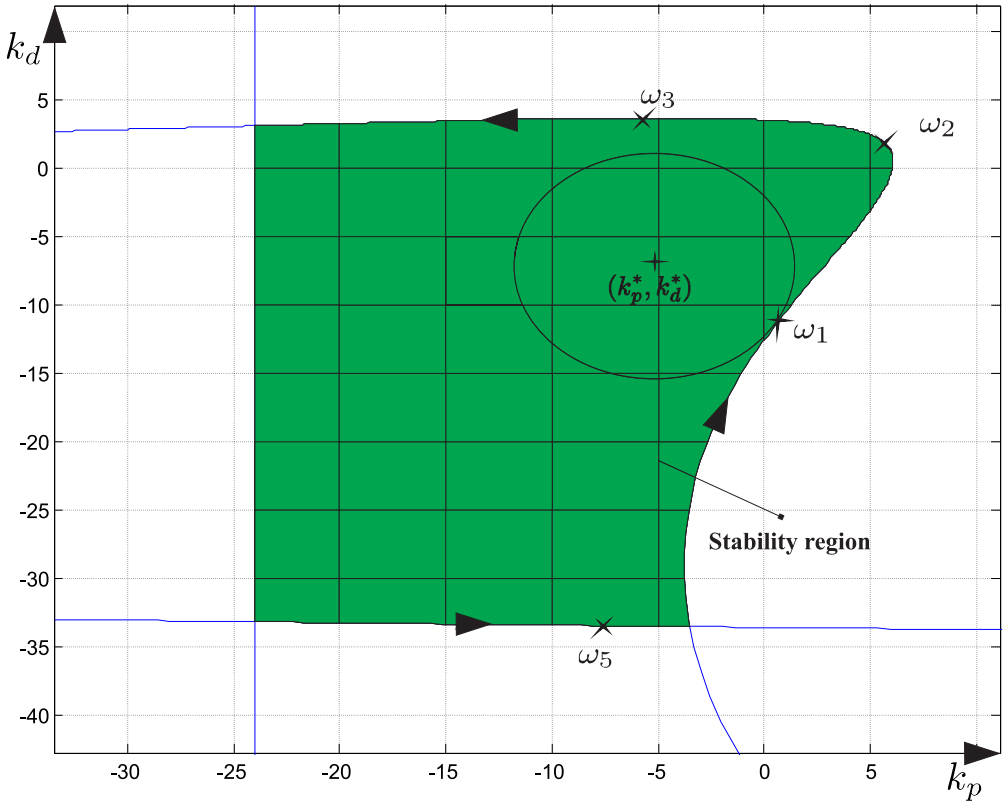


FIGURE 2.6: Stability region of $(k_p,\ k_d)$ for the example (2.48); non-fragile controller given by: $k_p^* = -6.15$, $k_d^* = -6.25$, $d = 8.2917$.

The fragility analysis for the PD-controller for the system (2.48) are summarized in table E.2 from Appendix E.

Example 2.3. *(Unstable, non-minimum phase): Consider a second-order, non-minimum-phase and unstable open-loop system, described by the following transfer function [115]:*

$$g(s) = \frac{s-2}{s^2-1/2s+13/4}e^{-\frac{1}{2}s}. \tag{2.49}$$

The particularity of this system is that the use of a PD-controller leads to a closed-loop system of neutral type. In Figure 2.7 the stability region in the (k_p, k_d) parameter space was depicted for the system (2.49).

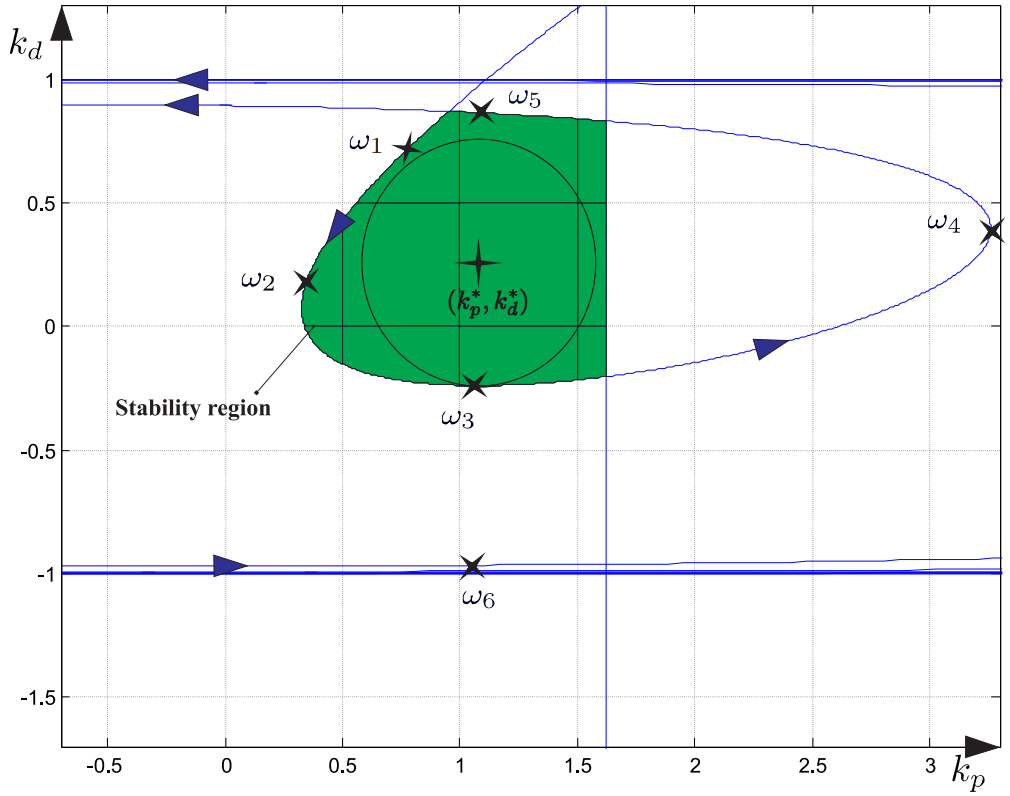


FIGURE 2.7: Maximum parameter deviation $d = 0.4910$ for the controller $(k_p^*, k_d^*) = (1.1021, 0.2514)$ for the unstable non-minimum phase example (2.49).

Table E.3 from Appendix E summarizes the fragility analysis for the controller $(k_p^*, k_d^*) = (1.1021, 0.2514)$ *for the system (2.49).*

In this section, a simple method for choosing a non-fragile PD-controller was presented for a class of strictly proper SISO systems with I/O time-delays by using geometric arguments. To prove the efficiency of the method, several illustrative examples have been considered.

The next subsection addresses the distributed time-delays problem. More precisely, the time-delays are considered to be varying and an appropriate analysis is proposed by using the properties of the delay distribution.

2.3.3 Distributed time-delays

A potential approach when dealing with varying time-delay problems is represented by the use of distributed delays as potential models in depicting delays behavior. Generally, the time-delay can be studied in order to determine the variation shape and based on it, a distribution model is created, which can be further included in the control law. Basically, the use of distributed time-delays is a reasonable trade-off between the *less likely* case of fixed time-delays and the

more complicated time-varying delay. Considering distributed time-delays, the accuracy of the model appears to be improved in the cases considered here without giving any overview of the literature on distributed delays. Such systems are present in many scientific fields such as economy [86], population dynamics [42], traffic control [83, 170], biological systems [109], network/Internet®-based systems [188, 193], etc.

In the sequel, the study is focused on the configuration from Figure 2.2, under the assumption that the delay is distributed. Under this assumption, the *simple* delay blocks ($\tau_1,\ \tau_2$) will be replaced by some appropriate functional delay blocks ($D_1(s),\ D_2(s)$). The new scheme is represented in Figure 2.8.

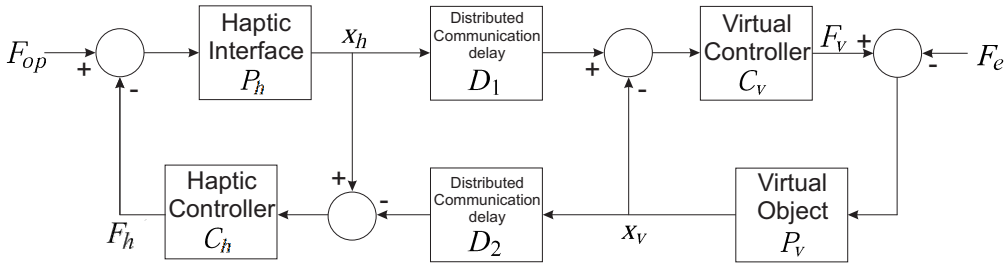


FIGURE 2.8: Bilateral Haptic System for distributed time time-delays.

From Figure 2.8, the equations describing the system response can be written as follows:

$$X_h(s) = P_h(s)\left(F_{op}(s) - C_h(s)\left(X_h(s) - D_2(s)X_v(s)\right)\right), \tag{2.50}$$

$$X_v(s) = P_v(s)\left(-F_e(s) + C_v(s)\left(-X_v(s) + D_1(s)X_h(s)\right)\right), \tag{2.51}$$

where $X_{h/v}(s)$ denotes the Laplace transform of the time signal $x_{h/v}(t)$; similarly for $F_{op}(s)$ and $F_e(s)$. Here, $\tau_1 > 0$ and $\tau_2 > 0$ denote the corresponding (forward and backward) time-delays. As they were already defined for the fixed delay case, the transfer functions $P_{h/v}(s)$ and $C_{h/v}(s)$ are taken as:

$$P_h(s) = P_v(s) = \frac{1}{s(ms+b)} =: P(s), \tag{2.52}$$

$$C_h(s) = C_v(s) = K_p + K_d s =: C(s). \tag{2.53}$$

In this case, the characteristic equation of the system is given by:

$$(1 + P_h(s)C_h(s))(1 + P_v(s)C_v(s)) - (-P_h(s)C_h(s)D_2(s))(-P_v(s)C_v(s)D_1(s)) = 0 \tag{2.54}$$

Considering $P_h = P_v = P$ and $C_h = C_v = C$, equation (2.54) results in:

$$(1 + P(s)C(s) + P(s)C(s)D_1(s))(1 + P(s)C(s) - P(s)C(s)D_2(s)) = 0. \tag{2.55}$$

Similarly to the constant delays case, equation (2.55) is simply equivalent to:

$$\chi_1(s)\,\chi_2(s) = 0, \tag{2.56}$$

where:

$$\chi_1(s) := (1 + P(s)C(s) + P(s)C(s)D_1(s)),$$

$$\chi_2(s) =: (1 + P(s)C(s) - P(s)C(s)D_2(s)).$$

Next, two common time-delay distributions will be discussed and analyzed:

- uniform distribution,
- gamma distribution with gap [121].

2.3.3.1 Uncertain delays: uniform distribution

In this case, the uncertainty will be considered to have a random variation, with no additional information about. Figure 2.9 presents such an uniform delay distribution.

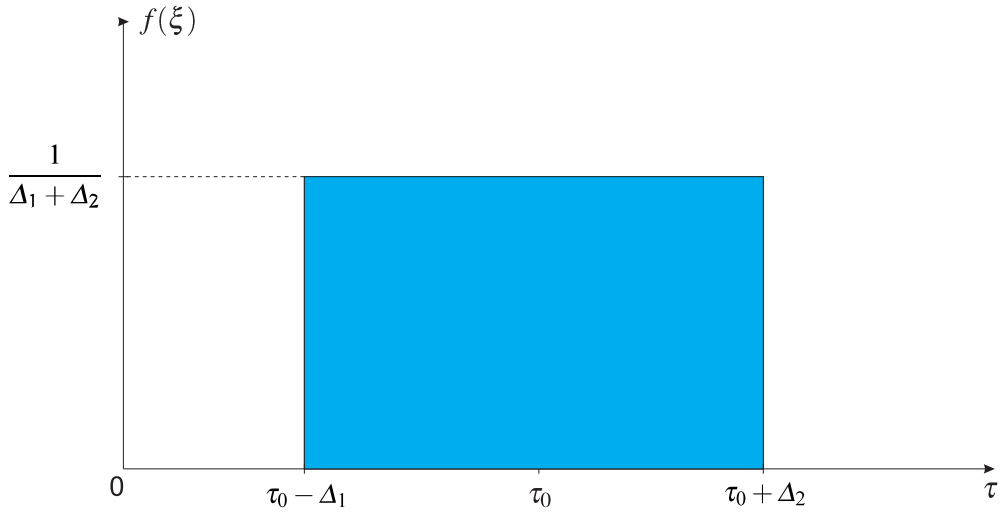


FIGURE 2.9: Uniform distribution.

Considering $\Delta_1, \Delta_2 > 0$ and $\tau_0 \geq \Delta_1$, the uniform distribution kernel is given by:

$$f(\xi) = \begin{cases} \frac{1}{\Delta_1+\Delta_2}, & \text{if } \tau_0 - \Delta_1 < \xi < \tau_0 + \Delta_2, \\ 0, & \text{otherwise}, \end{cases} \tag{2.57}$$

where τ_0 is the reference delay and Δ_1, Δ_2 represent the delay uncertainty in both directions.

The Laplace transform of the uniform distribution is given by:

$$D(s) = \frac{e^{-s(\tau_0-\Delta_1)} - e^{-s(\tau_0+\Delta_2)})}{s(\Delta_1+\Delta_2)} \tag{2.58}$$

With these considerations and considering $D_1 = D_2 =: D$, equations (2.50)-(2.51) rewrite as follows:

$$X_h(s) = H(s)\left(F_{op}(s) - C_{PD}(s)\left(X_h(s) - \frac{e^{-s(\tau_0-\Delta_1)} - e^{-s(\tau_0+\Delta_2)}}{s(\Delta_1+\Delta_2)}X_v(s)\right)\right), \tag{2.59}$$

$$X_v(s) = V(s)\left(-F_e(s) + C_{PD}(s)\left(-X_v(s) + \frac{e^{-s(\tau_0-\Delta_1)} - e^{-s(\tau_0+\Delta_2)}}{s(\Delta_1+\Delta_2)}X_h(s)\right)\right), \tag{2.60}$$

and the characteristic equation (2.54) for this case rewrites as follows (defined as a characteristic function):

$$\Psi(s,\, K_p,\, K_d,\, \tau_0,\, \Delta_1,\, \Delta_2) = \Psi_1(s,K_p,K_d) + \Psi_2(s,K_p,K_d)\left(\frac{e^{-s(\tau_0-\Delta_1)} - e^{-s(\tau_0+\Delta_2)})}{s(\Delta_1+\Delta_2)}\right)^2, \tag{2.61}$$

where:

$$\Psi_1(s,K_p,K_d) = m^2s^4 + (2K_dm+2bm)s^3 + (K_d^2+2K_db+b^2+2K_pm)s^2 + (2K_pb+2K_dK_p)s + K_p^2,$$

$$\Psi_2(s,K_p,K_d) = -K_d^2s^2 - 2K_dK_ps - K_p^2.$$

For the stability analysis in presence of time-delays the method presented in [105, 126] will be applied (this method was largely discussed in Subsections 2.3.2.3 as well as in the Appendix B, subsection B.2). Using a geometrical approach, this method draws the stability regions in the controller gains' space with respect to time-delays.

Method summary:

- The stability of the closed-loop system is given by the locations of the zeros of the characteristic equation. Introduce now the general form of characteristic function $\Psi : \mathbb{C}\times\mathbb{R}\times\mathbb{R}\times\mathbb{R}_+ \to \mathbb{C}$, used in the case of haptics (based on the scheme from Figure 1.4) given by:

  $$\Psi(s;K_p,K_d,\tau) = (1+P(s)(1+e^{-s\tau})(K_p+K_ds))(1+P(s)(1-e^{-s\tau})(K_p+K_ds)), \tag{2.62}$$

  which has an infinite (but, countable) number of roots (see, for instance [129]). In the case of uniform distributed delays, equation (2.62) is replaced by (2.61). It is worth mentioning that the roots of such a characteristic function are continuously dependent of the

system's parameters, and similarly to the constant delay case, a loss or gain of stability can be detected if characteristic roots cross the imaginary axis (see [152] or [116] and the references therein).

- Next, for a given $\tau \in \mathbb{R}_+^*$ (strictly positive), and for a frequency $\omega \in \mathbb{R}_+ \setminus \{k\pi/\tau;\, k \in \mathbb{N}_+\}$ the corresponding crossing points are given by:

$$\begin{cases} K_p^{\pm} = -\Re\left(\frac{1}{P(j\omega)(1\pm e^{-\tau j\omega})}\right) \\ K_d^{\pm} = -\frac{1}{\omega}\Im\left(\frac{1}{P(j\omega)(1\pm e^{-\tau j\omega})}\right) \end{cases}, \tag{2.63}$$

Otherwise:

$$\begin{cases} (K_p,\ K_d) = (K_p^+,\ K_d^+), & \text{for } \omega = \frac{2k\pi}{\tau}, \\ (K_p,\ K_d) = (K_p^-,\ K_d^-), & \text{for } \omega = \frac{(2k+1)\pi}{\tau} \end{cases}, \tag{2.64}$$

where (K_p^+, K_d^+) represents the solution set corresponding to the first part of (2.62), and (K_p^-, K_d^-) correspond to the second part.

- Then, when ω varies within some interval $\Omega_l \subset (0,\infty)$, (2.63)-(2.64) define appropriate curves that separate the gain-parameter space in several regions such that each region has the same number of unstable characteristic roots (see, for instance [116], for a more detailed discussion on the so-called D-decomposition method). As pointed out in Remark 2.17, an initial point must be chosen and checked in simulation if the closed-loop system is stable or not. Generally, the haptic systems are stable for small positive values of K_p, K_d (< 5). Further on, the main goal is to determine the shape of the stability region.

Based on the above method, the stability region for this case is presented in Subsection 2.3.3.3.

2.3.3.2 Uncertain delays: gamma distribution with gap

Studies on the problem of controlling objects over communication networks [157] pointed out that, in this case, the time-delays can be modeled by a gamma distribution with gap (see, for instance, [121] for further details). In the case of gamma-distribution with a gap, the kernel f is given by:

$$f(\xi) = \begin{cases} 0, & \xi < \tau, \\ \frac{(\xi-\tau)^{n-1} e^{-\frac{\xi-\tau}{T}}}{T^n (n-1)!}, & \xi \geq \tau, \end{cases} \tag{2.65}$$

where $n \in \mathbb{N}$, $T > 0$ and $\tau \geq 0$. Note that $f(\xi) \geq 0$ for all $\xi \geq 0$ and $\int_0^\infty f(\xi) d\xi = 1$. The gap is defined by τ, and the corresponding *average delay* of (2.65) satisfies:

$$\tau_m = \int_0^\infty \xi\, f(\xi)\, d\xi = \tau + nT. \tag{2.66}$$

Figure 2.10 presents an example of delay shape for a gamma distribution with gap.

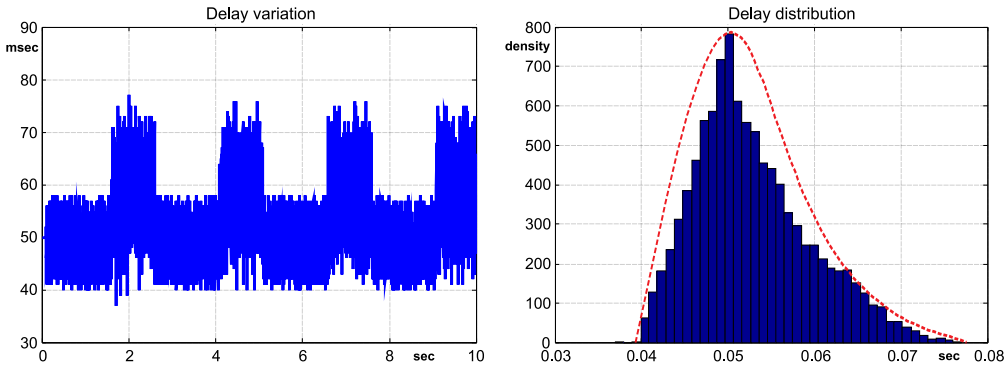


FIGURE 2.10: Example of gamma with gap distributed delay: shape and distribution ($\tau = 0.04$ *sec*, $T = 0.01$ *sec*, $n = 1$).

As $T \to 0_+$, the kernel (2.65) tends to a Dirac impulse centered at $\xi = \tau$ and one recovers the original system with a point-wise constant delay τ. The transition to $T = 0$ is smooth from a stability point of view, as the stability determining eigenvalues are continuous with respect to the parameter $T \geq 0$.

The Laplace transform of the gamma distribution with gap is given by:

$$D(s) = L(f) = \frac{e^{-s\tau}}{(1+sT)^n}. \tag{2.67}$$

In haptics, starting from equation (2.66), the delays' average values are given by:

$$\tau_1 = \tau_{1_m} = \hat{\tau}_1 + n_1 T_1, \quad \tau_2 = \tau_{2_m} = \hat{\tau}_2 + n_2 T_2, \tag{2.68}$$

where $\tau 1_m$, $\tau 2_m$ represent the average delay, and $\hat{\tau}_1$, $\hat{\tau}_2$ represent the corresponding delay's gap.

With these considerations, in the case of gamma distribution with gap, equations (2.6)-(2.7) rewrite as follows:

$$X_h(s) = H(s)\left(F_{op}(s) - C_{PD}(s)\left(X_h(s) - \frac{e^{-\hat{\tau}_2 s}}{(1+sT_2)^{n_2}} X_v(s)\right)\right), \tag{2.69}$$

$$X_v(s) = V(s)\left(-F_e(s) + C_{PD}(s)\left(-X_v(s) + \frac{e^{-\hat{\tau}_1 s}}{(1+sT_1)^{n_2}} X_h(s)\right)\right). \tag{2.70}$$

Considering $\tau_1 = \tau_2 = \tau + nT$ ($\hat{\tau}_1 = \hat{\tau}_2 = \tau$, $n_1 = n_2 = n$, $T_1 = T_2 = T$), the characteristic function for this case writes as follows:

$$\Psi(s,\ K_p,\ K_d,\ \tau,\ T,\ n) = \Psi_1(s, K_p, K_d) + \Psi_2(s, K_p, K_d)\left(\frac{e^{-\tau s}}{(1+sT)^n}\right)^2. \tag{2.71}$$

Based on the characteristic equation obtained above, the stability method previously presented will be applied, more precisely, equation (2.62) will be replaced by equation (2.71) in order to derive the stability zone for the gamma with gap distribution.

2.3.3.3 Illustrative examples

Considering the system presented in Figure 2.8, with $m = 1\ Kg$ and $b = 0.1\ Nsec/m$, the corresponding stability regions for the two cases are presented below.

Figure 2.11 depicts the stability region in (K_p, K_d) parameters' space for the case of uniform distribution with the characteristic equation (2.61), where $\tau_1 = \tau_2 = 0.05sec$ and $\Delta_1 = \Delta_2 = 0.01sec$.

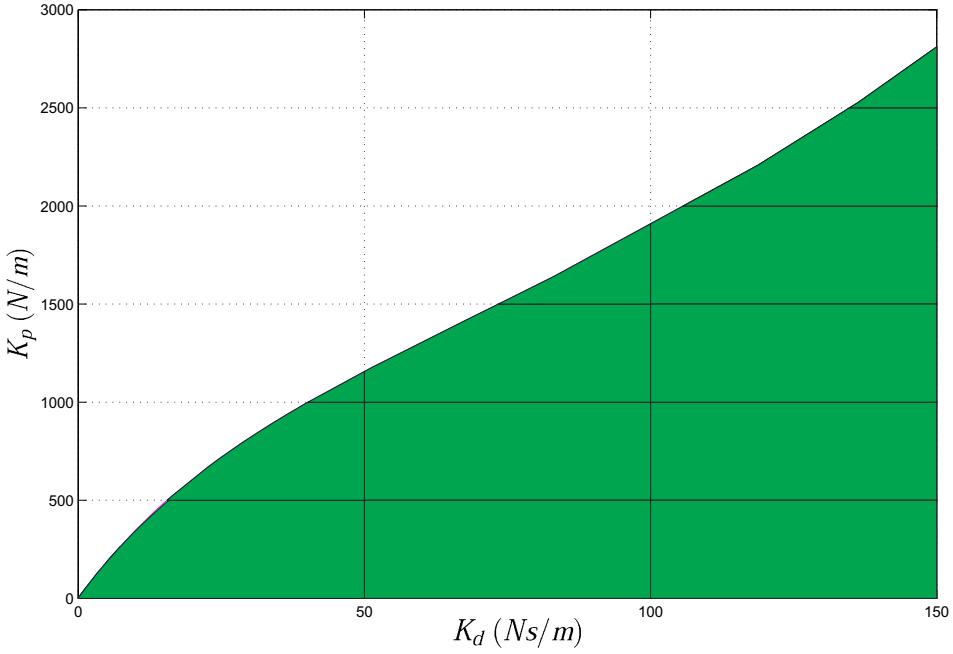


FIGURE 2.11: Stability region (in green) for (K_p, K_d) with $\tau_1 = \tau_2 = 0.05sec$ and $\Delta_1 = \Delta_2 = 0.01sec$.

Next, Figure 2.12 depicts the stability region in K_p, K_d parameters-space for the gamma with gap distributed time-delay, with the characteristic equation (2.71).

As it can be seen from the graphical representations, the stability region in the case of gamma distributed with gap delays is *larger*, which was expected since the uniform distributed delays offer a less restrictive choice in terms of delay variation. More precisely, in the uniform distributed case, only the minimum and maximum delays must be known, while for the gamma with gap distribution the delay must have a specific shape as shown in Figure 2.10.

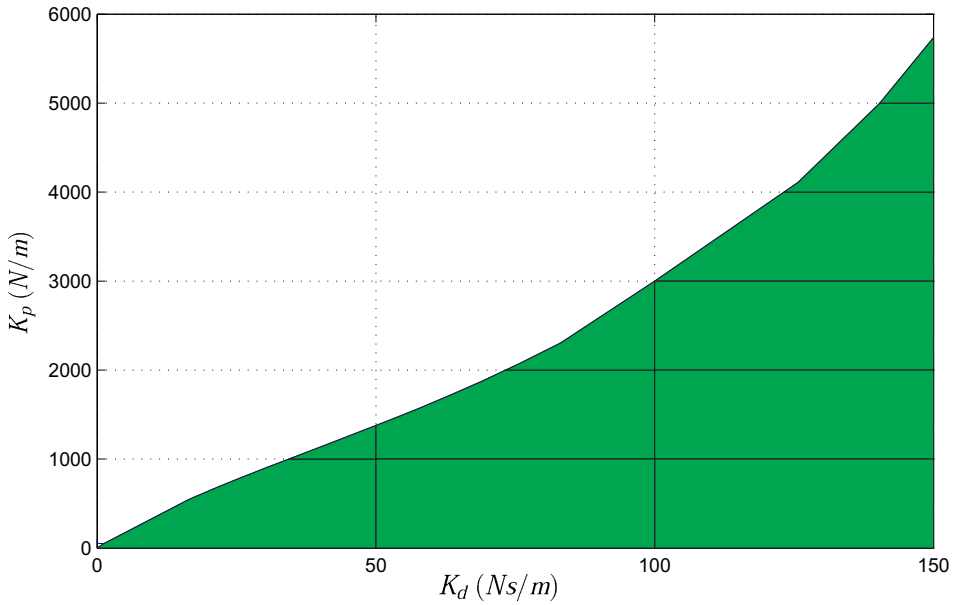


FIGURE 2.12: Stability region (in green) for (K_p, K_d) for a gamma distributed delay with gap ($\tau = 0.04\ sec$, $T = 0.01\ sec$, $n = 1$).

In this subsection, an analysis of PD controllers under uncertain time delay was presented. In the case of uncertain time-delays, a special attention is required for the behavior of the delay. Depending on this, the time-delays can be modeled in order to choose more efficiently the controller's parameters providing high performance in terms of end user perception and guarantying the system's stability. As it was presented, depending on the distribution type, the choice of controller's gains can be more or less restrictive. The next section presents the analysis of the Smith predictor-based controllers for haptic systems.

2.4 Analysis of Smith predictor-based controllers used in haptics

As mentioned a reliable method in solving time-delay problems remains the Smith predictor [174]. A complete study on the variations of the Smith predictor can be found in [137]. A Smith predictor-based control scheme for motion synchronization in virtual environments under large time-delays is presented in [35].

2.4.1 Preliminaries

Starting from the classic scheme of a haptic system, presented in Figure 1.3, a Smith predictor was added; the corresponding block scheme is presented in Figure 2.13.

Here, this *asymmetric* control scheme is proposed since the main objective is to increase the end user perception in terms of transparency. More precisely, the entire virtual part including the

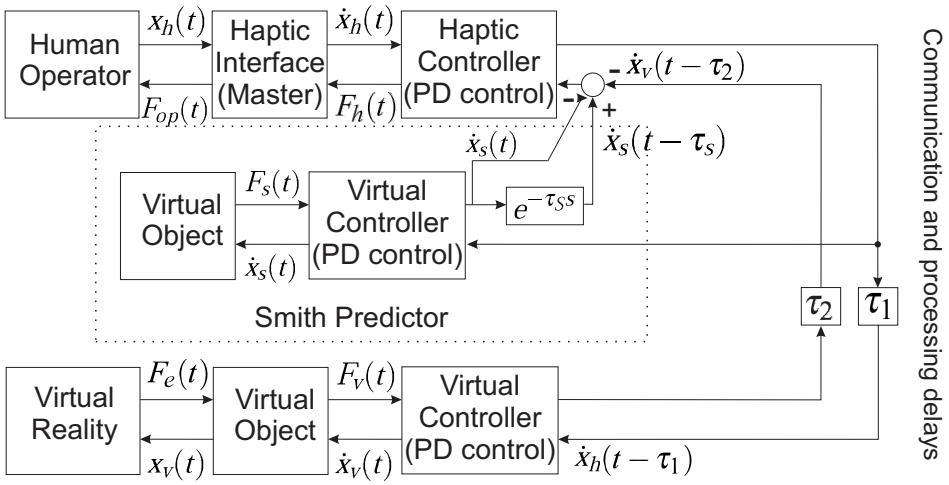


FIGURE 2.13: Block scheme for Haptic/Teleoperation Systems including Smith predictor (τ_S represents the time-delay used in the predictor).

controller is considered to be a "*compact*" plant affected by time delays. With this assumption, the goal of the Smith predictor is to compensate the round trip delay of the *virtual* plant (virtual side). It is worth mentioning that a second Smith predictor inserted in the virtual controller will introduce new uncertainties which will make the system more vulnerable to instability, without adding additional improvements in terms of human operator's perception. Figure 2.14 presents the control scheme of such a system.

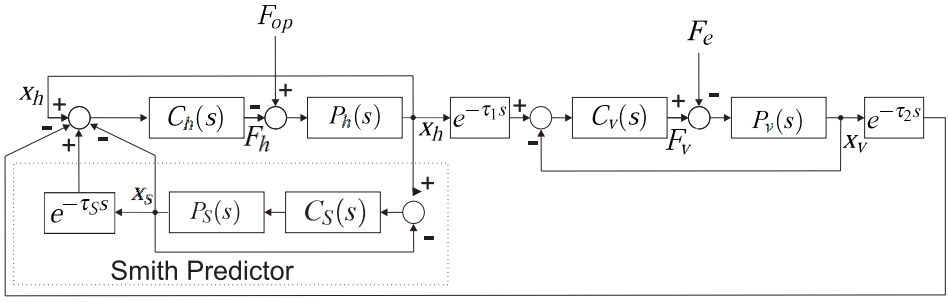


FIGURE 2.14: Control scheme of Smith predictor-based system.

Considering the system presented in Figure 2.14, the equations describing the response can be written as follows:

$$X_h(s) = P_h(s)\Big(F_{op}(s) - C_h(s)\big(X_h(s) - e^{-\tau_2 s}X_v(s) + (e^{-\tau_S s} - 1)X_S(s)\big)\Big), \quad (2.72)$$

$$X_v(s) = P_v(s)\left(-F_e(s) + C_v(s)\left(-X_v(s) + e^{-\tau_1 s}X_h(s)\right)\right), \quad (2.73)$$

$$X_S(s) = P_S(s)C_S(s)(X_h(s) - X_S(s)), \quad (2.74)$$

where τ_S represents the time-delay considered in the Smith predictor, $X_h(s)$, $X_v(s)$, $X_S(s)$ (position resulting from the Smith predictor), $F_{op}(s)$, $F_e(s)$, $P_h(s)$, $P_v(s)$, $P_S(s)$ (model used in the

Smith predictor) denote the Laplace transforms. The parameters used for the haptic interface, the virtual object and the Smith predictor are assumed to be equal. The transfer functions of $P_h(s)$, $P_v(s)$, $P_S(s)$, $C_h(s)$, $C_v(s)$, $C_S(s)$ are defined as follows:

$$P_h(s) = P_v(s) = P_S(s) =: P(s) = \frac{1}{s(ms+b)}, \quad C_h(s) = C_v(s) = C_S(s) =: C(s) = K_p + K_d s, \tag{2.75}$$

where m represents the mass and b is the damping coefficient.

From (2.72)-(2.74), the system's transfer function is obtained:

$$\frac{X_h}{F_{op}} = \frac{Q(s,\ K_p,\ K_d)}{P(s,\ K_p,\ K_d,\ \tau_1,\ \tau_2,\ \tau_S)}, \tag{2.76}$$

with:

$$P(s,\ K_p,\ K_d,\ \tau_1,\ \tau_2,\ \tau_S) = P_1(s,\ K_p,\ K_d) + P_2(s,\ K_p,\ K_d)(e^{-(\tau_1+\tau_2)s} - e^{-\tau_S s}). \tag{2.77}$$

Based on the case without predictor - (2.4), the controller's equation corresponding to the haptic side - $F_h(t)$, rewrites as follos:

$$F_h(t) = K_{d_h}(\dot{x}_h(t) - \dot{x}_v(t-\tau_2) + \hat{\dot{x}}_v(t-\tau_S)) - \hat{\dot{x}}_v(t) + K_{p_h}(x_h(t) - x_v(t-\tau_2) + \hat{x}_v(t-\tau_S) - \hat{x}_v(t)), \tag{2.78}$$

where $\hat{\dot{x}}_v$, $\hat{x}_v$ represent the estimated velocity and position for virtual object. The estimation is made by using a similar model to the one running on the virtual reality environment.

2.4.2 Stability analysis

In this subsection a stability analysis is proposed from two points of view:

- Stability regions in controller's gain parameters,
- Stability in delay parameters-space.

In the first case, the stability region will be represented in the controller parameters-space $(K_p,\ K_d)$, for fixed or distributed time-delays. The second case presents the stability regions in the delays-space (system delay $(\tau_1 + \tau_2)$ and the delay used in the Smith predictor τ_S) for a predefined tuning of the controller gains $(K_p,\ K_d)$. Resuming, the first case describes the stability limits in the controller gains-space for *known*, delays, while the second case presents the delays dependency in terms of stability for fixed PD gains.

2.4.2.1 Stability regions

Without any loss of generality, the following assumptions are made:

Assumption 3. The polynomials P and Q are such that $\deg(Q) \leq \deg(P)$.

Assumption 4. The polynomial P does not have any roots at the origin, that is $P(0) \neq 0$.

Assumption 5. The polynomials P and Q do not have common roots.

Assumption 6. The polynomials P and Q satisfy the following condition:

$$\lim_{s\to\infty} \left| \frac{Q(s;K_p,K_d)}{P(s,\ K_p,\ K_d,\ \tau_1,\ \tau_2,\ \tau_S)} \right| < \frac{1}{2}.$$

If one of the Assumptions 3-4 is violated, then the system cannot be stabilized for any positive delay. Assumption 5 imposes to have the system in its *simplest* form. Finally, if the system is of retarded type (imposed by Assumption 3), Assumption 6 is satisfied automatically. The neutral type systems will not be discussed here, since this case is not applicable in haptics. For discussions on the implications of these assumptions the readers are referred to [58, 107, 124].

According to the continuity of zeros with respect to the system's parameters (see, for instance, [48] for the continuity with respect to delays), the number of roots in the right half plane (RHP) can change only when some zeros of the closed-loop system appear and cross the imaginary axis. Therefore, a useful concept is the frequency crossing set Ω defined as the set of all real positive ω for which there exist at least a pair (K_p, K_d) such that:

$$P(j\omega;\ K_p,\ K_d,\ \tau_1,\ \tau_2,\ \tau_S) = 0. \tag{2.79}$$

Only positive frequencies ω need to be considered, that is $\Omega \subset (0,\infty)$, since obviously:

$$P(j\omega;\ K_p,\ K_d,\ \tau_1,\ \tau_2,\ \tau_S) = 0 \Leftrightarrow \overline{P(-j\omega;\ K_p,\ K_d,\ \tau_1,\ \tau_2,\ \tau_S} = 0.$$

Proposition 2.18. *For a given $\tau_0 \in \mathbb{R}_+$ and $\omega \in \Omega \subset \mathbb{R}_+^*$ a corresponding crossing point (K_p, K_d) is given by the solutions of the following system:*

$$\begin{cases} \Re\left(P(j\omega;\ K_p,\ K_d,\ \tau_1,\ \tau_2,\ \tau_S)/_{s=j\omega}\right) &= 0, \\ \Im\left(P(j\omega;\ K_p,\ K_d,\ \tau_1,\ \tau_2,\ \tau_S)/_{s=j\omega}\right) &= 0, \end{cases} \tag{2.80}$$

Remark 2.19. It is easy to see that $\forall\ \omega \in \Omega$ the following condition holds: $P(j\omega) \neq 0$. Otherwise, $P(j\omega) = 0$, that contradicts Assumption 3.

Let $\Omega_{K_p^*,K_d^*}$ denotes the set of all frequencies $\omega > 0$ satisfying (2.80) for at least one pair of $(K_p,\ K_d)$ in the rectangle $|\ K_p\ | \leq \bar{K_p}, |\ K_d\ | \leq \bar{K_d}$. Then, when ω varies within some interval

Ω_l, satisfying (2.80) defines a continuous curve. Denote $\mathcal{T}_l$ the curve corresponding to Ω_l, $\forall l \in 1,\ldots,\mathcal{N}$ and consider the following decompositions:

$$R_0 + jI_0 = j\frac{\partial P(s;\ K_p,\ K_d,\ \tau_1,\ \tau_2,\ \tau_S)}{\partial s}\Big|_{s=j\omega}, \tag{2.81}$$

$$R_1 + jI_1 = -\frac{\partial P(s;\ K_p,\ K_d,\ \tau_1,\ \tau_2,\ \tau_S)}{\partial K_d}\Big|_{s=j\omega}, \tag{2.82}$$

$$R_2 + jI_2 = -\frac{\partial P(s;\ K_p,\ K_d,\ \tau_1,\ \tau_2,\ \tau_S}{\partial K_p}\Big|_{s=j\omega}. \tag{2.83}$$

Similar to the constant time-delay case, the tangent of $\mathcal{T}_l$ can be expressed as follows:

$$\begin{pmatrix} \frac{dK_p}{d\omega} \\ \frac{dK_d}{d\omega} \end{pmatrix} = \begin{pmatrix} R_2 & R_1 \\ I_2 & I_1 \end{pmatrix}^{-1} \begin{pmatrix} R_0 \\ I_0 \end{pmatrix} = \frac{1}{R_2I_1 - R_1I_2} \begin{pmatrix} R_0I_1 - R_1I_0 \\ R_0I_2 - R_2I_0 \end{pmatrix}, \tag{2.84}$$

provided that: $R_1I_2 - R_2I_1 \neq 0$.

In order to derive the stability region of the system given by (2.76), [123] characterized the smoothness of the crossing curves and the corresponding direction of crossing.

Proposition 2.20. *The curve $\mathcal{T}_l$ is smooth everywhere except if the point corresponding to $s = j\omega$ is a multiple solution of (2.76).*

Remark 2.21. The crossing direction of the curve that corresponds to increasing ω will be called the *positive direction*. Also, the region on the left hand side as one heads in the positive direction of the curve will be called *the region on the left*. A complete presentation can be found in subsection 2.3.2.2.

2.4.3 Stability in delay parameters-space

Consider now that the controller gains are fixed $K_p = K_p^*$, $K_d = K_d^*$, and discuss the influence of delay parameters on the stability of the system.

Proposition 2.22. *[124] The crossing set Ω consists of a finite number of intervals of finite length and it is determined by solving:*

$$\left|\frac{Q(j\omega;K_p^*,K_d^*)}{P(j\omega;\ K_p^*,\ K_d^*,\ \tau_1,\ \tau_2,\ \tau_S)}\right| \geq \frac{1}{2}. \tag{2.85}$$

In what comes next, the following notations will be used:

$$h(j\omega) = \frac{P(j\omega,\ K_p^*,\ K_d^*,\ \tau_1,\ \tau_2,\ \tau_S)}{Q(j\omega,K_p^*,K_d^*)}.$$

According to [124], for a given $\omega \in \Omega$, the set $\mathcal{T}_\omega$ consisting of all the pairs $((\tau_1+\tau_2),\tau_S)$ can be found, satisfying $H(j\omega,\ K_p^*,\ K_d^*,\ \tau_1,\ \tau_2,\ \tau_S)=0$ as follows:

$$(\tau_1+\tau_2)=(\tau_1+\tau_2)^{u\pm}(\omega)=\frac{\angle h(j\omega)+(2u-1)\pi\pm q}{\omega},\quad u=u_0^{\pm},u_0^{\pm}+1,u_0^{\pm}+2,\ldots\ , \tag{2.86}$$

$$\tau_S=\tau_S^{v\pm}(\omega)\qquad =\frac{\angle h(j\omega)+2v\pi\mp q}{\omega},\qquad v=v_0^{\pm}(u),v_0^{\pm}(u)+1,v_0^{\pm}(u)+2,\ldots\ , \tag{2.87}$$

where $q\in[0,\pi]$ is given by:

$$q(j\omega)=\cos^{-1}\left(\frac{1}{2|h(\omega)|}\right) \tag{2.88}$$

and u_0^+,u_0^- are the smallest integers (may be dependent on ω) such that the corresponding values $(\tau_1+\tau_2)^{u_0^+ +}$, $(\tau_1+\tau_2)^{u_0^- -}$ are nonnegative, and v_0^+ and v_0^- are integers dependent on u. The position in Figure 2.15 corresponds to $((\tau_1+\tau_2)^{u+},\tau_S^{v+})$ and the mirror image about the real axis corresponds to $((\tau_1+\tau_2)^{u-},\tau_S^{v-})$.

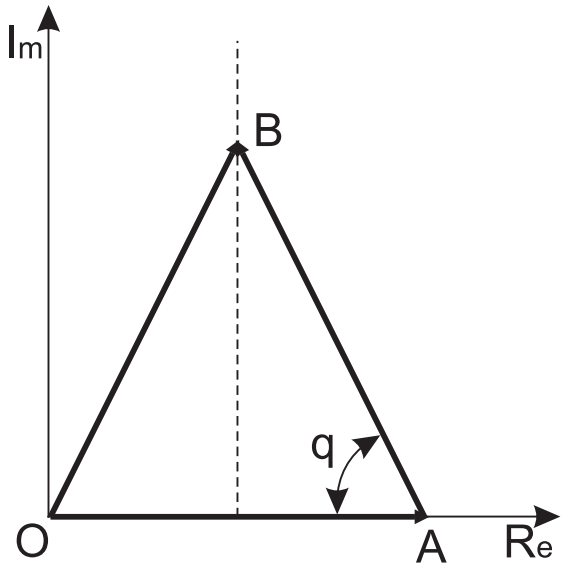


FIGURE 2.15: Triangle formed by 1, $h(s)e^{-(\tau_1+\tau_2)s}$ and $h(s)e^{-\tau_S s}$.

If $\mathcal{T}^+_{\omega,u,v}$ and $\mathcal{T}^-_{\omega,u,v}$ are defined as the singletons $((\tau_1+\tau_2)^{u+}(\omega),\tau_S^{v+}(\omega))$ and $((\tau_1+\tau_2)^{u-}(\omega),\tau_S^{v-}(\omega))$ respectively, then $\mathcal{T}_\omega$ can be characterized as follows:

$$\mathcal{T}_\omega=\left(\bigcup_{u\geq u_0^+,v\geq v_0^+}\mathcal{T}^+_{\omega,u,v}\right)\bigcup\left(\bigcup_{u\geq u_0^-,v\geq v_0^-}\mathcal{T}^-_{\omega,u,v}\right)$$

The set of stability crossing curves in delay parameter space is defined by:

$$\mathcal{T}=\bigcup_{k=1}^{N}\mathcal{T}^k,\quad \mathcal{T}^k=\bigcup_{\omega\in\Omega_k}\mathcal{T}_\omega \tag{2.89}$$

Remark 2.23. The distance between $((\tau_1+\tau_2),\ \tau_S)$ and $\mathcal{T}$ is a measure of fragility of the controller (K_p^*, K_d^*) with respect to delay uncertainty.

In order to better illustrate this approach, a short example from haptics is considered below. Considering $m = 1\ Kg$, $b = 0.1\ Ns/m$, for the chosen PD controller's gains $K_p = 1500\ N/m$ and $K_d = 80\ Ns/m$, Figure 2.16 presents the crossing curves of the system for this case. For each pair of $u,\ v$ fixed, when ω varies in some interval, based on equations (2.86)-(2.87), a crossing curve is defined as depicted in Figure 2.16. The stability region is determined knowing that the system is stable when the two delays are equal, and further on the stability frontiers may be determined on the graphic.

This result gives a general overview of the system limitations in terms of delay variations. The stability crossing curves are delimiting regions having the same number of unstable roots. More details can be found in [106, 124].

As a general rule, the delay used in the Smith predictor is considered to be fixed. Once the complete configuration is known, it is easy to see what are the minimum and maximum limits of the time-delays' uncertainties from the stability point of view.

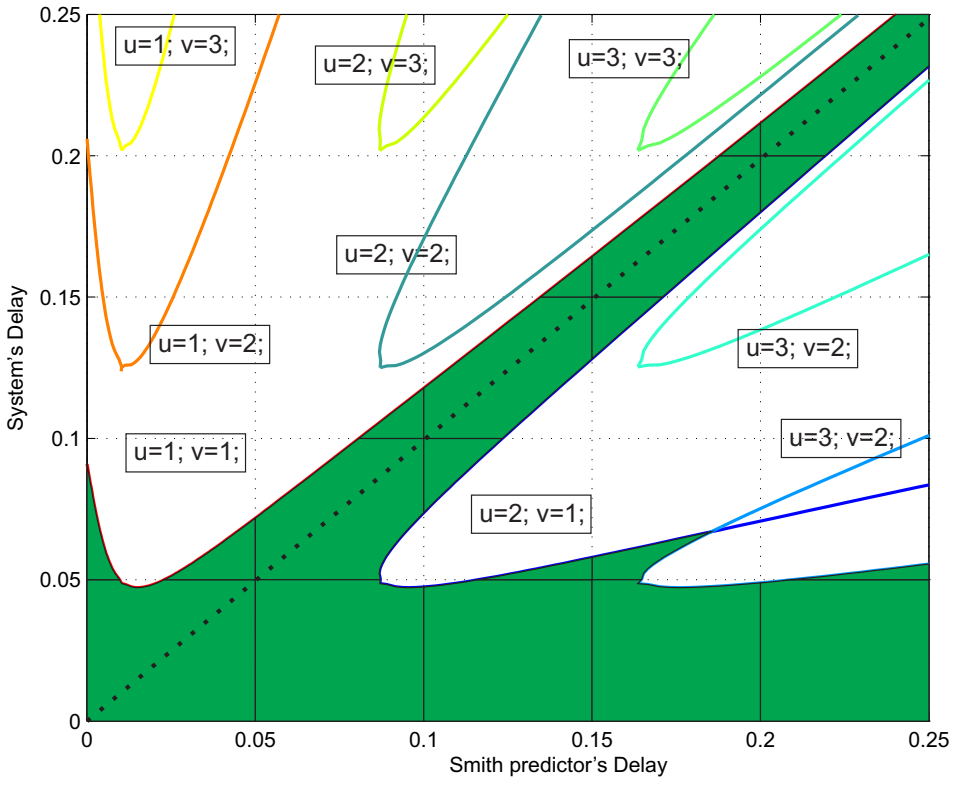


FIGURE 2.16: Example of Crossing curves for $K_p = 1500\ N/m$, $K_d = 80\ Ns/m$. Here, the stability zone is represented by the green zone.

2.4.4 Fragility of Smith predictors

The main goal of this subsection is to provide a method for deriving the biggest positive value d such that for a stabilizing controller with the Smith predictor built-in $(K_p^*,\ K_d^*)$, the system is also stabilized by any pair $K_p,\ K_d$ as long as:

$$\sqrt{(K_p - K_p^*)^2 + (K_d - K_d^*)^2} < d. \tag{2.90}$$

This problem can be more generally reformulated as: *finding the maximum controller gains deviation d such that the number of unstable roots of (2.82) remains unchanged.*

First, let's introduce some notation:

$$\mathcal{T} = \bigcup_{\ell=1}^{N} \mathcal{T}_\ell, \qquad \mathcal{T}_\ell = \{(K_p, K_d) | \omega \in \Omega_\ell\}, \tag{2.91}$$

$$\tilde{k}(\omega) = (K_p(\omega), K_d(\omega))^T, \qquad \tilde{k}^* = (K_p^*, K_d^*)^T, \tag{2.92}$$

where K_p^*, K_d^* are fixed.

Let's also denote $\mathrm{d}_{\mathcal{T}} = \min\limits_{\ell \in \{1,\dots,N\}} d_\ell$, where:

$$d_\ell = \min\left\{ \sqrt{(K_p - K_p^*)^2 + (K_d - K_d^*)^2} \ \ / \ \ (K_p, K_d) \in \mathcal{T}_\ell \right\}. \tag{2.93}$$

With the notation and the results above, the following proposition gives an estimation for the maximum controller's parameter deviation (i.e. non-fragile controllers characterization):

Proposition 2.24. *The maximum parameter deviation from (K_p^*, K_d^*), without changing the number of unstable roots of the closed-loop equation (2.79) can be expressed as:*

$$d = min\left\{ K_{d\infty}, |K_p^* - K_p(0)|, \min_{\omega \in \Omega_f} \left\{ \left\| \tilde{k}(\omega) - \tilde{k}^* \right\|_2 \right\} \right\}, \tag{2.94}$$

where:

$$K_{d\infty} := \begin{cases} \min\left\{ \left| K_d^* - \left| \frac{q_n}{p_m} \right| \right|, \left| K_d^* + \left| \frac{q_n}{p_m} \right| \right| \right\}, & \text{if} \quad m = n-1, \\ 0, & \text{if} \quad m < n-1, \end{cases}$$

where q_m, p_n represent the coefficient of the highest order of the polynomials P, Q and Ω_f is the set of roots of the function $f : \mathbb{R}_+ \mapsto \mathbb{R}$:

$$f(\omega) \triangleq \left\langle \left(\tilde{k}(\omega) - \tilde{k}^* \right), \frac{d\tilde{k}(\omega)}{d\omega} \right\rangle \tag{2.95}$$

As it was already presented in subsections B.2 and 2.3.2.3, similarly, the explicit computation of the maximum parameter deviation d can be summarized by the following algorithm:

Step 1: First, compute the *"degenerate"* points of each curve $\mathcal{T}_1$ (i.e. the roots of $R_1 I_2 - R_2 I_1 = 0$ and the multiple solutions of (2.76)).

Step 2: Second, compute the set Ω_f defined by *Proposition* 5 (i.e. the roots of equation $f(\omega) = 0$, where f is given by (2.95)).

Step 3: Finally, the corresponding maximum parameter deviation d_l is defined by (2.93).

Remark 2.25. *(On the gains' optimization):* It is worth mentioning that the geometric argument above can be easily used for solving other robustness problems. Thus, for instance, if one of the controller's parameters is fixed (prescribed), also the maximum interval guaranteeing closed-loop stability with respect to the other parameter can be explicitly computed. In particular if K_d ("derivative") is fixed, the corresponding stabilizing maximum gain interval can be derived.

2.4.5 Fixed and known delays

In this case, the use of a Smith predictor projects the problem in the non-delayed case. For $\tau_S = \tau_1 + \tau_2$, $P(s,\ K_p,\ K_d,\ \tau_1,\ \tau_2,\ \tau_S)$ (equation (2.77)) becomes $P(s,\ K_p,\ K_d) = P_1(s,\ K_p,\ K_d)$.

Using the method presented in Subsection 2.4.2.1, the stability zone in $(K_p,\ K_d)$ parameters is drawn in Figure 2.17, considering a sampling time delay of 0.001 sec (with $m = 1\ Kg,\ b = 0.1\ Nsec/m$).

According to this result and also to the practical experiments, in this case, the values of K_p and K_d are basically *free* to be chosen from the positive domain ($\mathbb{R}_+$).

2.4.6 Uncertain Delays

2.4.6.1 Uncertain delays: constant uncertainty

In this case, the overall closed-loop transfer function of the system described in Figure 2.14 becomes:

$$H_{x_h/F_{op}}(s;\ K_p,\ K_d,\ \tau_1,\ \tau_2,\ \tau_S,\ \Delta) = \frac{Q(s;\ K_p,\ K_d)}{P(s;\ K_p,\ K_d,\ \tau_1,\ \tau_2,\ \tau_S,\ \Delta)}. \tag{2.96}$$

where Δ represents the delay uncertainty. Considering τ_0 the nominal delay, the system's delays are defined to be $\tau_1 = \tau_2 = \tau_0 + \Delta$, and the Smith predictor's delay $\tau_S = 2\tau_0$ (because the round

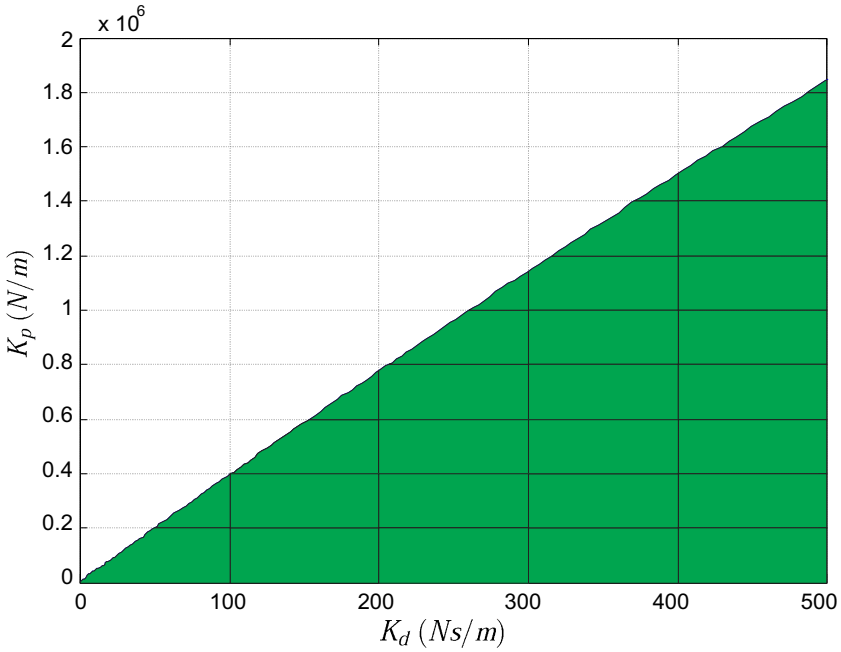


FIGURE 2.17: Stability region (in green) for (K_p, K_d) for a time delay of 0.001 sec.

trip delay $\tau_1 + \tau_2$ has to be compensated). With these considerations, the characteristic equation from (2.96) rewrites as follows:

$$P(s;K_p,K_d,\tau_0,\Delta) = P_1(s;K_p,K_d) \ + \ P_2(s;K_p,K_d)(e^{-2\tau_0 s} - e^{-2(\tau_0+\Delta)s}). \tag{2.97}$$

Using the method presented in Subsection 2.4.2.1, the stability region in $(K_p,\ K_d)$ parameters, corresponding to equation (2.97) is drawn in Figure 2.18, for a nominal delay $\tau_0 = 0.05\ sec$ and a fix delay uncertainty $\Delta = 0.01\ sec$ (with $m = 1\ Kg,\ b = 0.1\ Nsec/m$).

Distributed delays

In the sequel, different probability distributions are used to describe the time-delay pattern. Several *types* of time-delays, covering the most common situations, will be presented and analyzed from the stability point of view. Based on the controller equations (1.3)-(1.4):

$$F_h(t) = \underbrace{K_{d_h}(\dot{x}_h(t) - \dot{x}_v(t-\tau_2))}_{\text{delayed D-action}} \underbrace{+K_{p_h}(x_h(t) - x_v(t-\tau_2))}_{\text{delayed P-action}}, \tag{2.98}$$

$$F_v(t) = \underbrace{K_{d_v}(\dot{x}_h(t-\tau_1) - \dot{x}_v(t))}_{\text{delayed D-action}} \underbrace{+K_{p_v}(x_h(t-\tau_1) - x_v(t))}_{\text{delayed P-action}}, \tag{2.99}$$

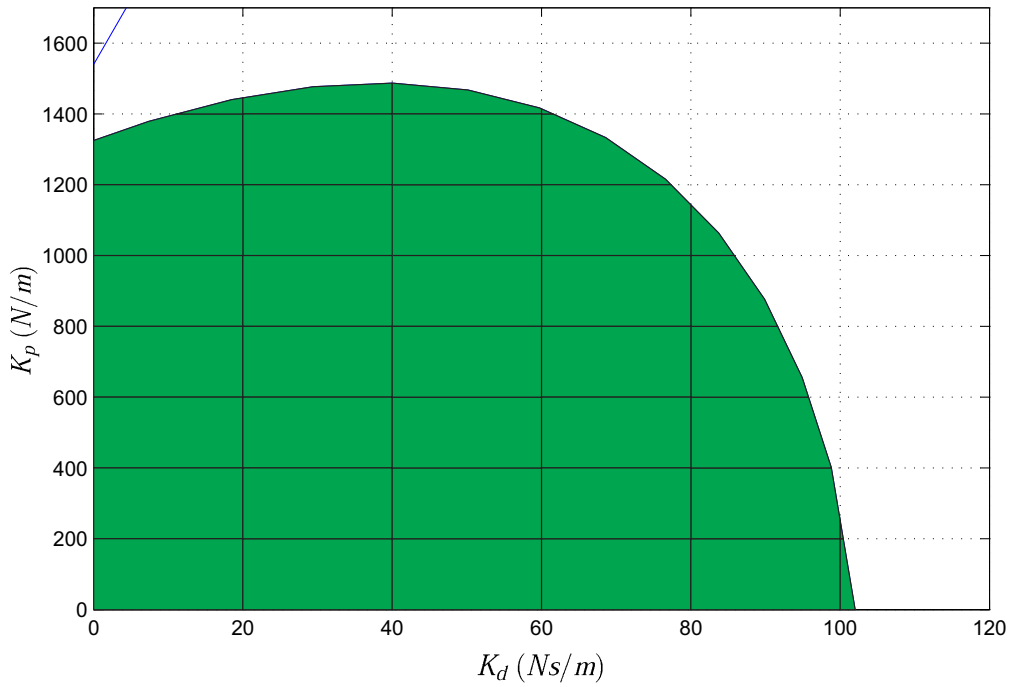


FIGURE 2.18: Stability region (in green) for (K_p, K_d) for a fix uncertainty $\Delta = \Delta_1 = \Delta_2 = 0.01\ sec$.

corresponding to the case with *simple* time-delays, for the distribution representation, the equations (2.98)-(2.99) become:

$$F_h(t) = \underbrace{K_d(\int_{\mathbb{R}} f(\theta)(\dot{x}_h(t) - \dot{x}_v(t-\tau_0-\theta))d\theta}_{\text{delayed D-action}} + \underbrace{K_p \int_{\mathbb{R}} f(\theta)(x_h(t) - x_v(t-\tau_0-\theta))d\theta}_{\text{delayed P-action}}, \quad (2.100)$$

$$F_v(t) = \underbrace{K_d \int_{\mathbb{R}} f(\theta)(\dot{x}_v(t) - \dot{x}_h(t-\tau_0-\theta))d\theta}_{\text{delayed D-action}} + \underbrace{K_p \int_{\mathbb{R}} f(\theta)(x_v(t) - x_h(t-\tau_0-\theta))d\theta}_{\text{delayed P-action}}, \quad (2.101)$$

where f represents the distribution kernel. Considering F the Laplace transformation of the distribution kernel, the equations (2.72)-(2.73), describing the system response, rewrites as follows:

$$\begin{aligned} X_h(s) &= H(s)\Big(F_{op}(s) - C_{PD}(s)\big(X_h(s) - F(s,\ \tau_2)X_v(s) + (e^{-\tau_S s} - 1)X_S(s)\big)\Big), \quad (2.102) \\ X_v(s) &= V(s)\left(-F_e(s) + C_{PD}(s)\left(-X_v(s) + F(s,\ \tau_1)X_h(s)\right)\right). \quad (2.103) \end{aligned}$$

With these considerations the general form of $P(s,\ K_p,\ K_d,\ \tau_1,\ \tau_2,\ \tau_S)$, rewrites as follows:

$$P(s,\ K_p,\ K_d,\ \tau_1,\ \tau_2,\ \tau_S) = P_1(s, K_p, K_d)\left(F(s,\ \tau_2)F(s,\ \tau_1) - e^{-\tau_S s}\right) + P_2(s, K_p, K_d) \quad (2.104)$$

2.4.6.2 Uncertain delays: uniform distribution

As previously mentioned in Subsection 2.3.3.1, the uniform distribution represents an appropriate approach when dealing with variable time-delays for which only the minimal and maximum values are known, with no additional information about the variation behavior. The distribution of an uniform time-delay was already presented in Figure 2.9 from Subsection 2.3.3.1.

Considering $\Delta_1, \Delta_2 > 0$ and $\tau_0 \geq \Delta_1$, the uniform distribution kernel is given by:

$$f(\xi) = \begin{cases} \frac{1}{\Delta_1+\Delta_2}, & \text{if } \tau_0 - \Delta_1 < \xi < \tau_0 + \Delta_2, \\ 0, & \text{otherwise,} \end{cases} \tag{2.105}$$

where τ_0 is the reference delay and Δ_1, Δ_2 represent the delay uncertainty in both directions.

The Laplace transform of the uniform distribution (already presented in Subsection 2.3.3.1) is given by:

$$F(s) = \frac{e^{-s(\tau_0-\Delta_1)} - e^{-s(\tau_0+\Delta_2)})}{s(\Delta_1+\Delta_2)}. \tag{2.106}$$

In the Smith predictor, the delay to be considered is $\tau_S = 2\tau_0$ because it must compensate the round trip delay $\tau_1 + \tau_2$, where $\tau_1, \tau_2 \in (\tau_0 - \Delta_1,\ \tau_0 + \Delta_2)$.

With these considerations, equations (2.102)-(2.103) rewrite as follows:

$$\begin{aligned} X_h(s) &= H(s)\left(F_{op}(s) - C_{PD}(s)\left(X_h(s) - \frac{e^{-s(\tau_0-\Delta_1)} - e^{-s(\tau_0+\Delta_2)}}{s(\Delta_1+\Delta_2)} X_v(s) + \right.\right. \\ &\quad \left.\left. + (e^{-\tau_S s} - 1)X_S(s)\right)\right), \end{aligned} \tag{2.107}$$

$$X_v(s) = V(s)\left(-F_e(s) + C_{PD}(s)\left(-X_v(s) + \frac{e^{-s(\tau_0-\Delta_1)} - e^{-s(\tau_0+\Delta_2)}}{s(\Delta_1+\Delta_2)} X_h(s)\right)\right), \tag{2.108}$$

and equation (2.104) becomes:

$$P(s,\ K_p,\ K_d,\ \tau_1,\ \tau_2,\ \tau_S) = P_1(s, K_p, K_d)\left(\left(\frac{e^{-s(\tau_0-\Delta_1)} - e^{-s(\tau_0+\Delta_2)})}{s(\Delta_1+\Delta_2)}\right)^2 - e^{-\tau_S s}\right) + P_2(s, K_p, K_d).$$

Using the method presented in Subsection 2.4.2.1, the stability zone in $(K_p,\ K_d)$ parameters is drawn in Figure 2.19 for the delay uncertainties $\Delta_1 = \Delta_2 = 0.015$ *sec*.

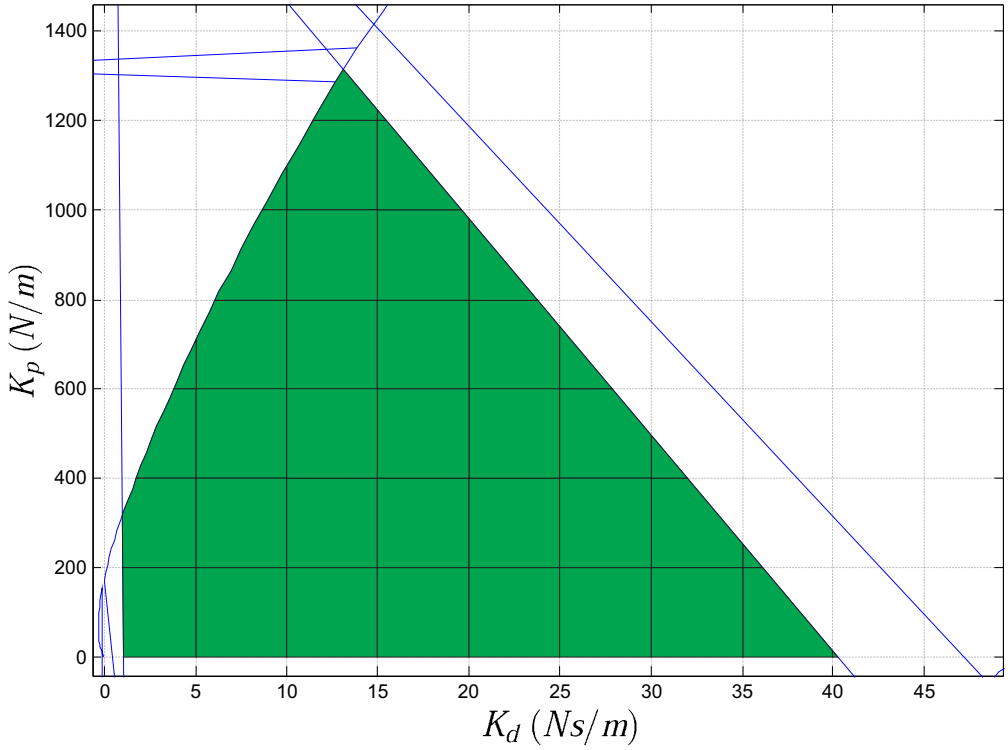


FIGURE 2.19: Stability region (in green) for (K_p, K_d) for $\Delta_1 = \Delta_2 = 0.015$ *sec*.

2.4.6.3 Uncertain delays: gamma distribution with gap

In the case of gamma-distribution with a gap, the kernel f is given by:

$$f(\xi) = \begin{cases} 0, & \xi < \tau, \\ \frac{(\xi-\tau)^{n-1} e^{-\frac{\xi-\tau}{T}}}{T^n (n-1)!}, & \xi \geq \tau, \end{cases} \tag{2.109}$$

where $n \in \mathbb{N}$, $T > 0$ and $\tau \geq 0$. Note that $f(\xi) \geq 0$ for all $\xi \geq 0$ and $\int_0^\infty f(\xi) d\xi = 1$. The gap is defined by τ, and the corresponding *average delay* of (2.109) satisfies:

$$\tau_m = \int_0^\infty \xi\, f(\xi)\, d\xi = \tau + nT. \tag{2.110}$$

An example of the gamma distribution with gap was already presented in Figure 2.10 from Subsection 2.3.3.2.

Using the Laplace transform of the gamma with gap distribution and the delays' average defined by equations (2.67) and (2.68) respectively, in the case of gamma distribution with gap,

equations (2.102)-(2.103) rewrite as follows:

$$X_h(s) = H(s)\left(F_{op}(s) - C_{PD}(s)\left(X_h(s) - \frac{e^{-\hat{\tau}_2 s}}{(1+sT_2)^{n_2}}X_v(s) + \right.\right. \tag{2.111}$$
$$\left.\left. + (e^{-\tau_S s} - 1)X_S(s)\right)\right),$$
$$X_v(s) = V(s)\left(-F_e(s) + C_{PD}(s)\left(-X_v(s) + \frac{e^{-\hat{\tau}_1 s}}{(1+sT_1)^{n_2}}X_h(s)\right)\right), \tag{2.112}$$

Considering $\tau_s = \tau_1 + \tau_2$, with $\tau_1 = \tau_2 = \tau + nT$ ($\hat{\tau}_1 = \hat{\tau}_2 = \tau,\ n_1 = n_2 = n,\ T_1 = T_2 = T$), equation (2.104) rewrites as follows:

$$P(s,\ K_p,\ K_d,\ \tau,\ T,\ n,\ \tau_S) = P_1(s,K_p,K_d)\left(\left(\frac{e^{-\tau s}}{(1+sT)^n}\right)^2 - e^{-\tau_S s}\right) + P_2(s,K_p,K_d)$$

Using the same stability method proposed in Subsection 2.4.2.1, Figure 2.20 presents the stability zone in K_p and K_d parameters space, considering $\tau = 0.04\ sec$, $T = 0.01\ sec$ and $n = 1$.

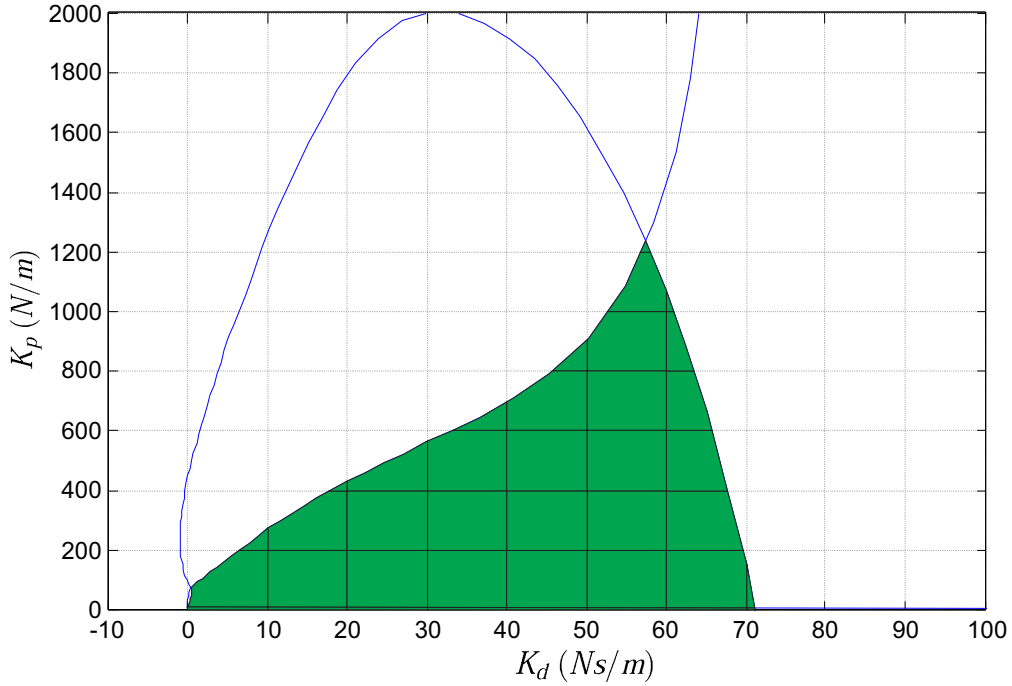


FIGURE 2.20: Stability region (in green) for (K_p, K_d) for a gamma distributed delay with gap ($\tau = 0.04\ sec$, $T = 0.01\ sec$, $n = 1$).

2.4.6.4 Uncertain delays: normal distribution

For this case, like for the previous ones, a fixed time-delay is considered for the Smith predictor and a variable delay on the system's side. The system's time-delays have the following form:

$$\tau_1 = \tau_2 = \tau_0 + \Delta, \tag{2.113}$$

where τ_0 is the nominal delay. Figure 2.21 presents an example of time-delay shape for a normal distribution.

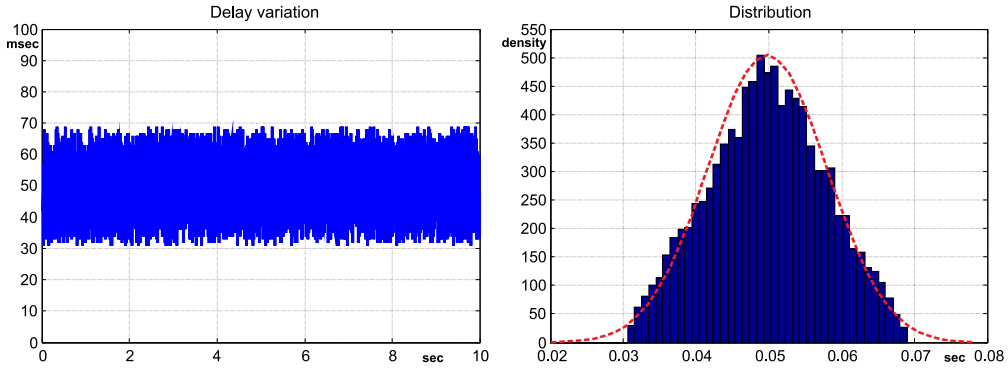


FIGURE 2.21: Example of normal distributed delay: shape and distribution (mean value $\tau_1 = \tau_2 = 0.05$ *sec*.

In the Smith predictor, the time-delay will be considered $\tau_S = 2\tau_0$ because the round trip delay $\tau_1 + \tau_2$ has to be compensated. The uncertainty Δ is considered to have a stochastic variation defined by a probability kernel f given by:

$$f(x) = \frac{1}{\sqrt{2\pi\delta^2}} e^{-\frac{x^2}{2\delta^2}}, \tag{2.114}$$

where δ^2 defines the standard deviation. The Laplace transform of the normal distribution (see [190] for further details) is given by:

$$F(s) = \int_0^{\infty} e^{-st} s^{-t^2} dt = \frac{\sqrt{\pi}}{2} e^{\frac{s^2}{4}} erf\left(\frac{s}{2}\right), \tag{2.115}$$

where $erf(\cdot)$ is the complementary error function, defined as:

$$erf(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-u^2} du = \frac{2}{\sqrt{\pi}} \left(x - \frac{x^3}{3.1!} + \frac{x^5}{5.2!} - \frac{x^7}{7.3!} + \ldots \right). \tag{2.116}$$

Let us consider $E(X)$ the mean of a random variable X and Δ modeled as a Gaussian white noise. The mean value of Δ is zero. Then,

$$F_h(t) = \underbrace{K_d \Big(E\left(\dot{x}_h(t)\right) - E\left(\dot{x}_v(t-\tau_2)\right) \Big)}_{\text{delayed D-action}} \underbrace{+K_p \Big(E\left(x_h(t)\right) - E\left(x_v(t-\tau_2)\right) \Big)}_{\text{delayed P-action}}, \tag{2.117}$$

$$F_v(t) = \underbrace{K_d \Big(E\left(\dot{x}_v(t)\right) - E\left(\dot{x}_h(t-\tau_1)\right) \Big)}_{\text{delayed D-action}} \underbrace{+K_p \Big(E\left(x_v(t)\right) - E\left(x_h(t-\tau_1)\right) \Big)}_{\text{delayed P-action}}, \tag{2.118}$$

Furthermore, using the continuity of $\dot{x}_h(t), \dot{x}_v(t)$ on equations (2.117)-(2.118) rewrite as:

$$F_h(t) = \underbrace{K_d\Big(\frac{d}{dt}E\left(x_h(t)\right) - \frac{d}{dt}E\left(x_v(t-\tau_2)\right)\Big)}_{\text{delayed D-action}} \underbrace{+K_p\Big(E\left(x_h(t)\right) - E\left(x_v(t-\tau_2)\right)\Big)}_{\text{delayed P-action}}, \tag{2.119}$$

$$F_v(t) = \underbrace{K_d\Big(\frac{d}{dt}E\left(x_v(t)\right) - \frac{d}{dt}E\left(x_h(t-\tau_1)\right)\Big)}_{\text{delayed D-action}} \underbrace{+K_p\Big(E\left(x_v(t)\right) - E\left(x_h(t-\tau_1)\right)\Big)}_{\text{delayed P-action}}, \tag{2.120}$$

which means that the analysis made in the known delay case still holds but for the mean values of $x_h(t)$ and $x_v(t)$.

Therefore, the choice $(K_p,\, K_d)$ ensuring the stability in the known time-delay case (Figure 2.17) will assure the mean-stability in the uncertain delay case [71, 184]. In other words, the Smith predictor is able to compensate the delays on any given time interval even if for point-wise values of Δ the system is unstable.

In the case of uncertain time-delays, a special attention is required for the *behavior* of the delay. Depending on this, the time-delay can be modeled in order to choose more efficiently the controller's parameters providing high performance in terms of end user perception and guarantying the system's stability. As it was previously presented, depending on the distribution type, the choice of controller's gains can be more or less restrictive. More precisely, for the uniform distribution, the choice of parameters is the most restrictive one, while for the normal distribution the choice is close to the *ideal* case. The uniform distribution corresponds to the case when little information about the delay variation is known (i.e. only the minimum and maximum values are needed in this case). This is why the uniform distribution case is the most restrictive one in terms of parameters' choice. On the other hand if the delay shape can be approximated by the gamma with gap or normal distribution the parameters' choice will be *bigger*. Concluding, the uniform distribution gives the most restrictive stability zone, based on a minimal information and the other two (gamma with gap and normal distribution) give a less restrictive stability zone, but based on more detailed information. Therefore, with limited information about the delay shape, tuning related to the uniform distribution must be considered.

As it was previously presented, depending on the distribution type, the choice of controller's gains can be more or less restrictive. More precisely, for the uniform distribution, the choice of parameters is the most restrictive one, while for the normal distribution the choice is close to the *ideal* case. The uniform distribution corresponds to the case when little information about the delay variation is known (i.e. only the minimum and maximum values are needed in this case). This is why the uniform distribution case is the most restrictive one in terms of parameters' choice. On the other hand if the delay shape can be approximated by the gamma with gap or normal distribution the parameters' choice will be *bigger*. Concluding, the uniform distribution gives the most restrictive stability zone, based on a minimal information and the

other two (gamma with gap and normal distribution) give a less restrictive stability zone, but based on more detailed information. Therefore, with limited information about the delay shape, tuning related to the uniform distribution must be considered.

2.5 Tuning and Physical Limitations

This section gives an overview of some physical limitation clues and also several remarks regarding the tuning of haptic and teleoperation systems.

As it was previously mentioned in haptics, as in teleoperation, there exist two functioning cases (situations):

- free motion - when there are no contacts,
- restricted motion - when the virtual object (slave robot) has contacts with the environment, and this phenomenon must be accurately transmitted to the end user via the haptic interface (master robot).

Between the two cases (free and restricted motions), there exist always the transition phases, which must be also taken into account due to the possibility of inducing unexpected situations or behaviors. To the best of the author's knowledge, the most *critical* situation may occur during the free to restricted transition which sometime can result in long time oscillations or even unstable behavior.

The first objective of the tuning procedure is to guarantee the stability of the system in all situations and functioning cases. Furthermore, under the assumption that the user is no longer moving the haptic interface (the velocity is zero and the position is constant), for achieving an asymptotically convergence to zero of the virtual object velocity and ensuring position coordination (between the haptic interface and virtual object or master/slave), the system must be stable.

The second objective is to guarantee the tracking error between the haptic interface and the virtual object (master and slave). Both functioning cases (free and restricted motion) must be carefully taken into account and properly analyzed. More precisely, in free motion, the desired behavior is to obtain a *perfect* tracking position and velocity between the haptic interface and the virtual object (or master/slave in the teleoperation framework). Since in most of the cases, the systems are affected by time-delays, it is worth mentioning that frequent direction changes at high velocities may induce important tracking errors. In this sense, the time response of the used actuators (generally DC motors) is limited, or/and other components of the system (except the communication network) may have slow responses. On the other hand, in restricted motion,

the tracking error should be maintained as small as possible. It is important to point out the idea that the feedback force is calculated based on this error, i.e. the force is proportional with the tracking error. Under this circumstance it is impossible to have zero tracking error, even in cases when there are no communication time-delays (ideal cases).

The third objective, and the most restrictive one among the majority of the cases, is the *transparency* constraint. As mentioned previously, transparency can be defined as the *telepresence sense* between the operator and the environment, [76]. For both systems - haptic and teleoperation, the transparency characterizes the end user sensation of acting directly with the virtual reality or the remote environment without any additional devices. Based on the two functioning cases - free and restricted motion, the transparency level must be achieved for each case under different criteria.

Ideally, in free motion, the force feedback felt by the human operator should be zero. In restricted motion, in order to establish the level of transparency, the *contact test* of the virtual object (or slave robot) with other objects is the most relevant test. Depending on the speed at the impact moment, the human operator should be able to feel the corresponding impact as he was directly having contact with the respective object. Generally, for the benchmarks, stiff walls will be used. An accurate impact feeling is provided by an *instantaneous* increase of the feedback force with no *viscosity effect* before.

Remaining in the stability domain, generally, the controller gains can be tuned in order to guarantee the desired behavior and performances for one case (free or restricted motion) with the price of the completely loss of the other case performances and behavior. More precisely, in free motion, the use of *small* gains guarantee the desired behavior in terms of tracking error and transparency. In the opposite, in order to have *good* performances in restricted motion, *high* gains should be used. At this point, for the systems affected by time-delays, if there are no special techniques used (as, for instance, gain scheduling or Smith predictor), a “trade-off” must be made in order to obtain the *optimum* behavior in both cases using the same gains. The DC motors must be carefully chosen in order to provide the expected feedback force. *Undersized* motors will be saturated quickly, leading a decrease of overall system’s performances.

Regarding the physical limitation of the system, the response may be slow due to the *inert reaction* of the motors. As it was already pointed out in Chapter 1, fast direction changes may result in a slow system response, see Figure 2.22.

In terms of safety, an important attention should be payed for the maximum force feedback. More precisely, a *high* force feedback could result in harmful actions for the human operator. For example, when working with virtual environments, sometimes the calculated force may induce high shocks or continuous forces, which can cause injuries to the human operator. With these considerations, the force feedback must be limited in order to avoid possible accidents.

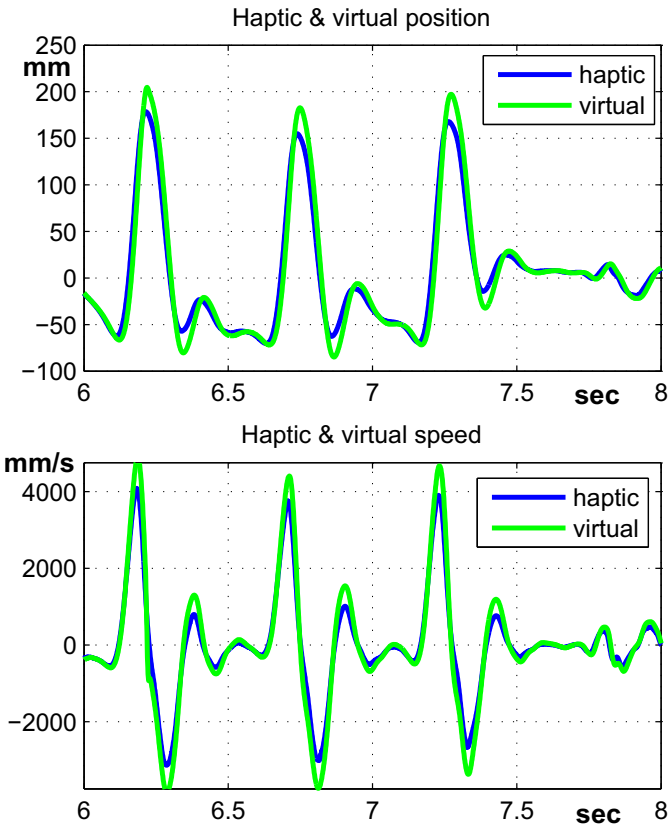


FIGURE 2.22: Fast direction changes.

Regarding the encoders, the resolution must be also taken into account. Even if nowadays, most of the encoders provide high resolution measurements, since teleoperation and haptic systems are working based on position and velocity, a high resolution measurement is required. The lack of accuracy may decrease the systems performance.

2.6 Conclusions

In this chapter, the theoretical tools for analyzing the stability of the delayed systems in different configurations, as well as the physical limitations and tuning clues from a practical point of view have been presented. More precisely, the chapter started with the analysis of the trade-off between stability and transparency for teleoperated systems, then the Proportional-Derivative (PD) controllers under time-delays used in haptics were analyzed and discussed, and finally, a specific analysis for the Smith predictor-based control used in haptics was proposed. The last section has introduced general aspects about the limitations of a real system and some practical hints for the tuning.

In Appendix B a method of analyzing the trade-off between stability and transparency based on the 4 channel control scheme used in teleoperation was presented. Using a geometrical representation, the proposed method *draws* an allowable gains domain, satisfying both conditions of stability and transparency. Several examples from teleoperation were proposed in order to complete the presentation.

As a general conclusion, this chapter has proposed the necessary theoretical tools for further developments in haptic systems under time-delays. Based on these results, the next two chapters presents two *new* methods for improving the system performances.

Chapter 3

Smith Predictor with distance feedback

3.1 Problem analysis

As mentioned before, a reliable method in solving time-delay problems similar to the one considered in the thesis is represented by the Smith predictor [174]. Based on the theoretical development presented in the previous chapter, a new approach adapted to haptics is detailed here. In haptics, as in teleoperation, the problem of time-delays and their effects on stability and transparency remains open (in [128] a method for improving the transparency is proposed by using a state prediction scheme for compensating the network communication delays). In haptics there are two time-delay sources: the communication channel and the processing time for the virtual reality environment. More precisely, as already highlighted, the delay effects can be felt in free motion by the viscosity *phenomenon*, while in restricted motion the contacts are not stiff.

In this chapter, a Smith predictor-based control [174] using the distance until a possible collision as feedback is proposed. Generally, the Smith predictor is working correctly when the delays are fixed and known and also the model used in the predictor is close to the real model and stable. The central idea is to use a predictor just on the haptic side in order to compensate the viscosity effect and to provide an accurate feeling in case of contacts.

Generally, for many systems affected by time-delays the Smith predictor-based control provides a reliable solution. There exist two main points that must be carefully analyzed when using this solution. The first one is the time-delay variation; under constant and known time-delays the Smith predictor projects the problem in the ideal case (the time-delays are not present). In order to guarantee a constant time-delay, a classical technique is to impose a fix quantum of time for each iteration, corresponding to the maximum time-delay. These solutions give interesting results when the delays are perfectly known and predictable. In the case of an unknown communication network delays, as the Internet®, the delays can no longer be predicted so accurately.

Considering a *larger* maximum time-delay may decrease the system performances substantially. To overcome this problem, passivity based solutions [102, 114, 191], distributed modeling or time varying solutions [103, 183] are only a few approaches proposed in the literature.

In Section 2.4, a complete study of the Smith predictor control-based for haptic/teleoperated systems under uncertain time-delays was presented using a distribution modeling approach.

The second point that must be carefully analyzed is the consistency of the plant model used in the Smith predictor. More precisely, an inconsistent model leads to an incorrect prediction, which can further turn into a decrease of performances or even a complete loss of stability. It can be shown that the Smith predictor has *some tolerance* at some model inconsistencies, but still remains *quite vulnerable* [44, 77, 95].

In the haptics and teleoperation cases, the model changes completely, from free to restricted motion or vice versa. In the sequel, the main advantages and inconveniences of using a Smith predictor-based control for haptic system will be highlighted.

It is worth noting that mechanical systems in presence of unilateral constraints are considered. Thus, a non-linear hybrid dynamics is analyzed, where the autonomous switch of dynamics are generated by the passing from free to constraint motion or reversed. In order to not complicate too much the problem, the transition between the two types of motion will not be particularly analyzed. Nevertheless, the model has to be changed on the predictor side when the dynamics changes. For solving this problem, a new approach is presented, by using the available information on the distance from the virtual reality simulator and introducing it in the predictor in order to maintain the similitude between the real and the predicted model.

In the previous chapter, a complete stability analysis of Smith predictor-based scheme for haptic system was presented, showing that *good* results can be obtained for different types of delays' distributions. These results correspond to the free motion case. In restricted motion, the results are no longer valid since contact force must be added in the dynamics. More precisely, the system is stable in steady state, but the problem *comes* from the transition phase. The main problem is recalled to the model considered in the Smith predictor which is no more similar to the virtual reality model in the case of contacts. Unfortunately, the basic form of the Smith predictor can compensate the delays only in free motion, in restricted motion the model is no longer valid, since the virtual environment is not included and will induce a poor impact sensation and, in some cases, an unstable behavior. More precisely, the system is stable in both cases, but the problem is due to the transition from free to restricted motion, the control law is not sufficient to guarantee the requested properties and will introduce long time oscillations until the system becomes stable. Figure 3.1 illustrates the system behavior when passing from free to restricted motion using a *classic* Smith predictor. As shown in the literature (see [119]), the study of this transition is very challenging even in the free-of delay case.

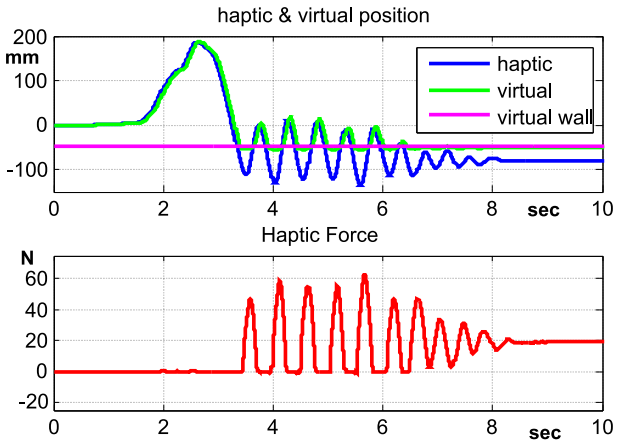


FIGURE 3.1: Oscillating behavior of the system when passing from free to restricted motion, using *classic* Smith predictor control.

Resuming all these aspects, the Smith predictor-based control represents a reliable solution in free motion. Once the system switches from free to restricted motion, the transition may induce long time oscillations, or sometimes even instability. It is worth mentioning that according to the experimental results in free motion for different uncertain time-delays, presented in Section 3.4, the Smith predictor-based solution provides important improvements from the transparency point of view, i.e. in terms of end user perception the behavior approaches to the non-delayed case. In order to make the Smith predictor-based approach a solution in *all situations*, the next section proposes a new approach for updating/changing the model used in the predictor.

3.2 Proposed solution

In haptics and teleoperation, as for the majority of the time-delay systems, the Smith predictor or its variants represent an appropriate solution for the control law. Haptic and teleoperated systems have two functioning cases (free and restricted motions) which makes the use of the Smith predictor more difficult since the model on the virtual (slave) side changes. More precisely, the model used in the predictor is working *correctly* in free motion, while in restricted motion is no longer accurate since there is an important difference between the predicted and the real model. A solution for such a problem is proposed in [37] for the teleoperation framework. They suggested the use of two models, corresponding to free and restricted motions, and a switching method based on the tracking error between the master and the slave position (i.e. to assume that there is contact when the tracking error is increasing over a certain limit). The advantage in haptics compared to the teloperation framework, is represented by the additional information available concerning the environment, which can be further used for improving the control law. The virtual reality can provide precise information about the neighborhood of the controlled object in order to anticipate the contacts, and based on the model update.

In the case of uncertain delays, it is very important to have an accurate model in the Smith predictor, in order to assure the stability of the system.In the sequel, a specific solution for haptics is proposed in order to improve the Smith predictor.

Let's remind the equations (1.1)-(1.2), representing the starting point, given by the *classical* dynamic (nonlinear) equations of motion for two similar robots in the haptics framework:

$$M_h(x_h)\ddot{x}_h(t) + B_h(x_h,\dot{x}_h)\dot{x}_h = -F_h(t) + F_{op}(t), \tag{3.1}$$

$$M_v(x_v)\ddot{x}_v(t) + B_v(x_v,\dot{x}_v)\dot{x}_v = -F_v(t) + F_e(t), \tag{3.2}$$

where x_h, x_v are the haptic interface/virtual object position, F_{op}, F_e are the human/environmental forces, F_h, F_v are the force control signals, M_h, M_v are the symmetric and positive-definite inertia matrices, and B_h, B_v are the Coriolis matrices of the haptic interface and virtual object systems, respectively.

According to the Smith predictor control scheme presented in Figure 2.13, the new blocks added result in a third robot which has to compensate the delay effects of the virtual object. The *classic* Smith predictor used for this case writes as follows:

$$M_s(x_s)\ddot{x}_s(t) + B_s(x_s,\dot{x}_s)\dot{x}_s = -F_s(t), \tag{3.3}$$

where x_s is the predicted object position, F_s is the predicted force control signal, M_s is the symmetric and positive-definite inertia matrix and B_s is the Coriolis matrix, both equal to the virtual reality object mass and Coriolis matrix, respectively.

In the free motion case, since $F_e(t)$ is equal to 0, the predictor model is similar to the one of the virtual object. Once the environmental force $F_e(t)$ is no longer equal to 0, i.e. there exist some hard contacts with the environment, the model used in predictor is not accurate anymore.

Consider now the controllers's equations (2.4)-(2.5) for the standard system, illustrated in Figure 2.2, given by:

$$F_h(t) = \underbrace{K_{d_h}(\dot{x}_h(t) - \dot{x}_v(t-\tau_2))}_{\text{delayed D-action}} \underbrace{+K_{p_h}(x_h(t) - x_v(t-\tau_2))}_{\text{delayed P-action}}, \tag{3.4}$$

$$F_v(t) = \underbrace{K_{d_v}(\dot{x}_h(t-\tau_1) - \dot{x}_v(t))}_{\text{delayed D-action}} \underbrace{+K_{p_v}(x_h(t-\tau_1) - x_v(t))}_{\text{delayed P-action}}, \tag{3.5}$$

where τ_1, τ_2 are the forward and backward time-delays and $K_{p_{h/v}}, K_{d_{h/v}}$ are the PD control gains.

Since the Smith predictor is used only on the haptic side, $F_h(t)$ rewrites as follows for this case:

$$F_h(t) = K_{d_h}(\dot{x}_h(t) - \dot{x}_v(t-\tau_2) + \hat{\dot{x}}_v(t-\tau_S)) - \hat{\dot{x}}_v(t) + K_{p_h}(x_h(t) - x_v(t-\tau_2) + \hat{x}_v(t-\tau_S) - \hat{x}_v(t)) \tag{3.6}$$

where $\hat{\dot{x}}_v$, $\hat{x}_v$ represent the estimated velocity and position for virtual object and τ_S represents the time delay considered in the Smith predictor. The estimation is made using a similar model to the one running on the virtual reality environment.

From these equations, it can be seen that there exists a perfect similarity between the predicted control and the real one. Let's remind the equations (2.72)-(2.74) describing the response of the system presented in Figure 2.14:

$$X_h(s) = P_h(s)\Big(F_{op}(s) - C_h(s)\big(X_h(s) - e^{-\tau_2 s}X_v(s) + (e^{-\tau_S s} - 1)X_S(s)\big)\Big), \tag{3.7}$$

$$X_v(s) = P_v(s)\left(-F_e(s) + C_v(s)\left(-X_v(s) + e^{-\tau_1 s}X_h(s)\right)\right), \tag{3.8}$$

$$X_S(s) = P_S(s)C_S(s)(X_h(s) - X_S(s)), \tag{3.9}$$

where:

$$P_h(s) = P_v(s) = P_S(s) =: P(s) = \frac{1}{s(ms+b)}, \quad C_h(s) = C_v(s) = C_S(s) =: C(s) = K_p + K_d s.$$

From equations (3.7)-(3.9), it can be seen that in case of hard contacts, i.e. $F_e \neq 0$, $X_v(s)$ and $X_S(s)$ are no longer similarly defined, since F_e does not have an equivalent component.

With all these considerations, the intuitive solution is to find a way to estimate the force on the predictor side. Here, a method to preserve the accuracy of the system's model used in the predictor is proposed, using the distance until collision available in the virtual reality scene. Based on the information received from the virtual environment, it is possible to predict the impact moment and to maintain the predictor's model *authenticity* resulting in an accurate system.

The main idea of the proposed solution is to introduce into the Smith predictor the environmental force F_{es} by using the distance between the controlled virtual object and other objects in the scene. Figure 3.2 illustrates the approach.

Based on Figure 2.14, in this case, the control scheme is drawn in Figure 3.3. As it can be seen in Figures 3.2-3.3, the information is also affected by communication delays, but this is not a critical problem. Even if some changes appear during the time period of the delay, the model will still be able to guarantee the desired behavior. In Section 3.4 these situations will be experimentally presented.

Introducing the environmental force in the predictor, the equation (3.3) rewrites as follows:

$$M_s(x_s)\ddot{x}_s(t) + B_s(x_s, \dot{x}_s)\dot{x}_s = -F_s(t) + F_{es}, \tag{3.10}$$

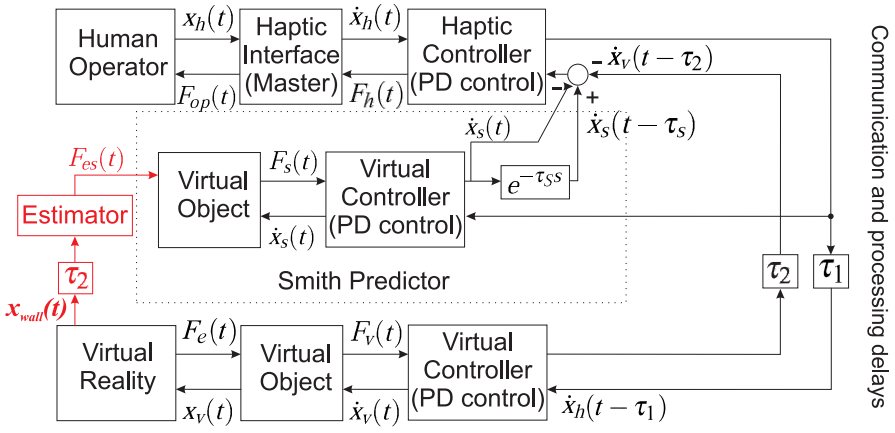


FIGURE 3.2: Smith predictor-based control with distance feedback for network based haptic systems

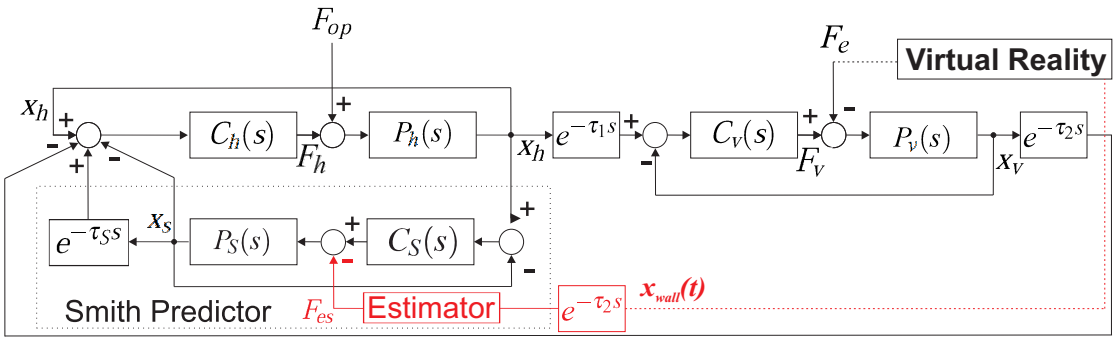


FIGURE 3.3: Control scheme of Smith predictor-based system with distance feedback for haptics

where F_{es} represents the estimated environmental force calculated based on the distance extracted from the virtual reality:

$$F_{es} = K_{wall}(X_S - \widetilde{X}_{wall}) + B_{wall}\dot{X}_s, \tag{3.11}$$

where K_{wall} and B_{wall} represent the stiffness and damping used to compute the virtual force environment, $\widetilde{X}_{wall}$ is the virtual wall position received from the virtual scene (on x,y,z) and X_s, $\dot{X}_s$ are the virtual object position and velocity estimations used in the Smith predictor.

Introducing F_{es} in equation (3.9), this rewrites as follows:

$$X_S(s) = P_S(s)\left(F_{es} + C_S(s)(X_h(s) - X_S(s))\right). \tag{3.12}$$

Now, after introducing F_{es}, the Smith predictor's equations are similarly defined in all situations - free and restricted motions. The method proposed here provides a reliable solution for *updating/changing* the model used in the predictor, using the information about the distance until

possible collisions, available in the virtual reality scene. Since the major problem of the Smith predictor-based control scheme was the inconsistency of the model, based on the information received from the virtual environment, it is possible to predict the impact moment and to have a correct model in the predictor, resulting in a more accurate system.

The next section will introduce the experimental platform that will be used for the validation of the proposed method.

3.3 3-DOF experimental platform

In this section, the experimental platform that will be used for the validation of the method presented previously, is briefly described. The main objective of this platform is to guarantee a full control in the presence of communication time-delays. In order to ensure a full control of the communication delays and processing time, all the control algorithms (for haptic interface/virtual object) and virtual environment simulations will be run on the same computer.

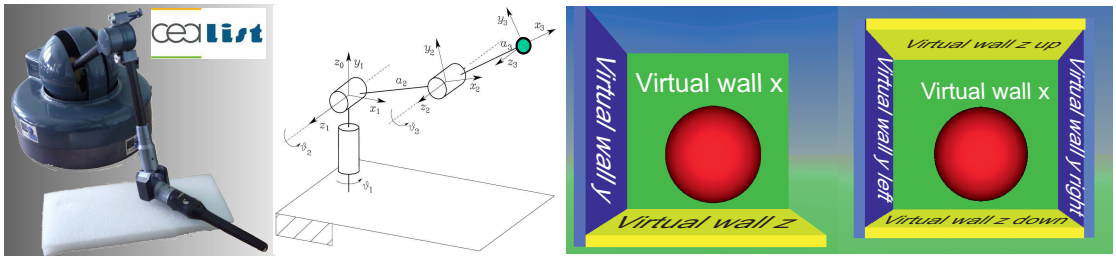
The haptic interface, Figures 3.4a and 3.4b, consists of three direct-drive motors and three optical quadrature encoders with 1000 pts/rev (with a gear ratio of 1/10). The controllers and the virtual simulation are running in real time mode (on RTAI Linux) with a sampling time of 1 ms.

Figures 3.4c and 3.4d illustrate the two virtual scenes (*simple* environment/virtual box) that will be considered for the experiments, as well as the virtual object. The virtual object is modeled to be a spherical mass (equal to the haptic interface mass) ($M_h = M_v$). The environmental force (F_{e_x}, F_{e_y}, F_{e_z}) resulting in case of an impact with the virtual environment (virtual walls) is defined by the following equation:

$$F_e = K_{wall}(X_v - X_{wall}) + B_{wall}\dot{X}_v, \tag{3.13}$$

where $K_{wall} = 20000\ N/m$ and $B_{wall} = 10\ Ns/m$ represent the stiffness and damping used to compute the virtual force environment in this case, X_{wall} is the virtual wall position and X_v, $\dot{X}_v$ are the virtual object position and velocity.

This experimental platform represents a simplified version of a *real* system, but it appears to be sufficient for haptics study purposes. A *real* system, as described in [8], consists of an appropiate workstation for the CAD application linked on one side to dedicated visualization cluster, and on the other side to a real-time physical simulation. The cluster is in charge with the physical simulation and is linked to a motion capture system, and the haptic interface. According to [8], on a 8 core machine, using VPS++, the computation time is lower than 2 *ms*, while using LMD++ the computational time may increase up to 10 *ms* for the virtual model considered. The proposed system was used in the automotive industry in order to study the feasibility of



a.Haptic Interface. b. Robot Geometry. c. *Simple* Virtual Scene d. Virtual Box Scene

FIGURE 3.4: Haptic System.

some manual operations like, painting processes, door fitting, accessibility for assembly of peripheral components, training and ergonomic analysis. Another complex application, designed for testing aircraft engines in order to reduce the development costs and to avoid the necessity of physical mock-ups for maintainability using an integrated haptic device and VR system is presented in [27].

3.4 Results and analysis

In this section, the new approach - *Smith predictor-based control with distance feedback* will be experimentally tested and analyzed in different scenarios and under different types of time-delay. All the experiments are carried out by a human operator, explaining why the conditions will not be exactly the same in all experiments. In order to create an efficient way of testing the proposed method, multiple scenarios from *simple* to *complex* were defined:

1. free motion (random motions on each axis),
2. restricted motion (wall contacts on each axis),
3. contact with moving objects, i.e. a sinusoidal *moving* walls[1] will be used in order to provide a more realistic scenario,
4. free and restricted motion inside a virtual box (random motions with or without contacts on each axis and multi-point contacts[2]),
5. free and restricted motion inside a virtual box with sinusoidal *moving* walls (random motions with or without contact on each axis and multi-point contacts),
6. free to restricted motion and vice versa transitions under random uncertainties $(-5mm; +5mm)$ applied on the *distance feedback*, with sinusoidal *moving* walls,

[1]Each wall will be moving sinusoidally based on the following function: $x_{moving} = x_{wall} + 5sin(t)\ mm$.
[2]Multi-point contact defines the simultaneous contacts on more than one axis

7. free to restricted motion and vice versa transitions inside a virtual box, under random uncertainties applied on the *distance feedback*, with sinusoidal *moving* walls.

The first scenario will point out the basic functionalities of haptic systems, using the proposed control method under different types of delays. The second scenario is meant to provide a more interesting situation, when the virtual scene is no longer static. The third and fourth scenarios propose a more challenging experiment, since the complexity level of the environment has been increased and fast model changes are required. The purpose of the last two scenarios is to test the proposed method in what we consider to be the *worst* case, since the model estimation is perturbed. More precisely, a perturbation will be added on the wall position (x_{wall}) used in the feedback.

Based on the analysis of the Smith predictor-based control used for haptic systems, presented in Section 2.4, considering the analyzed types of time-delay, five experimental time-delay circumstances will be presented:

- fixed and known time-delays,
- uncertain delays: constant uncertainty,
- uncertain delays: uniform distributed time-delays,
- uncertain delays: gamma distributed time-delays with gap,
- uncertain delays: normal distributed time-delays.

Resuming, the proposed method will be tested from *simple* to *complex* cases using different types of time-delays, as well as different scenarios.

Remark 3.1. *On the gain tuning.* The tuning of the gains will be made in order to obtain the *closest* results to the *ideal* case, respecting the stability limits for each distribution type. It is worth mentioning that K_p gain plays the main *role* in terms of performances, since both free and restricted motion are directly influenced by its value. More precisely, a small value of K_p guarantees small force feedback in free motion, while the restricted motion is penalized, resulting in a poor impact feeling. Generally, K_p is chosen as big as possible in order have low impact on the restricted motion. The K_d gain is chosen fulfilling the stability conditions for each case.

3.4.1 Constant time-delays

The experiments will start with the case of constant time-delays. Under the assumption that there are no model uncertainties or variation, when using the Smith predictor, the delayed systems

come to the free-off delay case, generally considered to be the *ideal* case. The main objectives are to test the capabilities of the method under *ideal* conditions and also to provide a *starting* point for the cases affected by time-delay uncertainties.

The gain tuning was made in order to obtain *close* performance to the *ideal* case (free of delay). The following controller's gains are used for all the experiments presented in this subsection, corresponding to the constant time-delays case, respecting the stability region presented in Figure 2.17, section 2.4.5:

$$K_p = 1800\ N/m, \qquad K_d = 80\ Ns/m,$$

and the system's time-delays $\tau_1 = \tau_2 = 50\ ms$, and $\tau_S = 100\ ms$.

Scenario 1. Figure 3.5 presents the free motion case for constant time-delays.

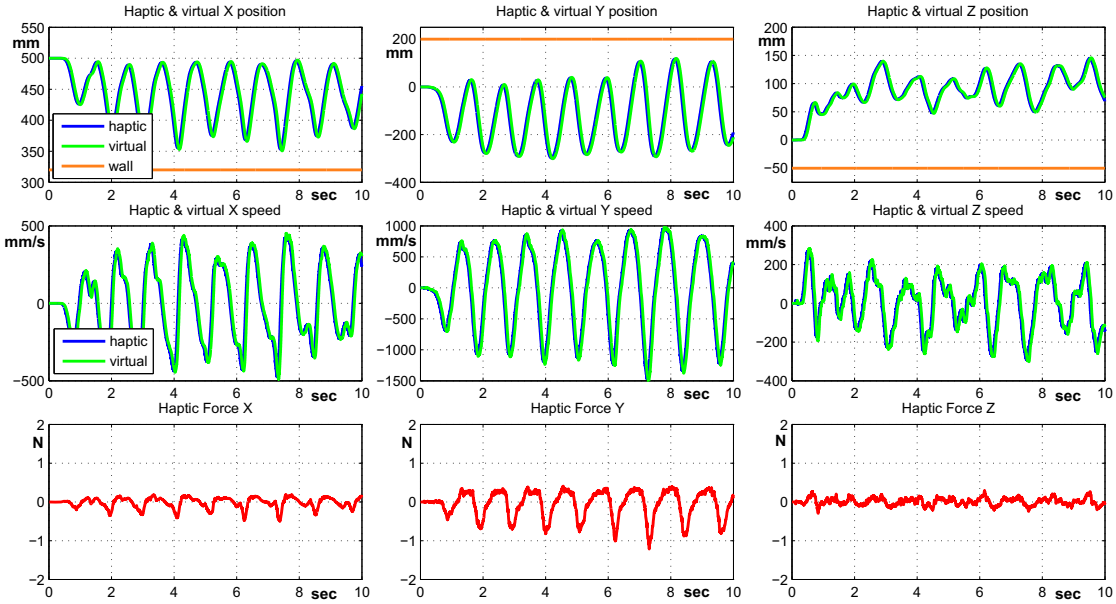


FIGURE 3.5: Free motion under constant time-delays.

As expected, this case comes to the ideal circumstances, when there exist no time-delays and no motion switches. The results match perfectly the ideal results, i.e. the force feedback is almost *null* ($< 0.5\ N$).

Scenario 2. Next, Figure 3.6 presents the restricted motion case, when the time-delays are constant.

This example presents the first *use* of the method proposed in this chapter. The model used in the predictor was correctly estimated and the results are almost identical to the ones proposed in the ideal case. It is worth pointing out that the Smith predictor in its basic form was inducing long time oscillations in the case of hard contacts (see Figure 3.1), while here the restricted motion is correctly handled.

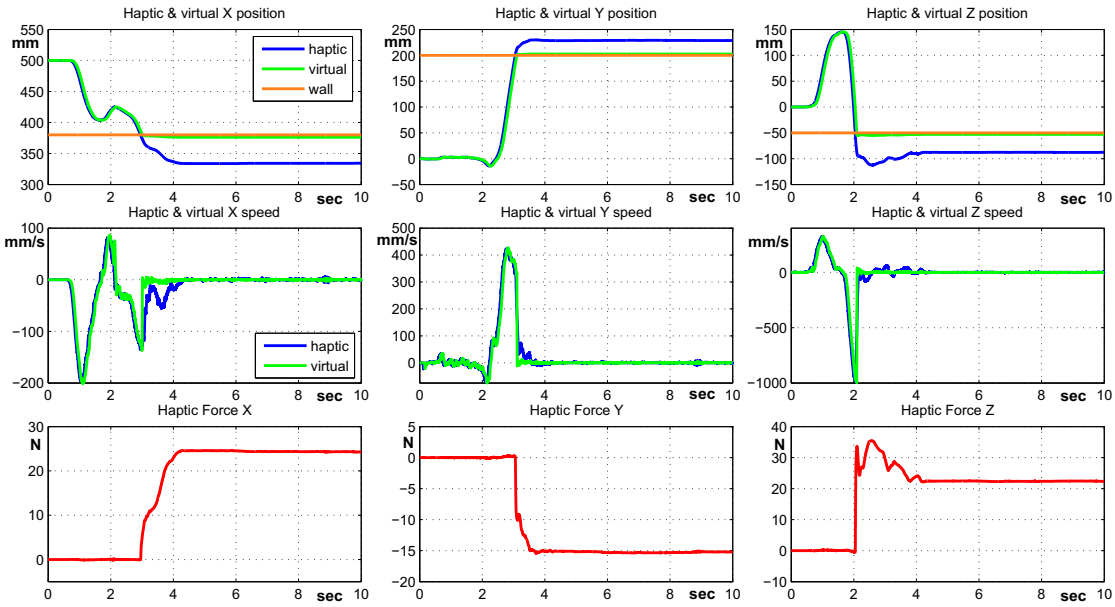


FIGURE 3.6: Restricted motion under constant time-delays.

Scenario 3. In the sequel, since the proposed method is based on the distance between the controlled virtual object and other objects from the virtual scene, a *moving* object will be considered in order to illustrate the method capabilities for a dynamic situation. Figure 3.7 presents the results.

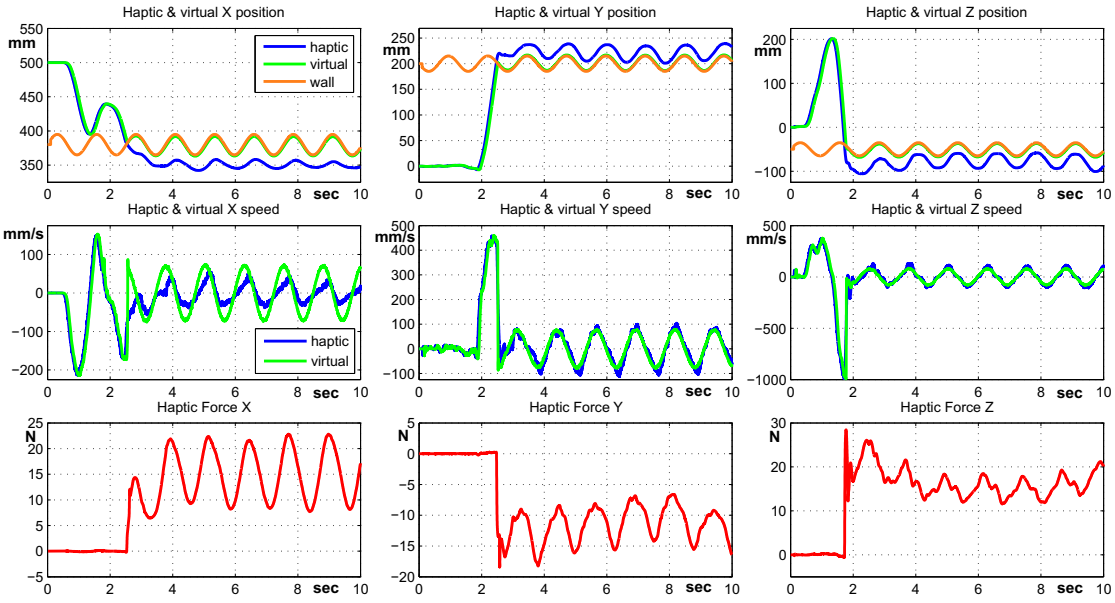


FIGURE 3.7: Restricted motion under constant time-delays and *moving* object.

The impact moment is perfectly reproduced, and furthermore, while maintaining the force, the human operator is able to feel the moves of the object. The force oscillations are given by the sinusoidal move of the virtual wall.

Scenario 4. As it was presented in the beginning of this section, a perturbation will be also added on the position feedback (used in the Smith predictor) in order to test the vulnerability in case of uncertainties. Figure 3.8 presents the results.

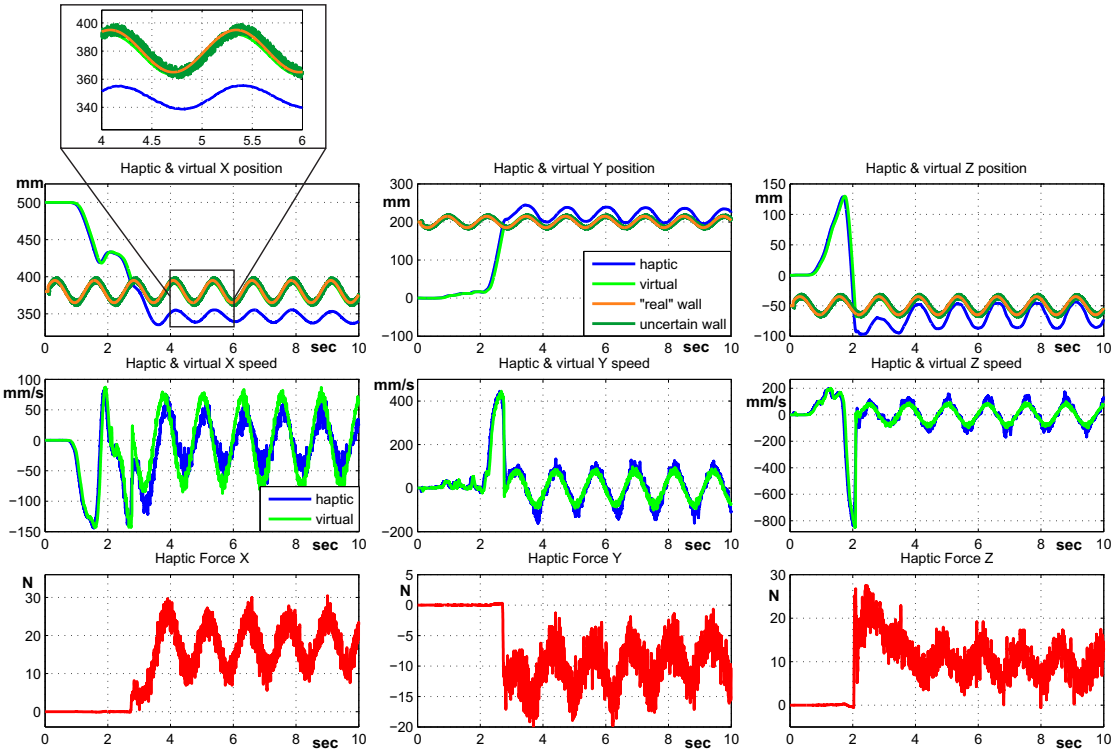


FIGURE 3.8: Restricted motion under constant time-delays *moving* object and uncertain feedback.

It is important to mention that the uncertain wall is shifted from the "real" wall because the information is also affected by time delays, as it was presented in the zoom of Figure 3.2. Similar to the previous cases, the impact moment is perfectly provided and also the object's moves are felt by the human operator. The presence of the perturbation induces a vibration that can be slightly felt by the human operator. The vibrating effect is visible due to the large thickness of the force, caused by the fast oscillations. For larger perturbations ($> \pm 10$ *mm*), the vibrations will also increase, providing an unpleasant manipulation. It is worth mentioning that in *real life* situations, this case has a small probability of occurring.

For the last three examples of this subsection, a more complex virtual scene will be considered. More precisely, a *virtual* box, as it was presented in Figure 3.4.d, will be used.

Scenario 5. Figure 3.9 exposes the random moves inside a *virtual* box.

The main objective of this experiment is to point out the method capabilities under *fast* changes conditions. Since the moves are made inside the box, there exist many switches from free to restricted motion and viceversa, and also many multi-point contact situations. As it can be seen in Figure 3.9, the results appear to be consistent with the reality constraints, the human operator

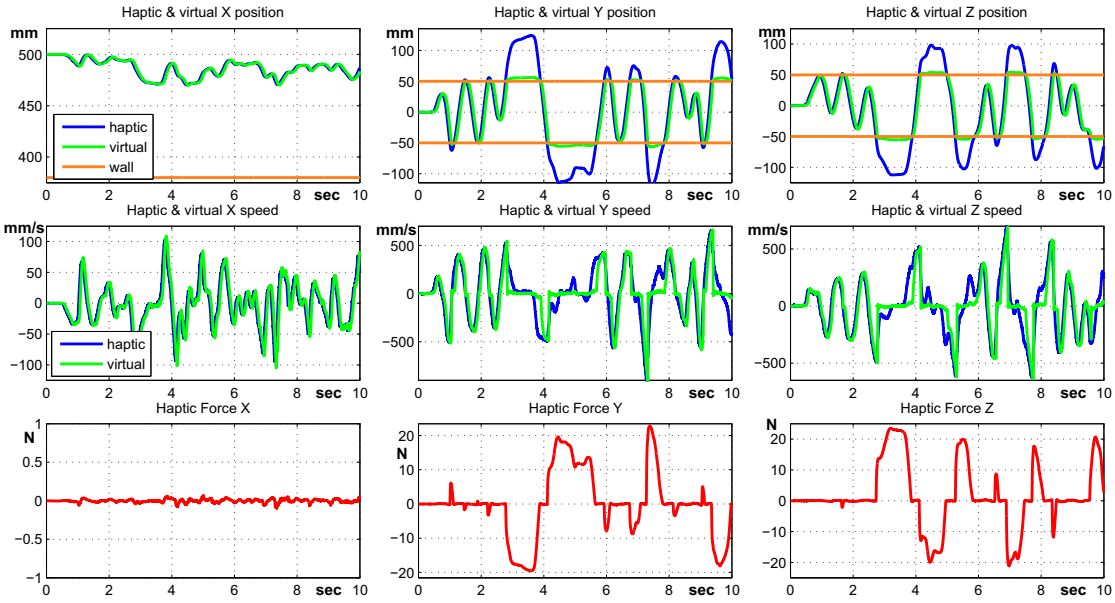


FIGURE 3.9: *Random* moves inside a virtual box under constant time-delays.

can feel accurately all the impacts[3], and the force depends directly to the speed at the contact moment, providing a realistic behavior.

Scenario 6. This example will point out the case when more virtual objects from the scene are moving. In order to illustrate this situation, the walls of the virtual box will move sinusoidally. Figure 3.10 presents this case.

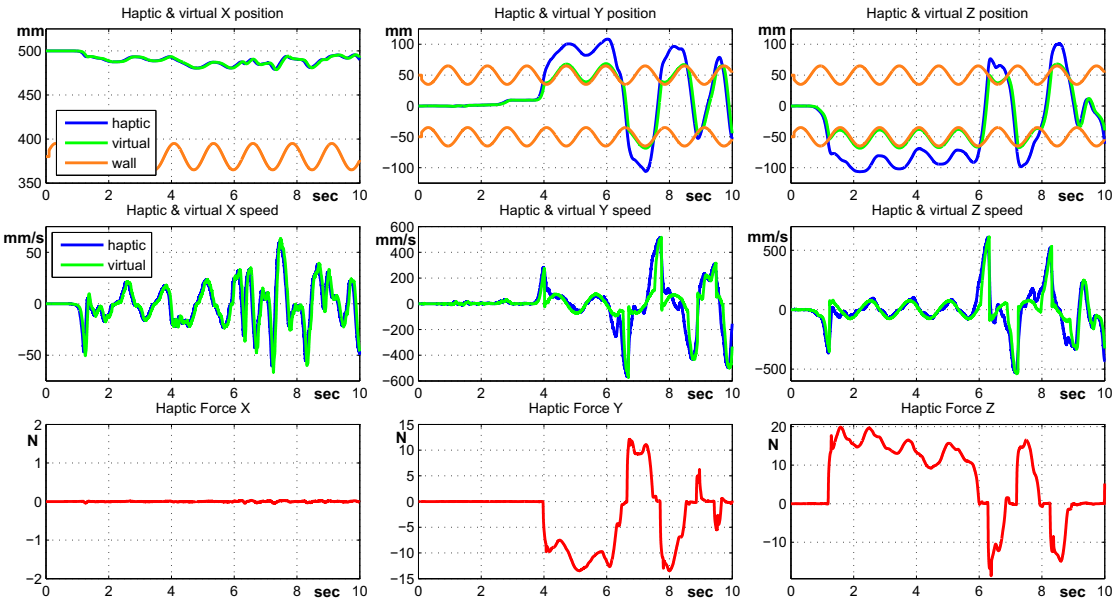


FIGURE 3.10: *Random* moves inside a virtual box under constant time-delays and *moving* walls.

[3]The impacts correspond on the graphic at each exceed of the virtual wall limits

Here, the goal is to test the method in more realistic dynamical situations. The obtained results are consistent with the reality in the sense that the impact can be felt in a "realistic" way and also the multi-point contacts situations are correctly provided. It is worth mentioning that if the force is maintained (see Figure 3.10, Z axis starting from ≈ 1.3 to ≈ 6 *sec*), the user is able to feel the moves of the virtual objects, i.e. in this case, the sinusoidal moves in all directions (x, y, z).

Scenario 7. In this example, a perturbation will be added on the feedback. Figure 3.11 introduces this case.

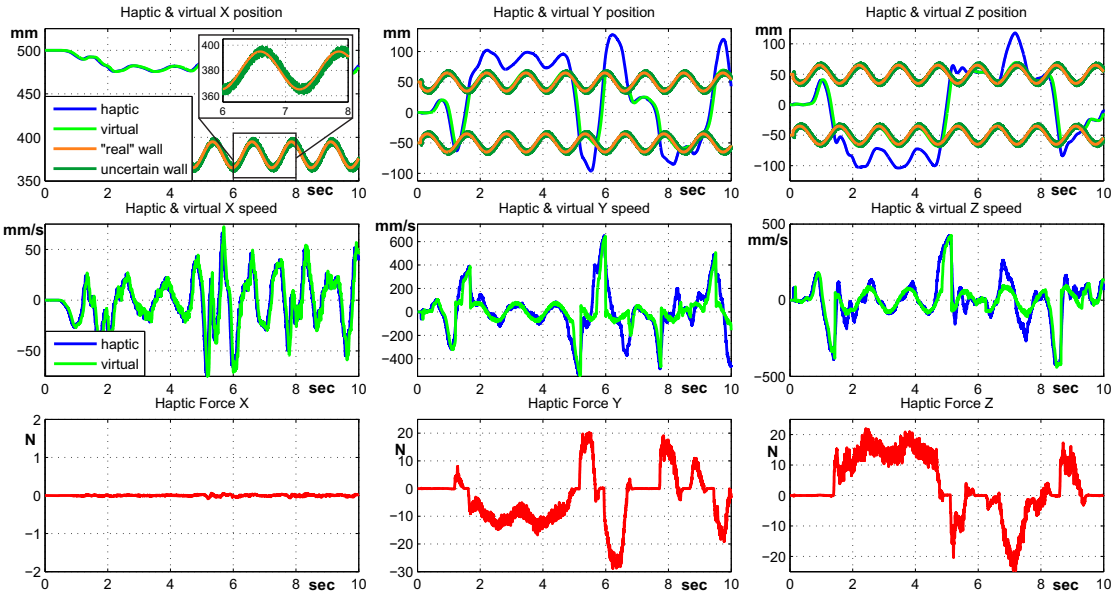


FIGURE 3.11: *Random* moves inside a virtual box under constant time-delays, *moving* walls and uncertain feedback.

As expected, the impact feeling is correctly provided and the switching between the two motions is guaranteed. Similar to the scenario 4, when the force is maintained (see Figure 3.11, Y and Z axes starting from 1.5 to 5 *sec*), there appears some vibrations (visible on the graphic due to the large thickness of the force on Y and Z axes) which is slightly felt by the human operator. Apart this vibration which can become critical for larger perturbations ($> \pm 10$ *mm*), all the desired performances are guaranteed.

This subsection presented the experimental results under constant time-delays for each proposed scenario. Since this situation corresponds to an ideal case, these results will be considered as reference results, and will be considered as a starting point for the next experiments under non-constant time-delays. In order to point out the improvements offered by this approach, it is important to consider the long time oscillating behavior of the Smith predictor in its basic form, presented in Figure 3.1. Also the method guarantees the performances not only in *simple* situations (restricted motion over a fixed virtual wall), but also for more complex environments with dynamical changes and under feedback perturbations.

3.4.2 Uncertain time-delays: constant uncertainty

Based on the theory presented in Subsection 2.4.6.1, a complete experimental exemplification is presented here. The backward and forward delays of the system will be considered to be constant and equal to 50 *ms*. The delay used in the Smith predictor will be considered to be different from the system's delay.

In each example, two situations will be addressed, depending on the uncertainty value:

- first, the uncertainty is considered to be $+10$ *ms* for each forward and backward delay, resulting in a 120 *ms* time-delay on the predictor side,
- second, the uncertainty is considered to be -10 *ms*, resulting in a 80 *ms* time-delay used in the predictor.

The following controller's gains are used for all the experiments presented in this subsection, corresponding to the constant uncertainties cases:

$$K_p = 1200\ N/m, \qquad K_d = 60\ Ns/m,$$

and the system's time-delays $\tau_1 = \tau_2 = 50$ *ms*. These gains were chosen in order to obtain the *closest* performance to the *ideal* case, while respecting the stability region from Figure 2.18.

Scenario 1. Figures 3.12 and 3.13 present the free motion case for the two uncertainties (120 *ms* and 80 *ms* respectively).

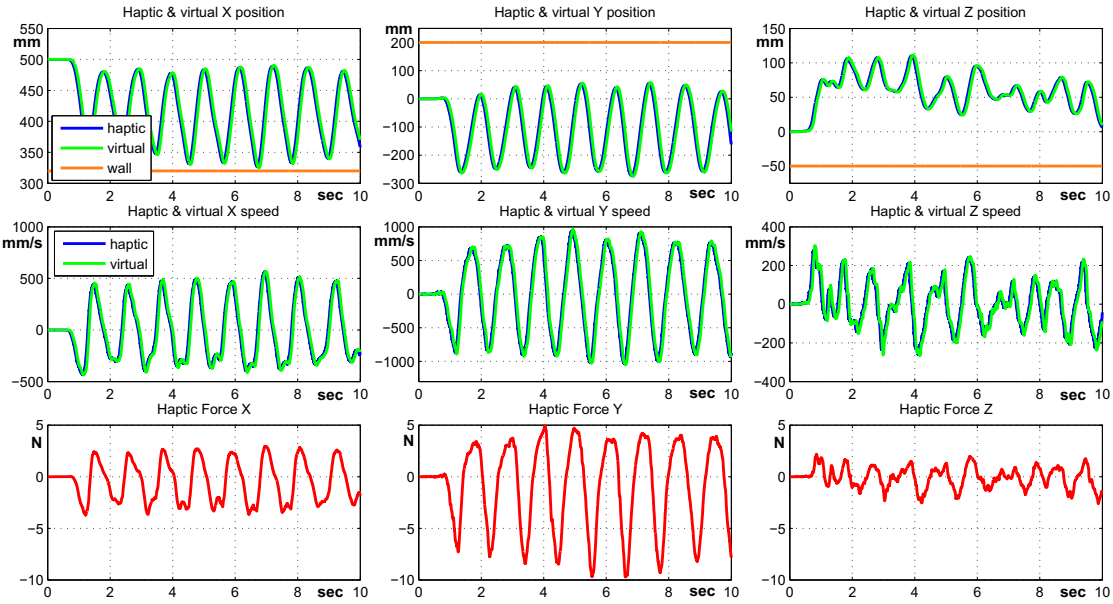


FIGURE 3.12: Free motion under constant uncertainty ($\tau_S = 120$ *ms*).

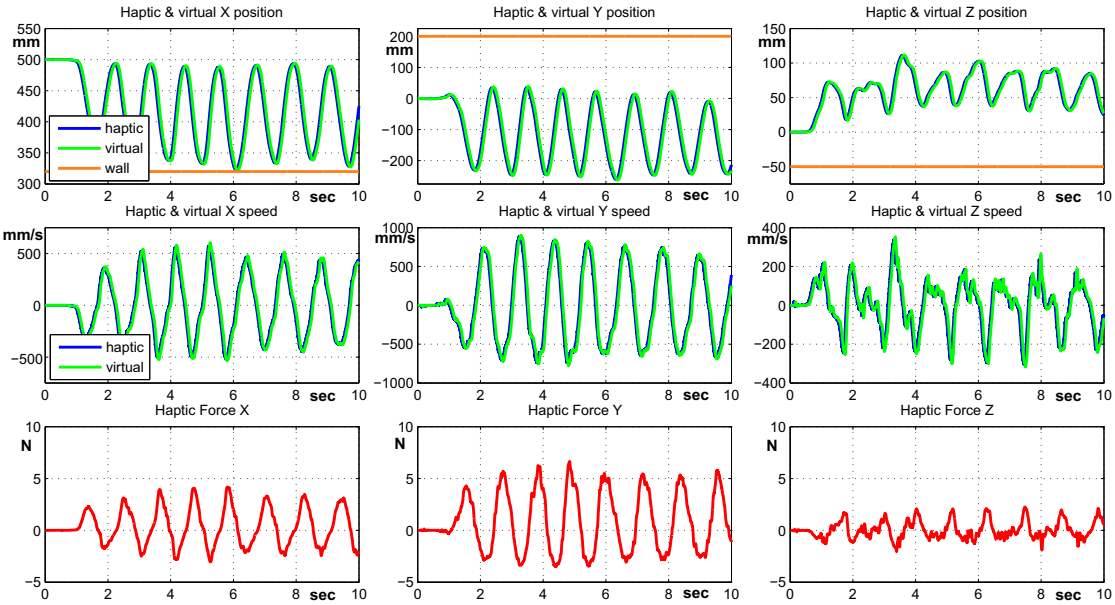


FIGURE 3.13: Free motion under constant uncertainty ($\tau_S = 80\ ms$).

As in the first case, when $\tau_S = 120\ ms$, the predictor is acting as a *movement compensator*, since the force and velocity have the same sense (see Figure 3.12 force and velocity for each axis). More precisely, in this case, the force is *helping* the human operator in moving the haptic interface. This phenomenon can be explained by the phase advance which appears when the delay considered in the Smith predictor is larger than the system's delay.

For the second case, when $\tau_S = 80\ ms$, the Smith predictor is working *normally*, with a force feedback smaller than 5 N (except some peaks due to direction changes at high velocities).

In the first case ($\tau_S = 120\ ms$), there exists no viscosity effect, more precisely the predictor acts as a friction and mass compensator. In the second case ($\tau_S = 80\ ms$), the viscosity effect is larger than in the case of constant delays, but it is still *low* ($< 5\ N$), providing an agreeable manipulation.

Scenario 2. In this example, the *simple* restricted motion case will be presented. Figures 3.14 and 3.15 illustrate this scenario for the two uncertainties (120 *ms* and 80 *ms* respectively).

In the first case ($\tau_S = 120\ ms$), the impact sensation is *more "violent"*, since the force is helping the human operator until the impact, and then is changing the sense instantaneously in order to reflect the impact (see Figure 3.14 Y and X axes).

In the second case ($\tau_S = 80\ ms$), the impact phenomena is typical for a haptic system, i.e. there is a *small* viscosity effect before the impact, since the force is acting in the opposite sense of the velocity.

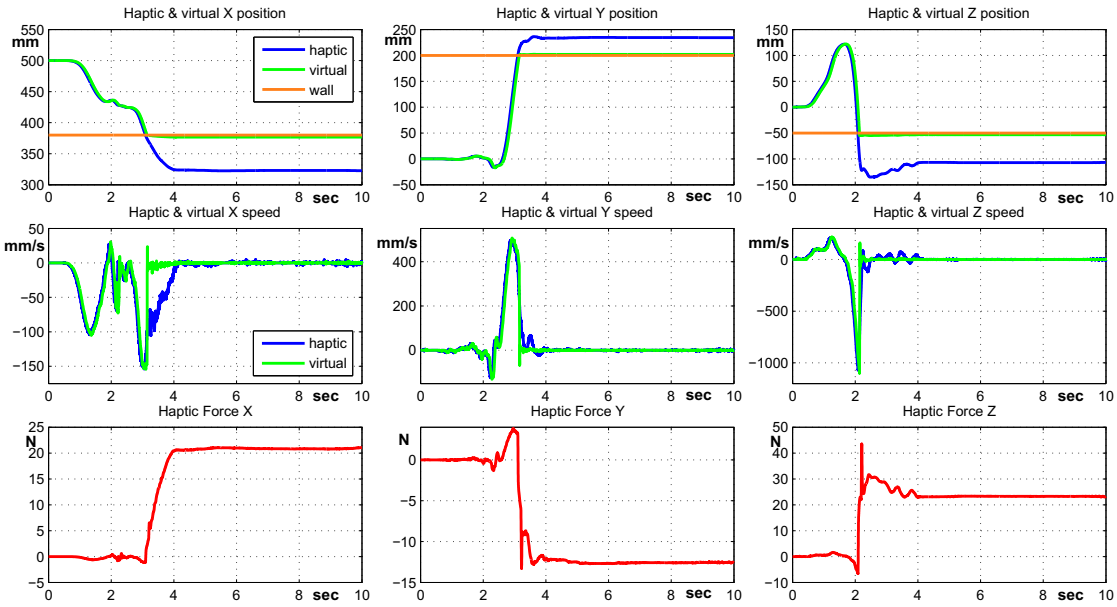


FIGURE 3.14: Restricted motion under constant uncertainty ($\tau_S = 120$ *ms*).

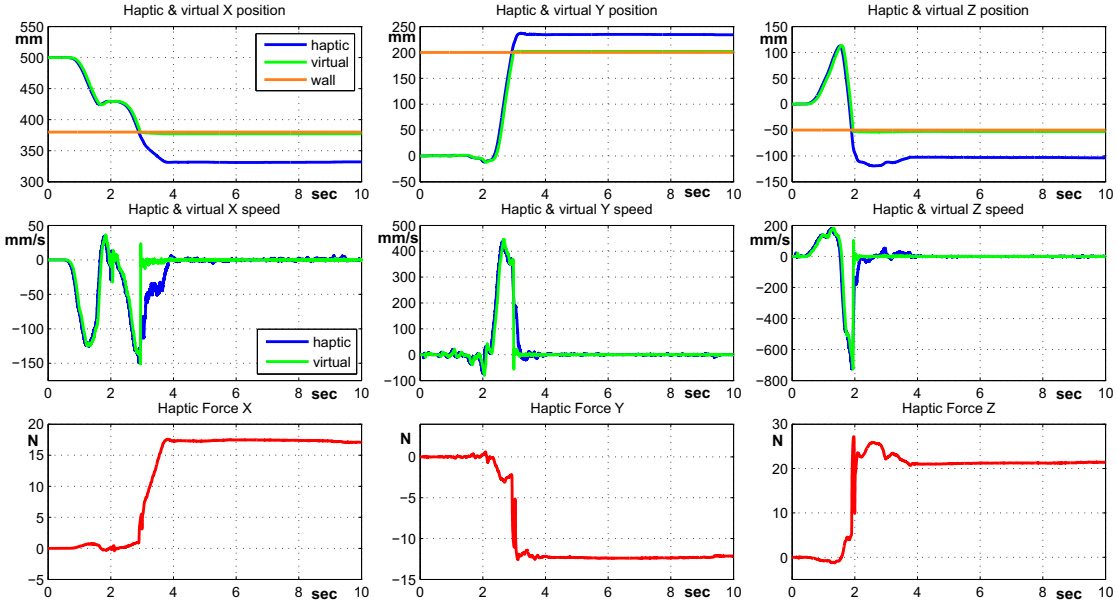


FIGURE 3.15: Restricted motion under constant uncertainty ($\tau_S = 80$ *ms*).

In both cases, the impact sensation is correctly provided and close to the ideal case. The only difference between the two uncertainties considered ($\tau_S = 120$ *ms* and $\tau_S = 80$ *ms*) is the speed at the impact moment, which results in a more important difference between the force before and after the impact. This phenomenon is the result of the phase advance of the first case. The overall performances are consistent in both cases, and the model used in the Smith predictor is correctly provided in free motion, as well as in restricted motion.

Scenario 3. This example presents the restricted motion, corresponding to the two uncertainties when the virtual object is considered to be moving, i.e. a sinusoidal variation will be considered.

Figures 3.16 and 3.17 show the results for this case.

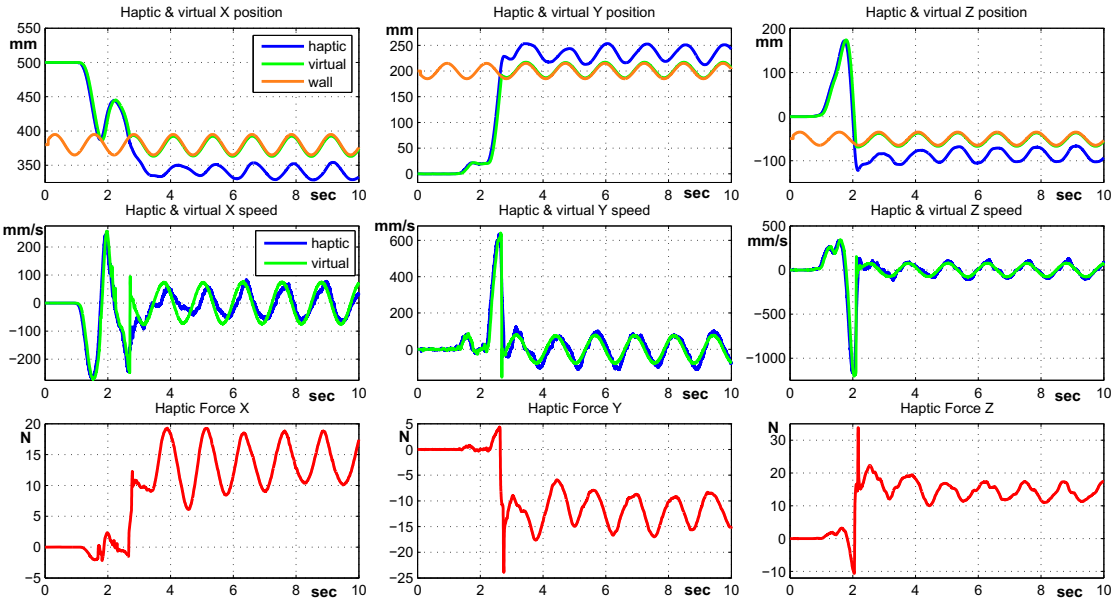


FIGURE 3.16: Restricted motion under constant uncertainty ($\tau_S = 120\ ms$) and moving virtual object.

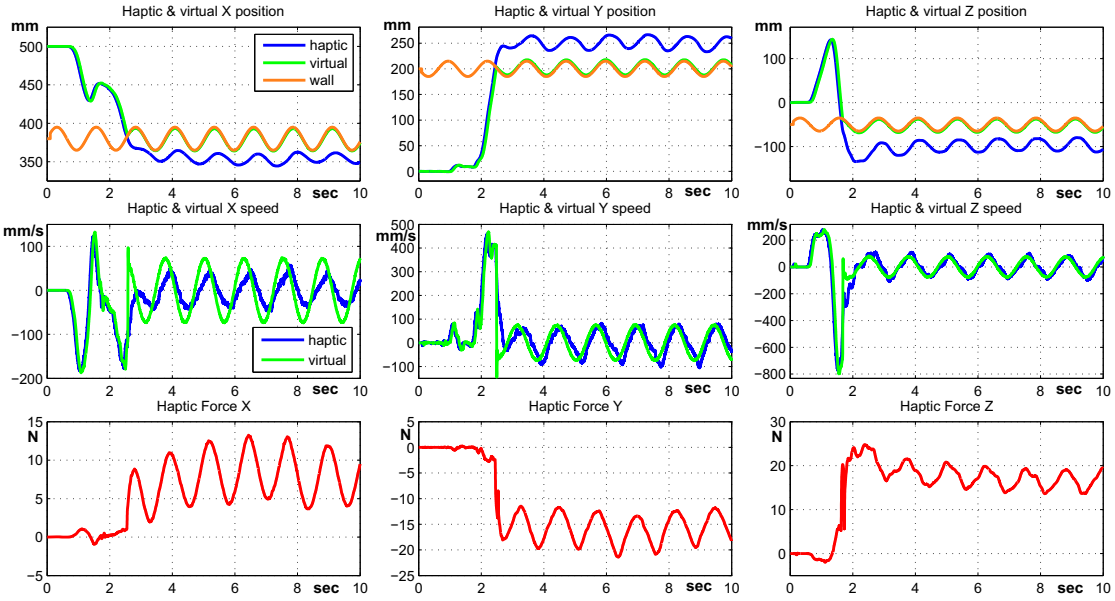


FIGURE 3.17: Restricted motion under constant uncertainty ($\tau_S = 80\ ms$) and moving virtual object.

Similar to the previous scenario, in both cases, the impact moment is correctly provided. In the first case ($\tau_S = 120\ ms$), the phase advance phenomenon (visible on the graphic due to the same sense between the force and velocity), provides a more important impact sensation since the difference between the forces before and after impact is larger than it is the second case ($\tau_S = 120\ ms$). Furthermore, if the human operator is maintaining the force (as both Figures

illustrate), the moves of the virtual object are felt at the haptic end, explaining the sinusoidal shape of the force.

Scenario 4. Now, some perturbations will be added on the position feedback, while preserving the same conditions from the previous scenario (restricted motion, corresponding to the two uncertainties when the virtual object is considered to be sinusoidally varying). Figures 3.18 and 3.19 present the results for this case.

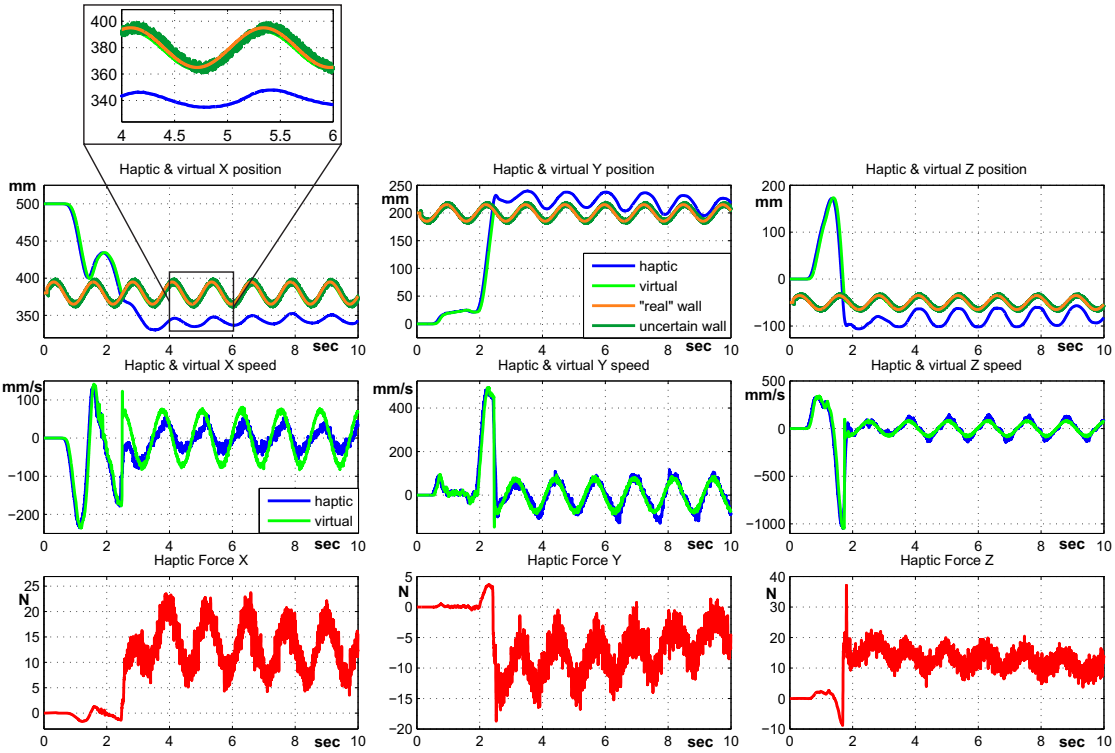


FIGURE 3.18: Restricted motion under constant uncertainty ($\tau_S = 120$ *ms*) and moving virtual object and uncertain feedback.

For both uncertainties, the impact moment is correctly provided, similarly to the previous cases. As expected, in the steady state position (as Figures 3.18-3.19 show on all three axes) the sinusoidal moves of the virtual walls appear to be consistent with the reality, since the oscillations of the force are correlated with the virtual wall position. The difference between this scenario and the previous one comes with the thickness of the force. The large thickness of the force (Figures 3.18-3.19, forces on all three axes) illustrates the vibration induced by the perturbation applied on the feedback. The vibrations are similar to the ones encountered in the *ideal* case. The human operator may feel these vibrations, but since there are quite *low* ($< \pm 5$ *mm*), the effect is not disturbing. For larger perturbations ($> \pm 10$ *mm*), the vibrations may increase substantially resulting in an unpleasant manipulation.

For the last examples of this subsection, a *virtual* box - Figure 3.4.d, will be used.

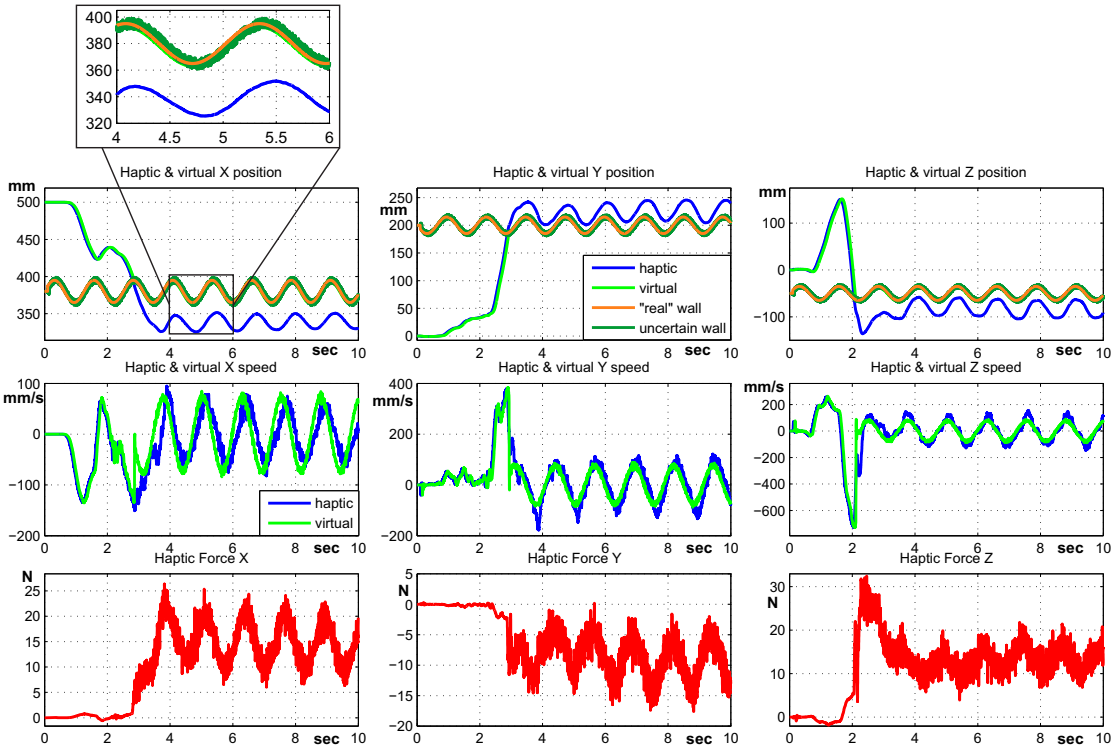


FIGURE 3.19: Restricted motion under constant uncertainty ($\tau_S = 80$ *ms*), moving virtual object and uncertain feedback.

Scenario 5. Figures 3.20 and 3.21 present the results for this scenario for the *random* moves with and without contacts inside the *virtual box*.

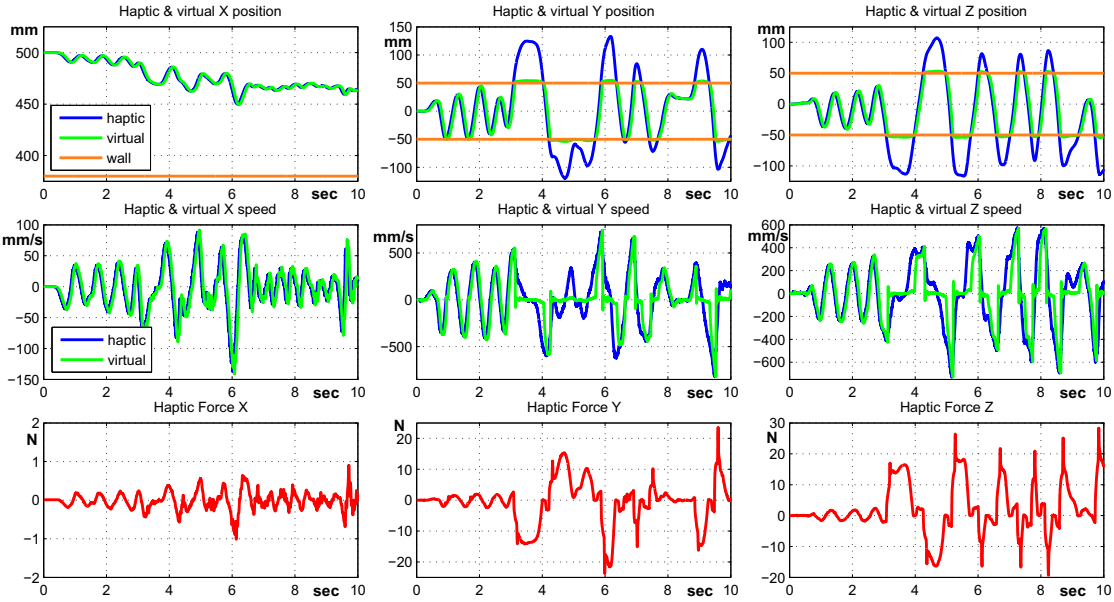


FIGURE 3.20: *Random* moves inside a virtual box under constant uncertainty ($\tau_S = 120$ *ms*).

From these two examples, it is important to notice that the Smith predictor is working correctly

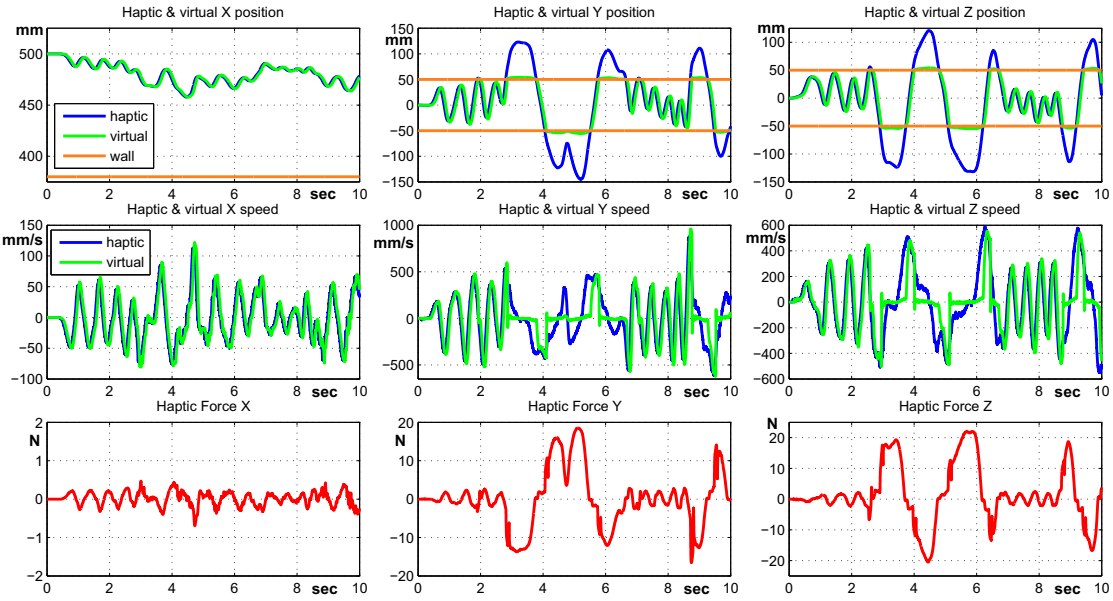


FIGURE 3.21: *Random* moves inside a virtual box under constant uncertainty ($\tau_S = 80$ *ms*).

under fast changes, and the single/multi-point contacts are provided in a natural way. The contacts can be seen in both Figures 3.20-3.21 on Y and Z axes every the the virtual wall limits are exceeded. The end user is able to feel correctly that he is *working* inside a box.

Scenario 6. In this example, a closed varying virtual environment will be used, i.e. the walls of the virtual box will sinusoidally move. Figures 3.22 and 3.23 illustrate this case.

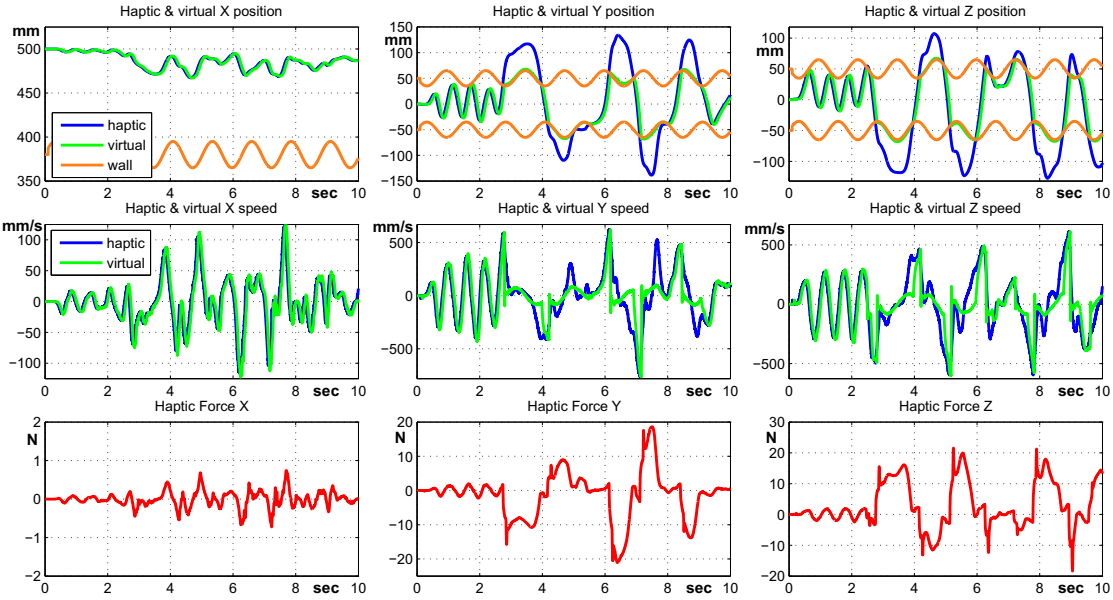


FIGURE 3.22: *Random* moves inside a virtual box under constant uncertainty ($\tau_S = 120$ *ms*) and moving virtual object.

The main goal of this example is to show that the proposed method is able to handle fast changes from free to restricted motion or viceversa, under a virtual scene where the objects' positions

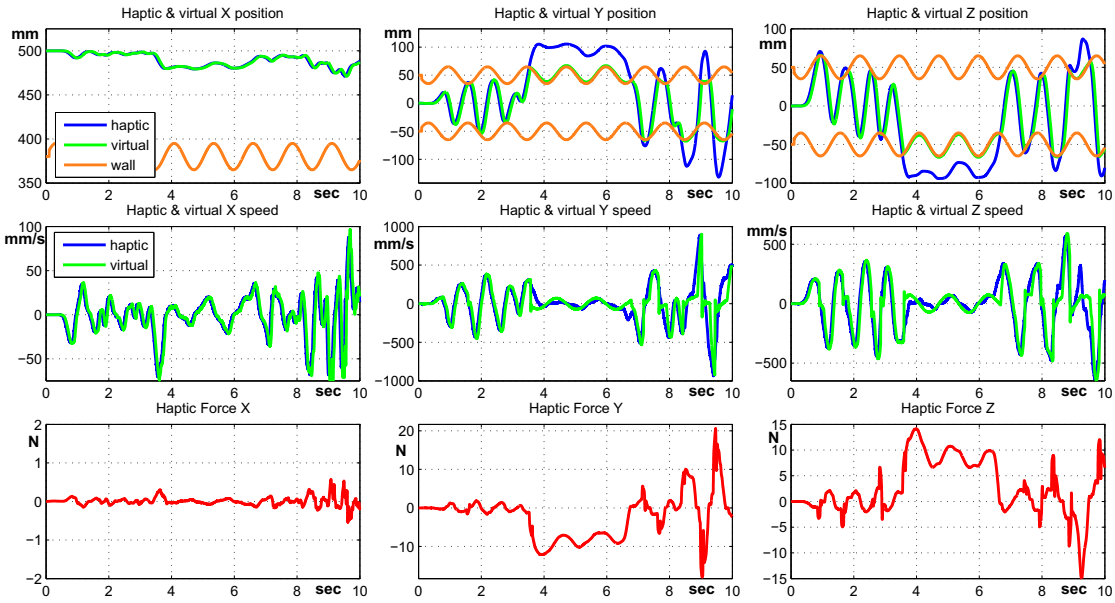


FIGURE 3.23: *Random* moves inside a virtual box under constant uncertainty ($\tau_S = 80$ *ms*) and moving virtual object.

are varying. This case is interesting since in a *real world* environment the objects' positions may vary. The obtained results are consistent with the reality, the system provides the human operator with a realistic interaction, and the moves of the virtual objects from the scene are correctly felt at the haptic end. Also in case of maintaining the force, Figure 3.22 Y axis from ≈ 2.5 to ≈ 4 *ms*, and Figure 3.23 Y and Z axes from ≈ 4 to ≈ 6.5 *ms* the user is able to feel the sinusoidal moves of the virtual walls, since the force is *tracking* the position.

Scenario 7. This last example introduces the perturbed case of the virtual box with sinusoidal *moving* walls. The results are illustrated in Figures 3.24 and 3.25.

For both uncertainties, the impact moments are correctly provided, similarly to the previous cases. Compared to the previous example, in the steady state position (Figure 3.24, Y and Z axes from ≈ 4 to ≈ 7.5 *ms* and Figure 3.25, Y and Z axes from ≈ 3.5 to ≈ 6.5 *ms*) the haptic interface is providing a vibrating response, due to the perturbations applied to the wall position (similar to the *ideal* case). Resuming, the end user is able to feel correctly the contacts (single or multi point), as well as that he is working inside a box, the only inconvenient comes in long time (> 0.5 *sec*) steady states when the vibrating response is revealed.

In this subsection, the results corresponding to fixed uncertainties were presented. The overall performances are consistent with the reality, and the impact moments are correctly provided in all situations and for both uncertainties. It is worth mentioning that in free motion, when the delay used in the Smith predictor is larger than the system's one (first case in all scenarios), there appears a phase advance resulting in an effect of mass and friction compensator.

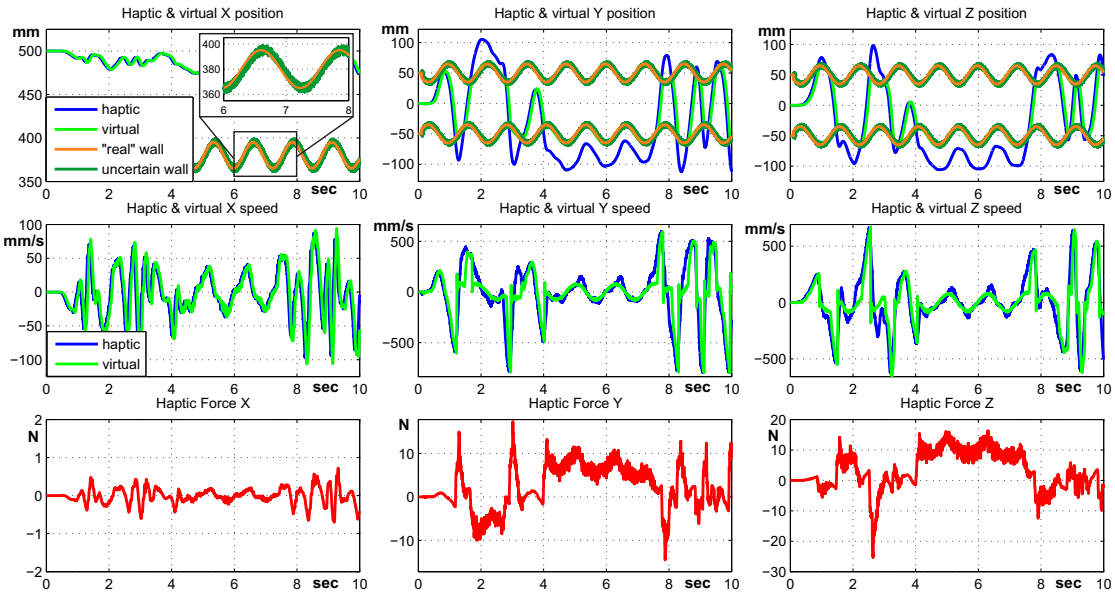


FIGURE 3.24: *Random* moves inside a virtual box under constant uncertainty ($\tau_S = 120\ ms$), moving virtual object and uncertain feedback.

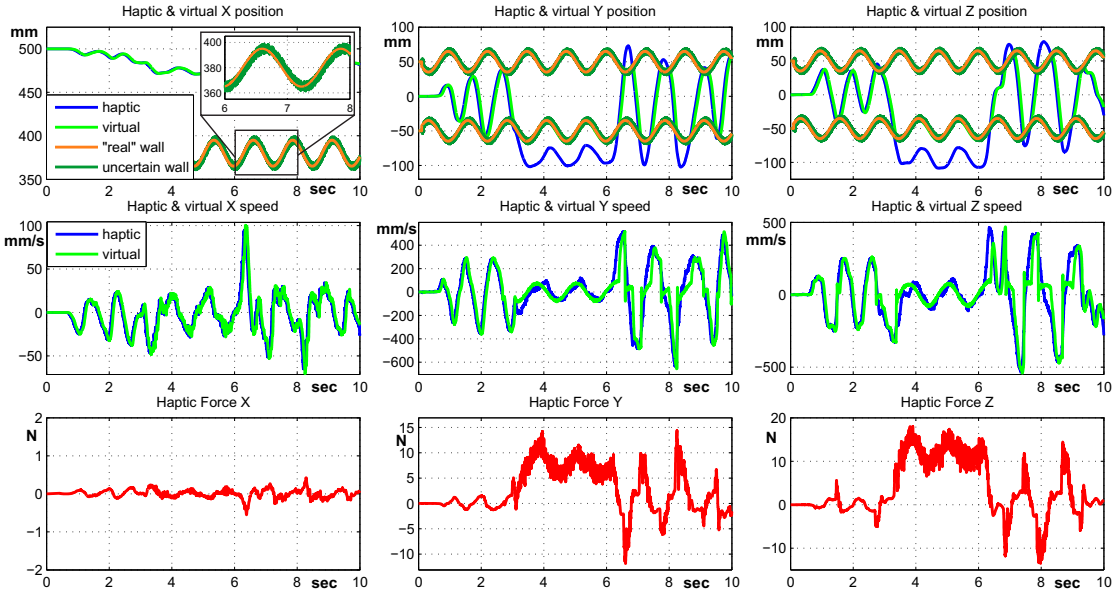


FIGURE 3.25: *Random* moves inside a virtual box under constant uncertainty ($\tau_S = 80\ ms$), moving virtual object and uncertain feedback.

3.4.3 Uncertain time-delays: uniform distribution

Starting from the theory presented in Subsection 2.4.6.2, the results corresponding to an uniform distributed time-delay for different scenarios are presented below. In the next examples, the backward and forward delays of the system will be considered to have the variation shape presented in Figure 3.26, corresponding to uniform distribution. The delay used in the Smith predictor will be considered to be constant.

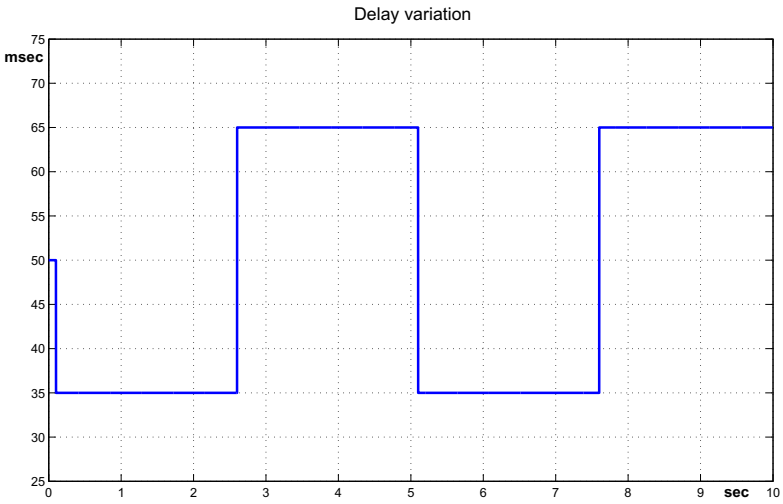


FIGURE 3.26: Delay variation shape corresponding to uniform distribution.

The following controller's gains are used for all the experiments presented in this subsection, corresponding to the uniform distributed delays case:

$$K_p = 1000\ N/m, \qquad K_d = 15\ Ns/m,$$

and the time-delay delay used in the Smith predictor $\tau_S = 100\ ms$.

These gains were chosen in order to obtain the *closest* performance to the *ideal* case, while respecting the stability region from Figure 2.19.

Scenario 1. In Figure 3.27 the free motion scenarios is presented for uniform distributed time-delays.

It can be noticed that, as in the previous subsection, when the delay used in the Smith predictor is larger than the system's delay, there appears a phase advance, resulting in mass and friction compensation. This phenomenon can be noticed in Figure 3.27 corresponding to the interval $\tau_1 = \tau_2 = 35\ ms$, when the force and velocity are acting in the same sense, while for the other interval $\tau_1 = \tau_2 = 65\ ms$ the force and velocity sense are opposite. The overall performances are respecting the desired requirements, i.e. the viscosity effect is low ($< 5\ N$, except some peaks resulting from sense changes at high velocities) and the system is pleasant to manipulate.

Scenario 2. Next, Figure 3.28 presents the *simple* restricted motion case, when the time-delays are uniformly distributed.

The impact moment is provided in a realistic way, since the force increases fast without creating any preliminary damping. It worth noting that the impact sensation is *less* stiffer ($\approx 60\%$ from the ideal case) due to the small values of the gains (especially K_p). An interesting test proposed in this scenario was the transition from one delay to another ($\tau_1 = \tau_2 = 35\ ms$ to $\tau_1 = \tau_2 = 65\ ms$ and viceversa) around the impact moment which is *a quite delicate* situation (see for example

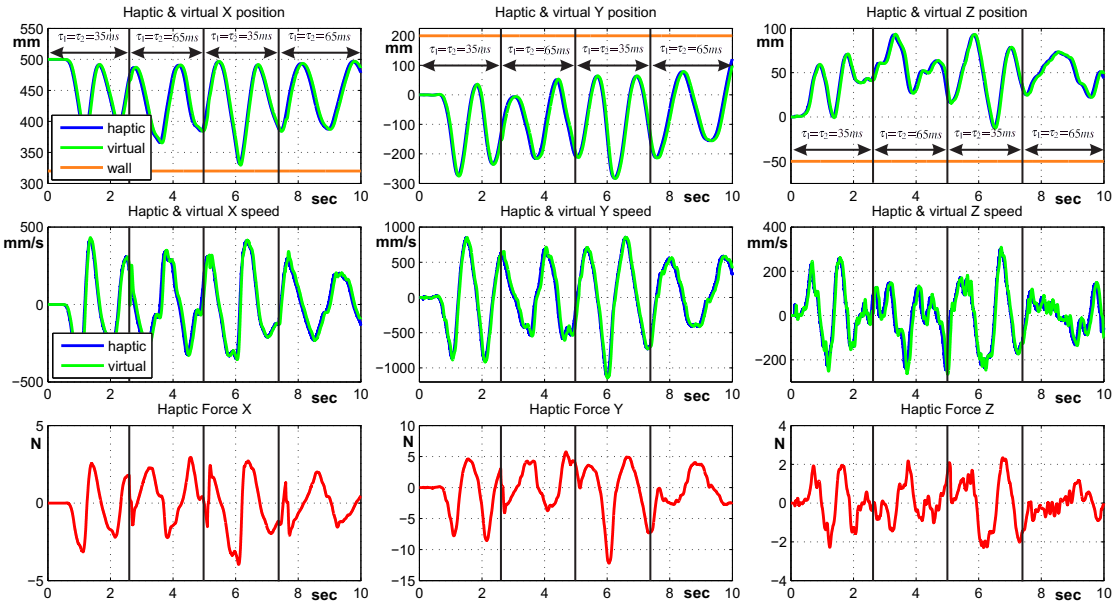


FIGURE 3.27: Free motion under uniform distributed time-delays.

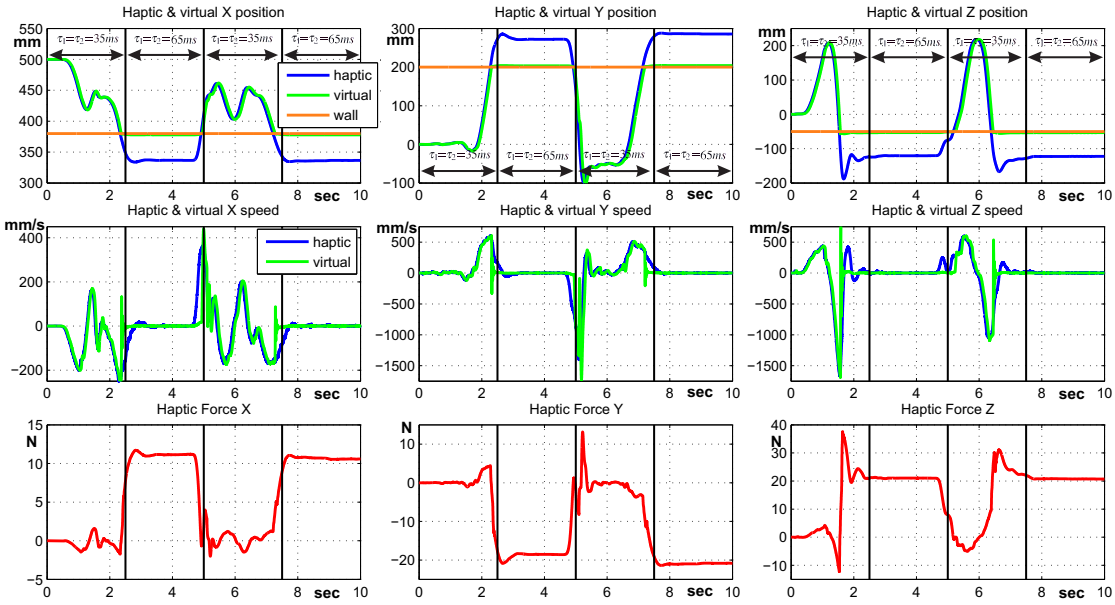


FIGURE 3.28: Restricted motion under uniform distributed time-delays.

Figure 3.28, *X* and *Y* axes around 2.5 *sec*), as well as, while the force was maintained (see for example Figure 3.28, *Z* axis from ≈ 6.5 to the end). As it can be seen from the graphic, the system behavior is appears to be consistent with the reality, providing a realistic manipulation.

Scenario 3. In the sequel, a *moving* object will be considered in order to test the system's capabilities of transmitting to the end user the moves of the objects from the virtual scenes. Figure 3.29 presents the results.

The goal of this scenario is to test if the moves of the virtual object can be correctly felt by the

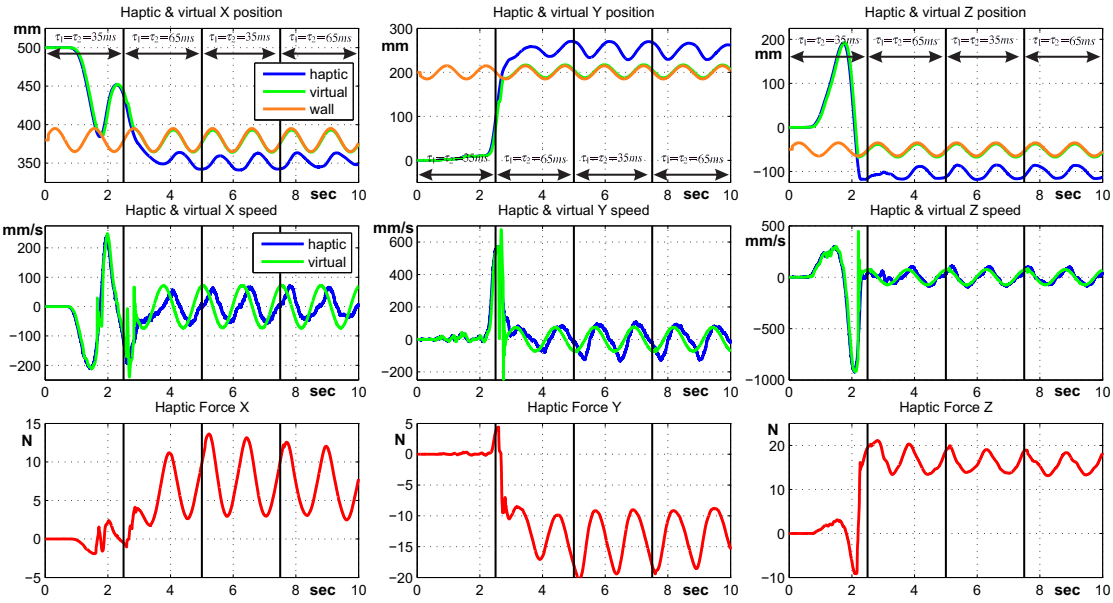


FIGURE 3.29: Restricted motion under uniform distributed time-delays and *moving* object.

human operator and the haptic end. Thus, while maintaining the force, as Figure 3.29 shows from ≈ 2 *sec* to the end, the human operator is able to feel the moves of the virtual wall. This phenomenon is also revealed by the synchronized oscillations of the force with the variation of the wall position (visible on all three axes in Figure 3.29). Furthermore, on Y axis, the transition from free to restricted motion was realized while passing from one delay to another ($\tau_1 = \tau_2 = 35$ *ms* to $\tau_1 = \tau_2 = 65$ *ms*) in a completely transparent way to the user. Also, while the force is maintained (starting from ≈ 2 *sec* to the end on all axes) the time-delay changes are completely imperceptible by the end user. Finally, the impact moment is reproduced similarly to the previous case without any additional phenomenons.

Scenario 4. In this example, while maintaining the same conditions from the previous scenario, a perturbation will be added on the position feedback (used in the Smith predictor) in order to test the vulnerability in case of uncertainties. Figure 3.30 presents the results.

This example points out the influence of a possible perturbation on the distance feedback used in the Smith predictor. Compared to the previous example where there were no perturbations, the difference can be seen while the force is maintained by the large thickness of the force denoting the presence of the vibrations in the haptic interface (see Figure 3.30, Y axis from ≈ 2.5 to ≈ 5 *ms*, and Z axis from ≈ 2 to ≈ 5 *ms* and ≈ 8 *ms* to the end). As already mentioned, these vibrations are felt by the human operator without a disturbing effect. Resuming, similar to the previous cases, the impact moment is correctly provided and also the object's moves are felt by the human operator. The presence of the perturbation induces a vibration, that can be slightly felt the human operator. For more important perturbations ($> \pm 10$ *mm*) the vibration will also increase, providing an unpleasant manipulation.

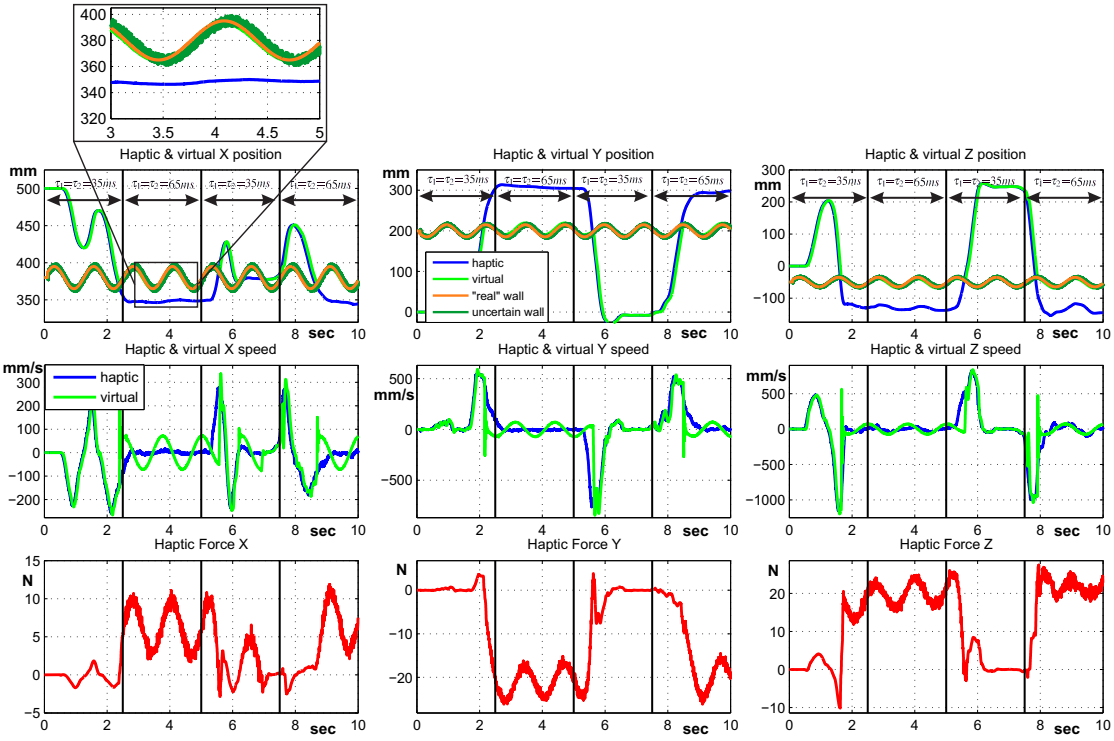


FIGURE 3.30: Restricted motion under uniform distributed time-delays *moving* object and uncertain feedback.

In the next examples of this subsection, a more complex virtual scene will be considered, i.e. a *virtual* box, as it was presented in Figure 3.4.d, will be used.

Scenario 5. Figure 3.31 exposes the random moves inside the “virtual” box.

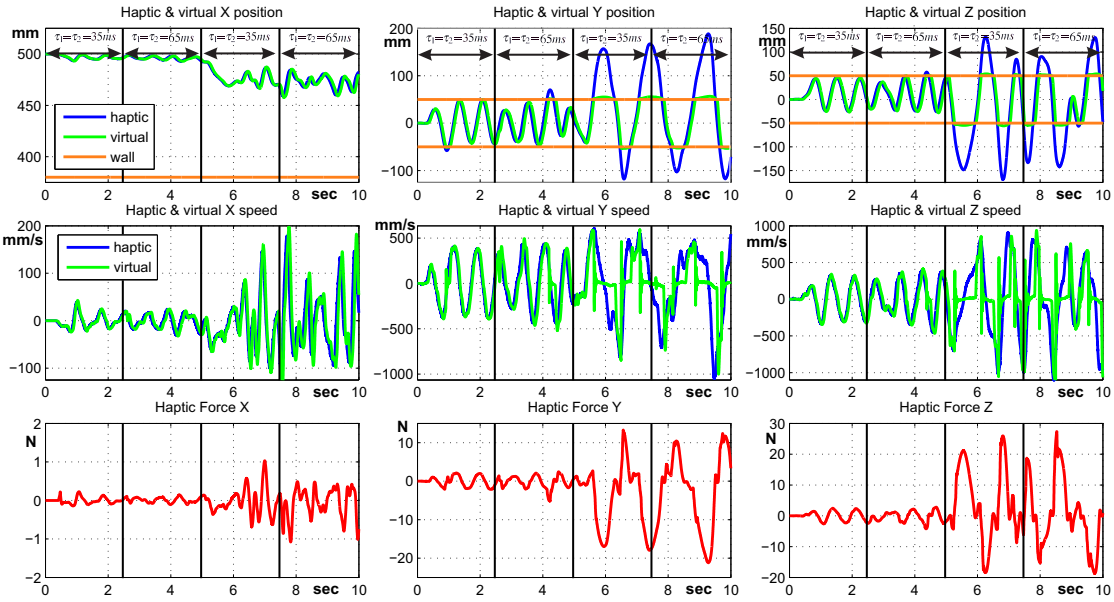


FIGURE 3.31: *Random* moves inside a virtual box under uniform distributed time-delays.

The main objective of this experiment is to point out the method capabilities under *fast* changes from free to restricted motion or viceversa and uncertain time-delays. More precisely, in free motion the viscosity effect is low (<4 *N*), depicted in Figure 3.31 when there are no virtual wall limits exceeds. On the other hand, any contact situations are correctly felt by the human operator through the haptic interface, illustrated in Figure 3.31 wherever there is any virtual wall limit exceed. Concluding, the results are consistent with the reality, the human operator can feel accurately all the impacts (single and multi-point) in a realistic manner.

Scenario 6. This example will point out the case when more virtual objects from the scene are moving. Figure 3.32 illustrates this case for a virtual box with sinusoidally moving walls under uniform distributed delays.

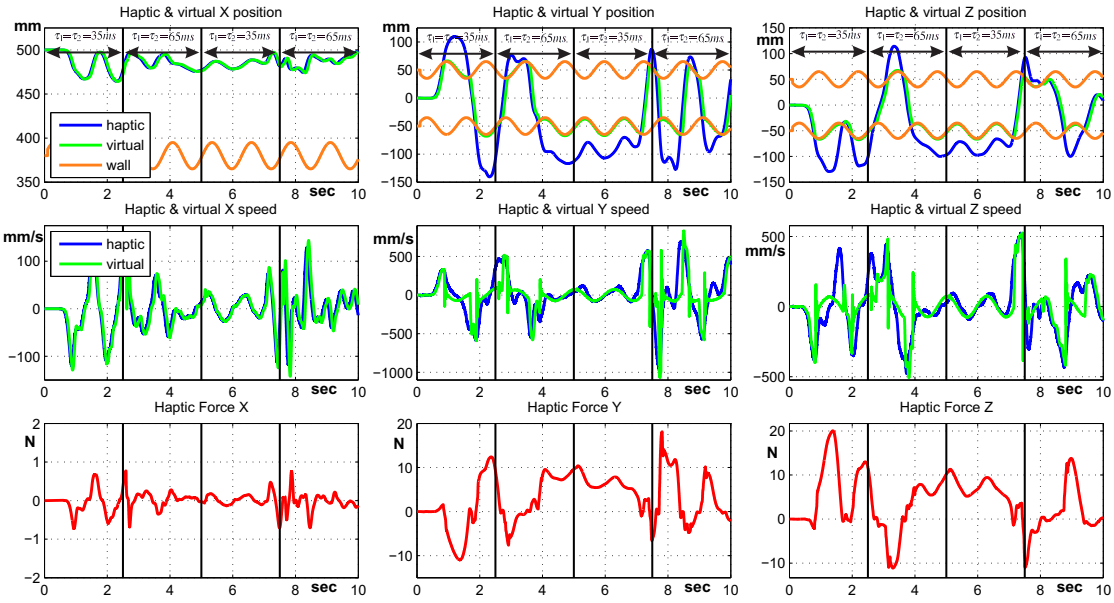


FIGURE 3.32: *Random* moves inside a virtual box under uniform distributed time-delays and *moving* walls.

Compared to the previous scenario, here the main goal is to the test the system capabilities to reproduce the virtual objects moves through the haptic interface under fast switches from free to restricted motion or viceversa. The first objective was to provide a realistic impact with moving objects illustrated by any limit exceed of the virtual wall. The second objective was to reconstruct the object moves while the force is maintained depicted in Figure 3.32 on Y and Z axis from ≈ 4 to 7.5 *sec*. Similar to the previous case, the obtained results are respecting the desired performances in free and restricted motion, and furthermore, if the force is maintained, the user is able to feel the moves of the virtual objects, i.e. in this case, the sinusoidal moves on Y and Z axes.

Scenario 7. The last example of this subsection introduces the perturbed feedback case - Figure 3.33.

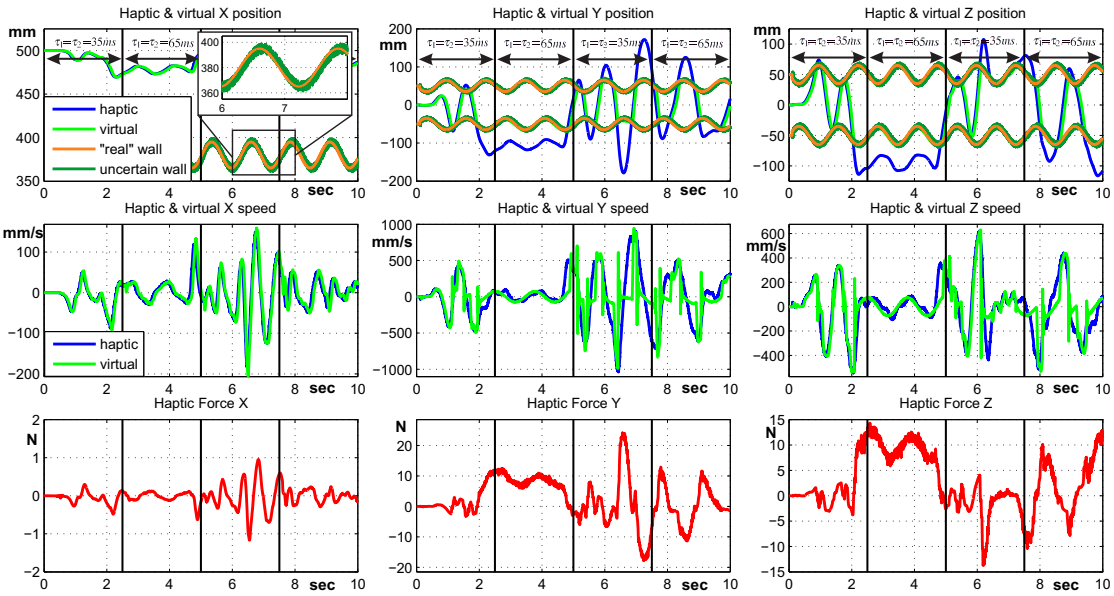


FIGURE 3.33: *Random* moves inside a virtual box under uniform distributed time-delays, *moving* walls and uncertain feedback.

As expected, the impact feeling is correctly provided and the switching between the two motions is guaranteed. The only inconsistency of this case is the vibrating response of the haptic interface when the force is maintained, illustrated by the large thickness of the force (see Figure 3.33, Y and Z axes from ≈ 2 to ≈ 5 *sec*). It is worth mentioning that in this case the vibrations are not very disturbing, but for more important ones (± 10 *mm*) these vibrations may become annoying. In real life circumstances, such situations do not appear as being necessarily realistic.

The minimum and maximum delays considered in this example are the equivalent of a $\pm 30\%$ delay variation. It is worth mentioning that an increase of the limits will penalize more the stability zone by reducing the maximum value of the K_p gain. As mentioned previously the impact feeling is correctly provided, i.e. the force is increasing fast, but less stiffer ($\approx 60\%$ from the ideal case) due to the important decrease of the K_p gain (1100 compared to 1800 in the ideal case). Also, a small value of K_p gain will result in a more important steady state error in restricted motion. As a general conclusion, this distribution offers the best solution in terms of controllers gains when the only information available regarding the delay are the minimum and the maximum limits, without any constrains on shape.

3.4.4 Uncertain time-delays: gamma distribution with gap

Based on the theory presented in Subsection 2.4.6.3, this subsection presents the examples corresponding to a gamma with gap distributed time-delay for different scenarios. The delay shape corresponding to this distribution was presented in Figure 2.10. Similar to the previous subsection, the delay used in the Smith predictor will be considered to be constant.

The following controller's gains will used for all the experiments presented in this subsection, corresponding to the gamma distribution with gap case:

$$K_p = 1100\ N/m, \qquad K_d = 55\ Ns/m,$$

and the time-delay used in the Smith predictor $\tau_S = 100\ ms$.

Similar to the previous cases, these gains were chosen in order to obtain the *closest* performance to the *ideal* case, while respecting the stability region from Figure 2.12.

Scenario 1. Figure 3.34 presents the free motion case under gamma distributed delays with gap.

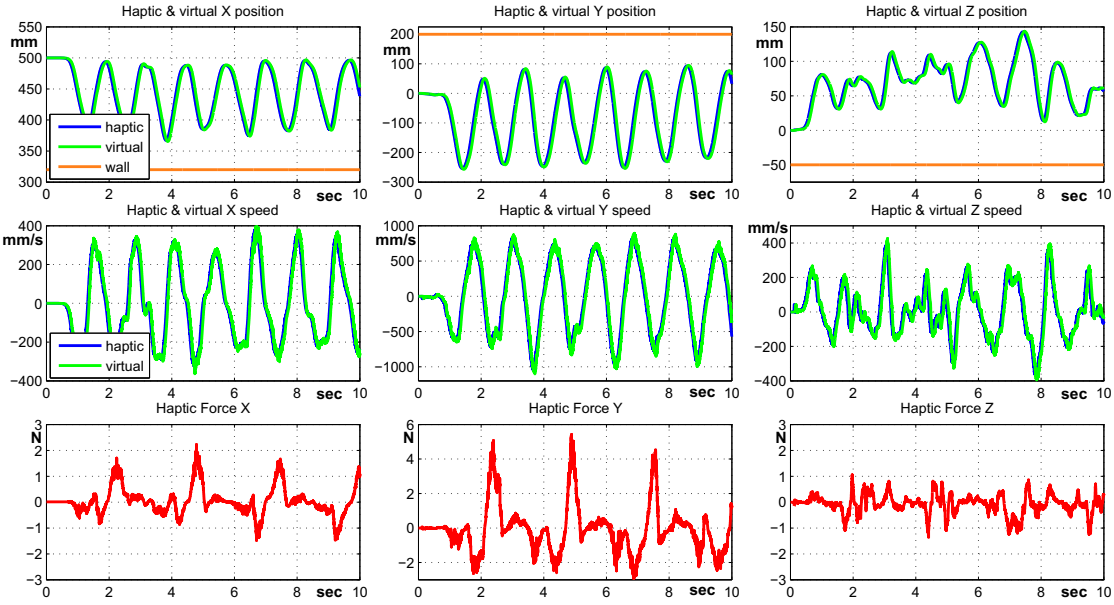


FIGURE 3.34: Free motion under gamma with gap distributed time-delays.

As expected, the overall performances are consistent with the reality, i.e. the viscosity effect is low ($< 5\ N$, except some peaks resulting from sense changes at high velocities) and the system is pleasant to manipulate.

Scenario 2. Next, Figure 3.35 presents the restricted motion case, when the time-delays are gamma with gap distributed.

The impact moment can be accurately felt due to the fast increase of the force, and in a "stiffer" way compared to the uniform distribution (70% compared to 60% from the ideal case). Two impacts are made on each axis in order to cover all the delay variations. The obtained results confirm the methods capabilities of guaranteeing the performance under uncertain time delays, and transparent to the user.

Scenario 3. Next, Figure 3.36 presents the results in the case of hard contacts with *moving* objects.

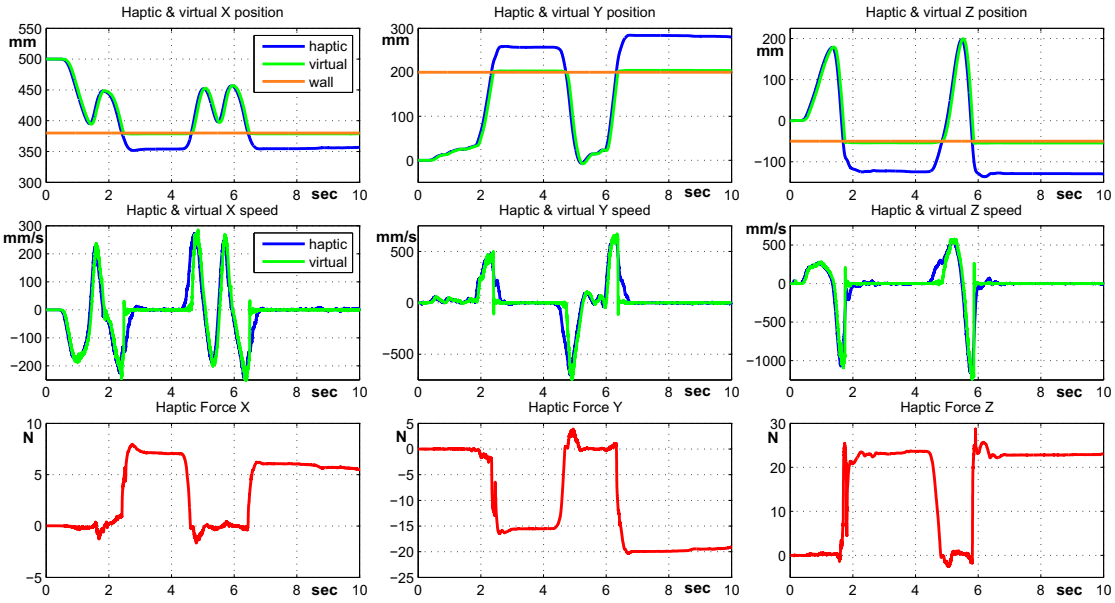


FIGURE 3.35: Restricted motion under gamma with gap distributed time-delays.

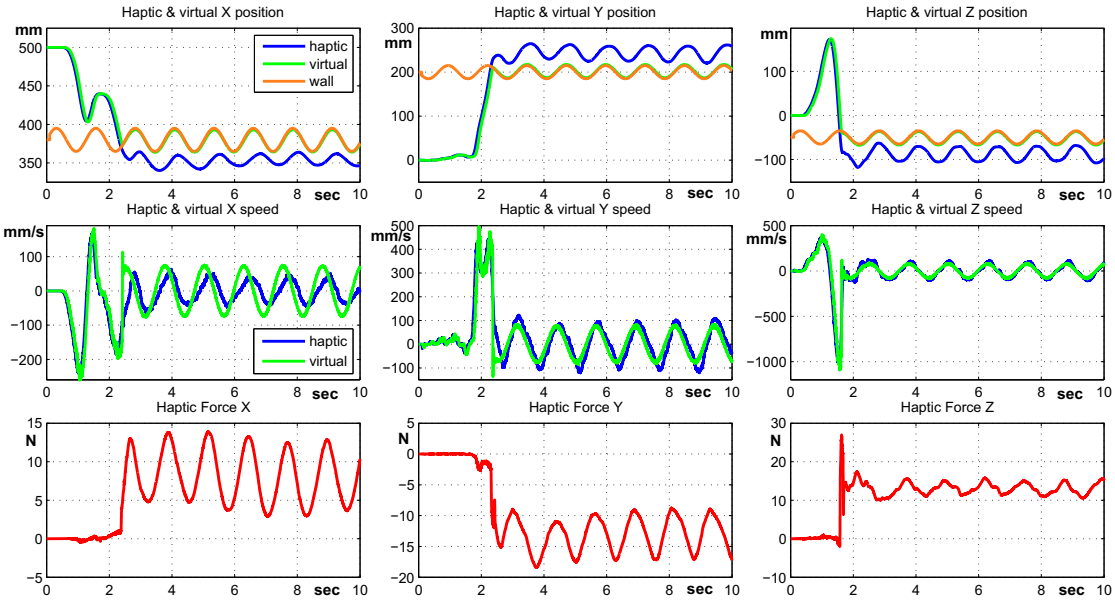


FIGURE 3.36: Restricted motion under gamma with gap distributed time-delays and *moving* objects.

As mentioned before, this scenario points out the hard contacts with moving objects. The impact moment is reproduced similarly to the previous case, and furthermore, while the force is maintained (on all axes starting from $2-2.5$ *sec* to the end), the system is able to *transmit* to the end user the moves of the objects from the virtual scenes. The moves can be visualized by the synchronized oscillations of the force with the varying position of the virtual wall.

Scenario 4. In this example, the position feedback (used in the Smith predictor) will be perturbed in order to test the vulnerability in case of uncertainties. Figure 3.37 presents the results.

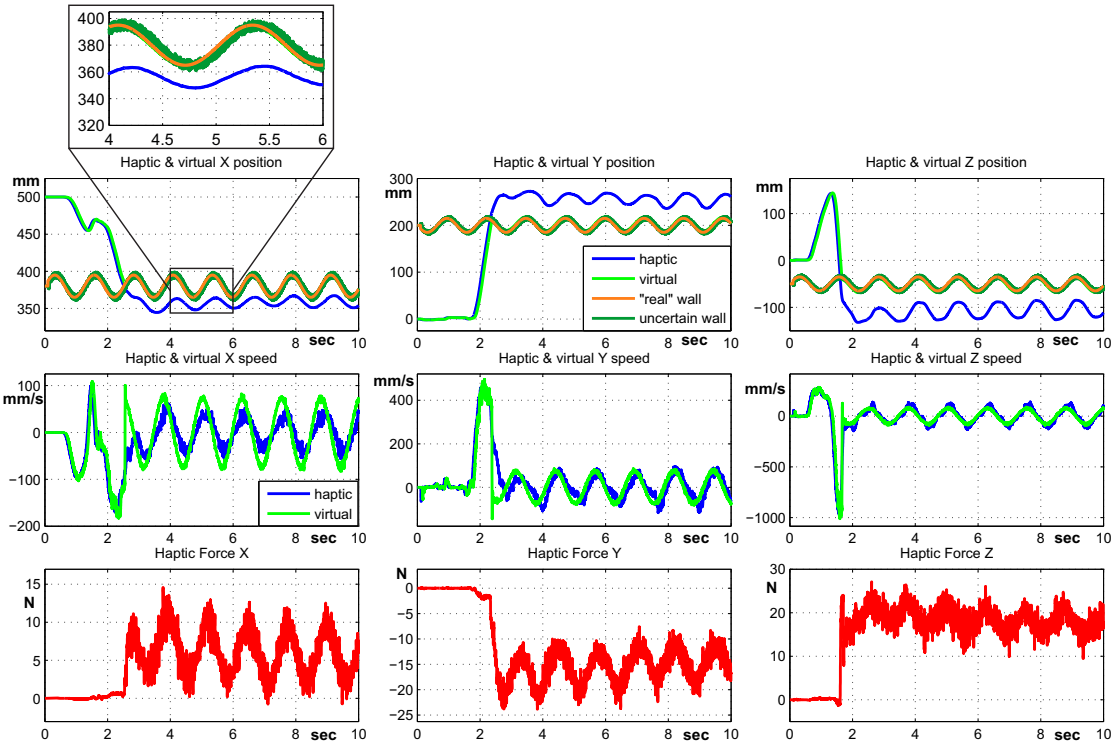


FIGURE 3.37: Restricted motion under gamma with gap distributed time-delays *moving* object and uncertain feedback.

As expected, similar to the previous cases, the impact moment is correctly provided and also the object's moves are "correctly" felt by the human operator. The presence of the perturbations induce a vibration in the haptic interface (visible on the graphic due to the large thickness of the force on all three axes), which can be slightly felt the human operator. For larger perturbations ($> \pm 10$ *mm*) the vibrations will also increase, providing an unpleasant manipulation.

In the last examples of this subsection, a *virtual* box - Figure 3.4.d, will be used, in order to test the method for more complex virtual scenes.

Scenario 5. Figure 3.38 shows the random moves inside the *virtual* box under gamma distributed time-delays with gap.

As mentioned previously, the purpose of this experiment is to demonstrate the Smith-predictor-based solution under *fast* changes between free and restricted motions. According to Figure 3.38, the viscosity effect in free motion (illustrated by the non-exceeded virtual wall limits) is low (< 5 *N*), while in restricted motion (illustrated by the exceeded virtual wall limits) the impacts are provided in a stiff way. The results are consistent, i.e. all the impacts (single and multi-point) are accurately provided to the human operator, as well as a low viscosity effect is guaranteed.

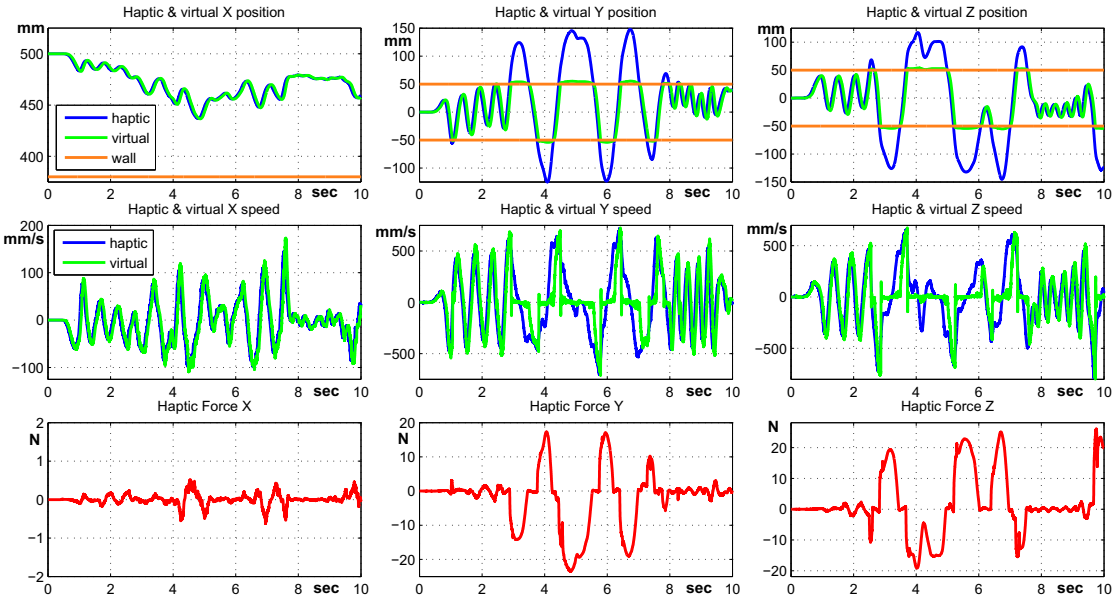


FIGURE 3.38: *Random* moves inside a virtual box under gamma with gap distributed time-delays.

Scenario 6. This example presents the case when more virtual objects from the scene are moving. Based on the same circumstances as the previous scenario and *adding* moving walls, Figure 3.39 illustrates this case.

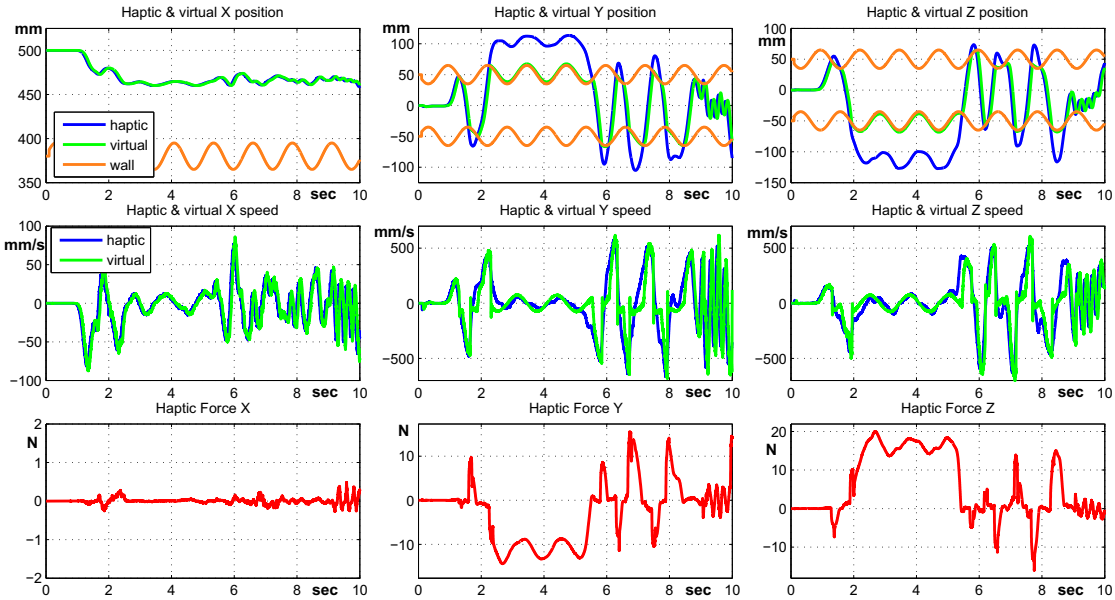


FIGURE 3.39: *Random* moves inside a virtual box under gamma with gap distributed time-delays and *moving* walls.

In this scenario the main purpose is to test the method capabilities of providing a realistic behavior under fast switches between the two motions (free and restricted), while the objects from the scenes have *moving* positions. From Figure 3.39, it can be seen that the impact is correctly

provided without any constraint on the position of the moving walls. Furthermore, if the force is maintained (see Figure 3.39m from ≈ 2 to 5.5 *sec*), the user is able to feel the moves of the virtual objects, i.e. in this case, the sinusoidal moves on Y and Z axes.

Scenario 7. In this last example of this subsection, Figure 3.40 introduces the perturbed feedback case of the virtual box with moving walls.

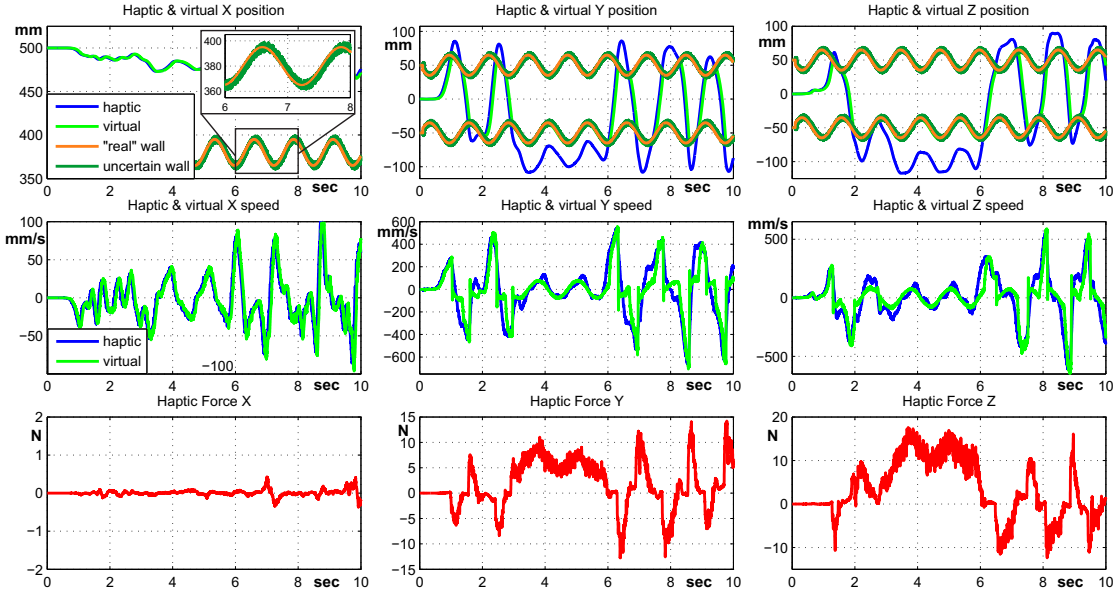


FIGURE 3.40: *Random* moves inside a virtual box under gamma with gap distributed time-delays, *moving* walls and uncertain feedback.

Compared to the results from the previous scenario, the main difference comes when the force is maintained, see Figure 3.40, axes Y and Z from ≈ 3 to 6 *sec*. More precisely the increased thickness of the force denotes the vibrating behavior of the haptic interface. Since the perturbations were considered to be ± 5 *mm*, the effect felt by the human operator is acceptable, but as previously mentioned for more important perturbations ($> \pm 10$ *mm*), the vibrations increase resulting in an unpleasant behavior. On the other hand, the impact feeling is correctly provided without any constraints regarding the virtual object position, and the switching between the two motions is guaranteed. Similar to the previous cases of this scenario, the only inconvenient is the vibrating response of the haptic interface when the force is maintained.

In this subsection the experimental results under gamma with gap distributed delays were presented. As a general conclusion, the obtained results confirm the method capabilities for this case, since the desired performances are acquired. Studies on the problem of controlling objects over communication networks [157] pointed out that, in this case, the time-delays can be modeled by a gamma distribution with gap (see [121] for further details).

3.4.5 Uncertain time-delays: normal distribution

According to the theory presented in Subsection 2.4.6.4, this subsection presents the examples corresponding to normal (Gaussian) distributed time-delays for different scenarios. The delay shape corresponding to this distribution was presented in Figure 2.21. Like in the previous two subsections, the delay used in the Smith predictor will be chosen to be constant.

Since from the gains' tuning point of view, this case comes to the *ideal* case, the controller's gains, used for the experiments presented in this subsection, were chosen to be equal to the ones used in the ideal case:

$$K_p = 1800\ N/m, \qquad K_d = 80\ Ns/m,$$

and the time-delay delay used in the Smith predictor $\tau_S = 100\ ms$.

Scenario 1. In Figure 3.41 the free motion case is presented.

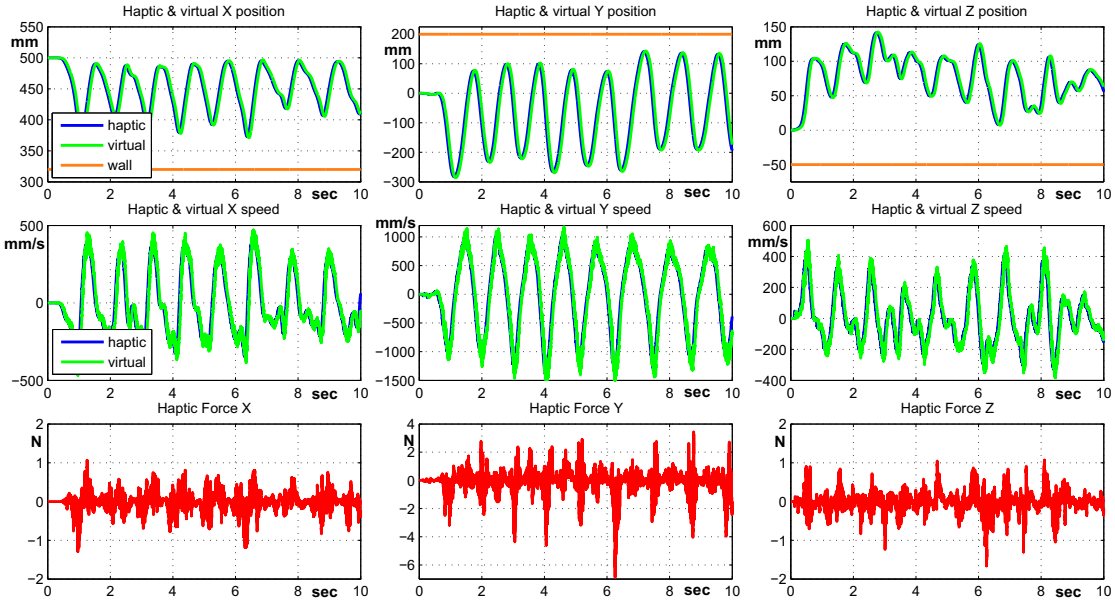


FIGURE 3.41: Free motion under normal distributed time-delays.

As expected, the results are *slightly* similar to the ideal case, i.e. the viscosity effect is $< 2\ N$ compared to the $< 0.5\ N$ in the *ideal* case (except some peaks when the direction is changed at high velocities), explained by the delay variations.

Scenario 2. Next, Figure 3.42 presents the restricted motion case, when the time-delays are normal distributed.

The impact moment can be accurately felt in a stiff way, close to the *ideal* case ($\approx 90\%$ from the *ideal* case). Since the gains are identical to the ones used in the ideal case, the only difference is made by the delay variations. From the end user point of view the differences between the *ideal* case and the normal distributed delays are basically non-distinguishable.

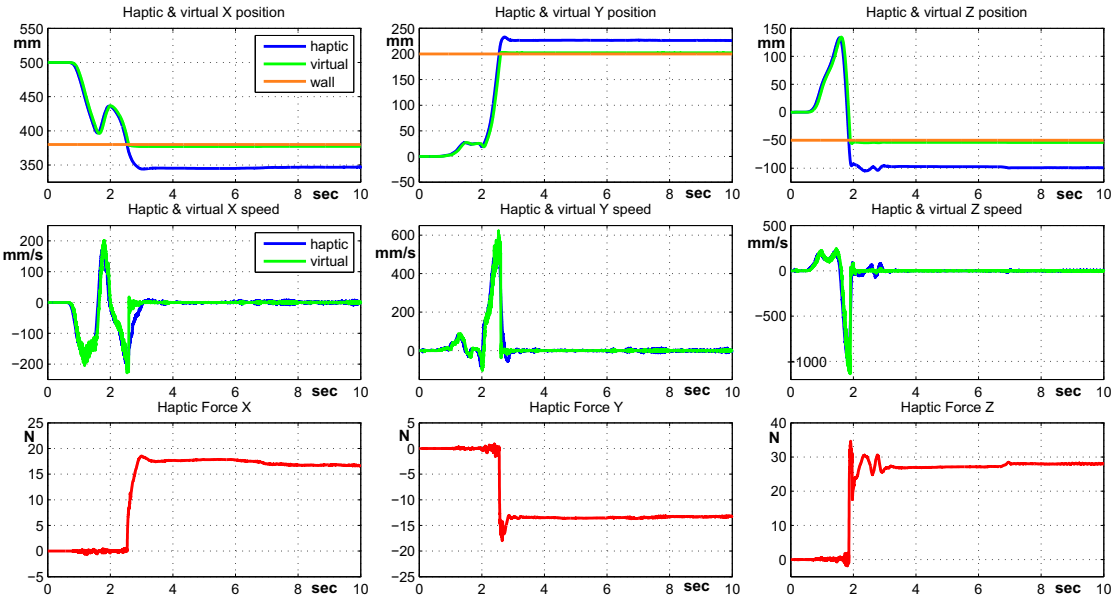


FIGURE 3.42: Restricted motion under normal distributed time-delays.

Scenario 3. In the sequel, Figure 3.43 presents the results in the case of contacts with *moving* objects.

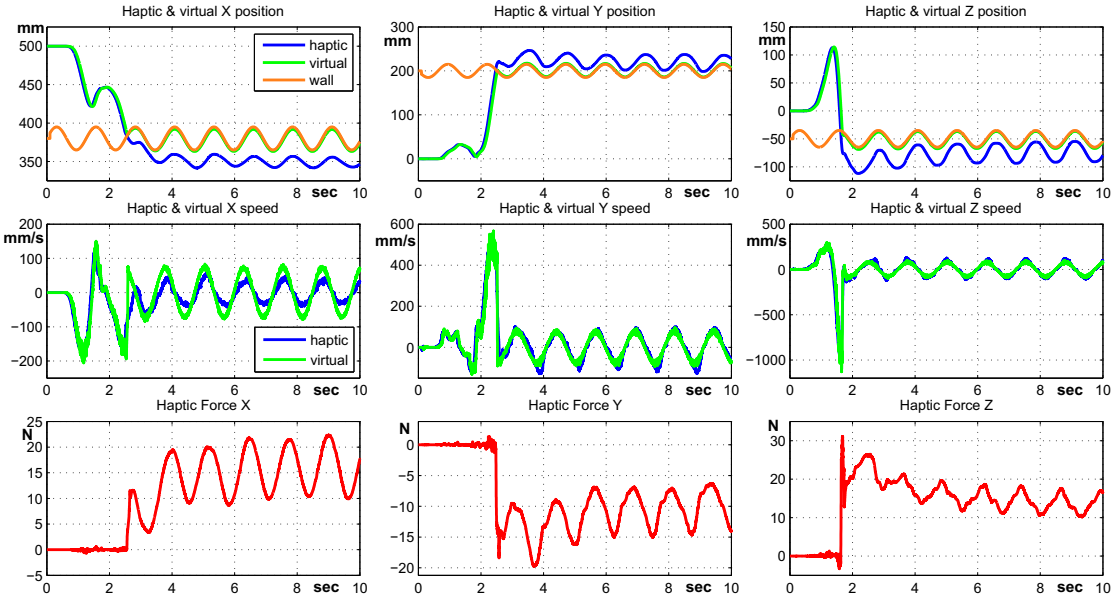


FIGURE 3.43: Restricted motion under normal distributed time-delays and *moving* objects.

In this scenario, the main objective is to test the method capacity of reproducing the virtual object move through the haptic interface. More precisely, when the force is maintained the force oscillations are following the exact moves of the virtual object, in this case the sinusoidal trajectory (see Figure 3.43, all axes starting from ≈ 2 to the end). Similar to the previous

scenario, the impact moment is reproduced correctly and in a stiff way, providing to the human operator a pleasant and realistic way of manipulation.

Scenario 4. Figure 3.44 presents the results when the position feedback (used in the Smith predictor) is perturbed.

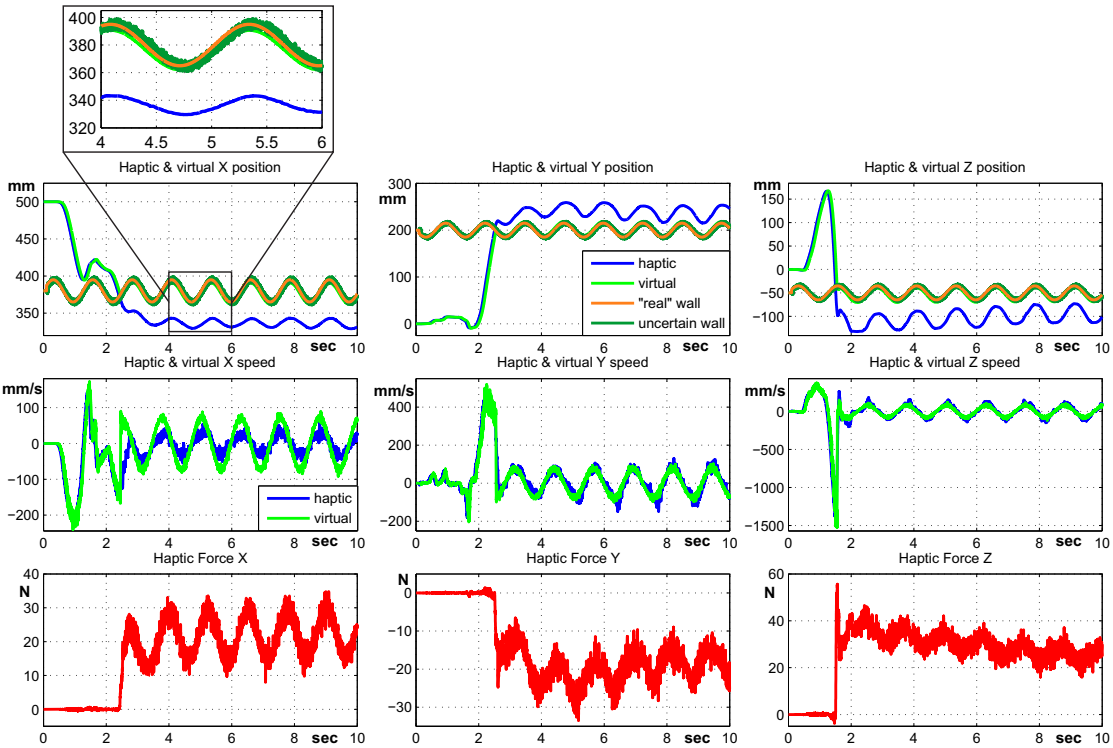


FIGURE 3.44: Restricted motion under normal distributed time-delays *moving* object and uncertain feedback.

In addition to the previous case, the presence of the perturbations results in a vibrating behavior of the haptic interface. As mentioned, this effect can be observed on the graphic due tot the large thickness of the force. Similar to the previous cases, the impact moment is correctly provided without any influence from the perturbation. The presence of the perturbation ($<$ ± 5 *mm*) induces a vibration, that can be slightly felt the human operator. For larger perturbations ($>$ ± 10 *mm*) the vibration will also increase, providing an unpleasant manipulation.

The last examples of this subsection will be carried out using a *virtual* box - Figure 3.4.d, in order to test the method for more complex virtual scenes.

Scenario 5. Figure 3.45 shows the random moves inside the *virtual* box under normal distributed time-delays.

As mentioned previously, the main objective of this experiment is to point out the capabilities of *switching* from free to restricted motion or viceversa, under *fast* changes conditions. The free motions (when there are no exceeds of the virtual wall limits), as well as the restricted motions (when the limits of the virtual wall are exceeded) are correctly provided and felt to/by the end

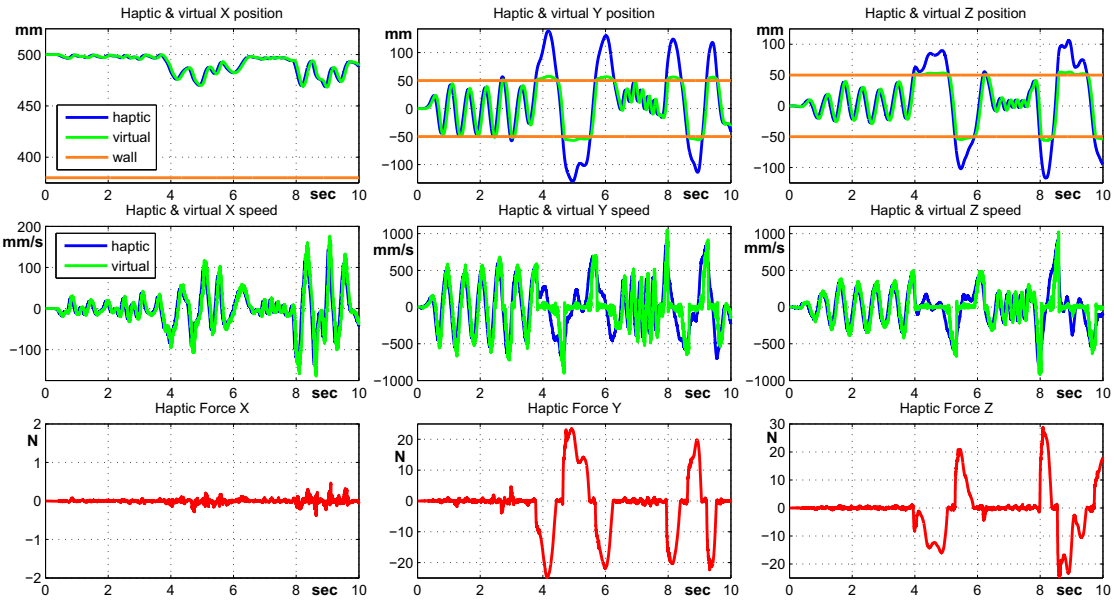


FIGURE 3.45: *Random* moves inside a virtual box under normal distributed time-delays.

user. The obtained results are *closed* to the *ideal* case, due to the identical gains used in the controllers.

Scenario 6. In this example, the case when more virtual objects from the scene are moving is presented. Figure 3.46 illustrates this case for a virtual box with sinusoidally moving walls.

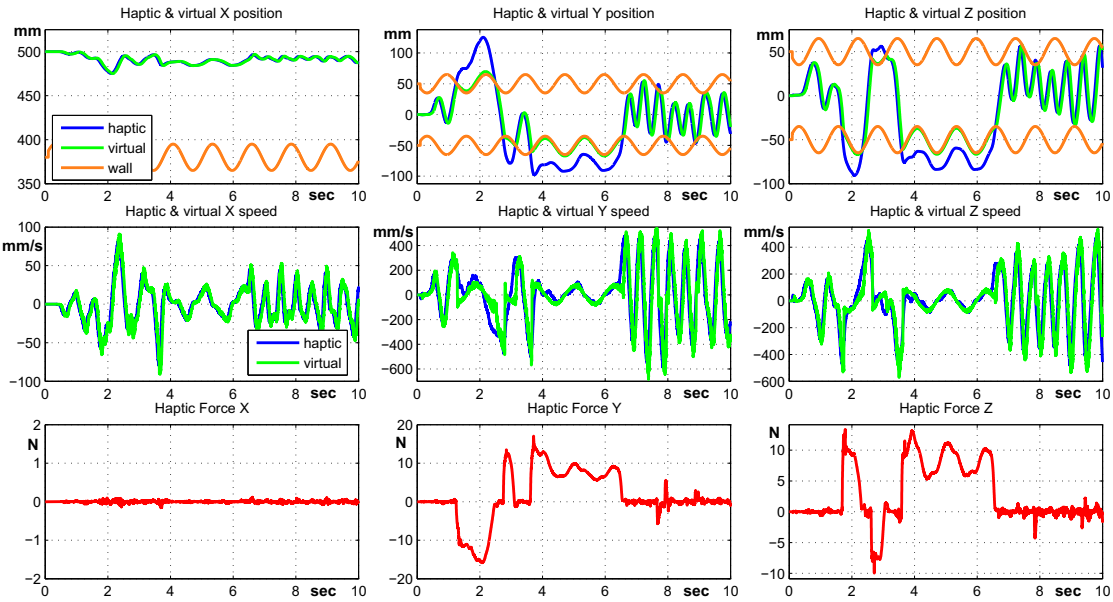


FIGURE 3.46: *Random* moves inside a virtual box under normal distributed time-delays and *moving* walls.

As already pointed out, this scenario highlight the method capabilities of acting correctly in dynamical complex environment under fast switches from free to restricted motion and viceversa. In addition to the previous example, the main purpose is to test the impacts with moving objects,

as well as the capability of transmitting to the end user the virtual object moves. More precisely, in Figure 3.46, axes Y and Z starting from ≈ 3.5 to 7 *sec*, the force is maintained, and it can be seen the that the force is tracking the virtual wall oscillations, and thus providing through to the human operator the moves of the virtual object on both Y and Z axes. The impacts are correctly provided as well as the transitions from free to restricted motion and viceversa.

Scenario 7. The last example - Figure 3.47 introduces the perturbed feedback case under normal distributed delays.

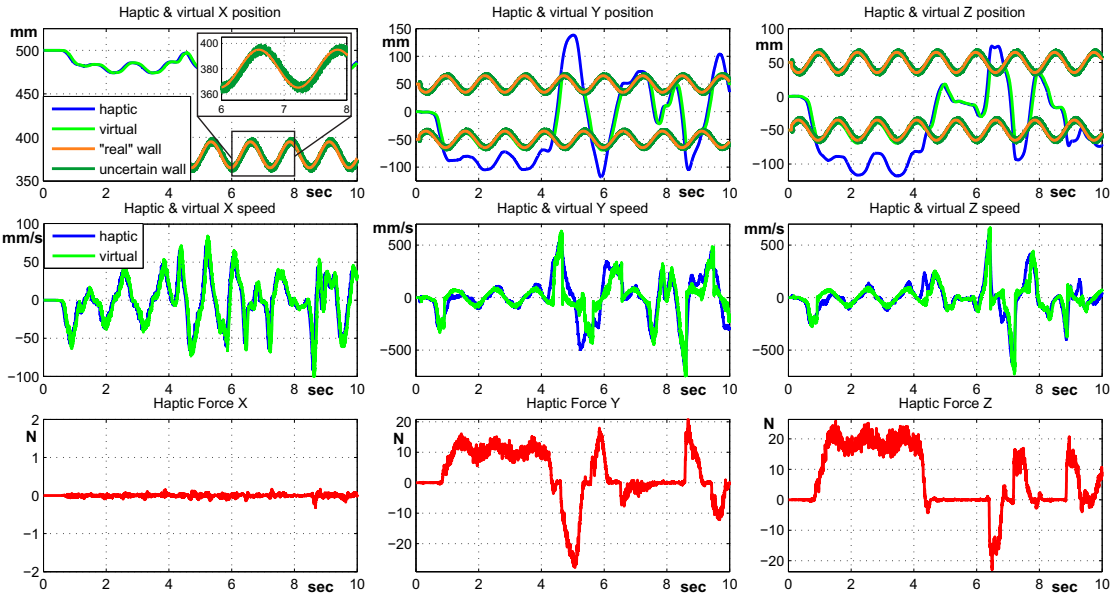


FIGURE 3.47: *Random* moves inside a virtual box under normal distributed time-delays, *moving* walls and uncertain feedback.

As the previous cases, the impact feeling and virtual object moves are correctly provided. Similar to the previous cases, the vibrating response of the haptic interface when the force is maintained is the only inconvenient of this case. The vibrations can be observed by the important thickness of the force, see Figure 3.47, axes Y and Z starting from ≈ 1 to 4*sec*. As mentioned for other cases, the effect felt by the human operator is acceptable, but for more important perturbations ($> \pm 10$ *mm*), the vibrations increase substantially, resulting in an unpleasant behavior.

In this subsection, the experiments for all the scenarios corresponding to the normal distribution were presented. It is worth mentioning that this distribution gave the *closest* results to the *ideal* case among all the others. This is because the gains used in the controller are perfectly identical to the ones used in the *ideal* case at further more the medium delay variation is *almost* fully compensated by the Smith predictor.

This subsection ends the experimental validation of the proposed method. In the next section, some discussions and concluding remarks will point out the main ideas of this chapter.

3.4.6 Discussions

In this section, a complete experimental validation was presented. Based on the scenarios presented at the beginning, several delay types were tested based on the theory presented in the previous chapter.

As already mentioned, the reference case. i.e. *ideal* case, is considered to be the constant delays one, when the use of the proposed method projects the problem in the free-of delay case. Table 3.1 resumes the time-delays considered for each circumstance, where the system delays corresponds to the forward/backward time-delays, while the total delays is represented by the sum of the two.

Time-delay circumstance	System time-delay (forward/backward)	Smith predictor time-delay
Fixed time-delays	50 *msec* (fixed)	2*x*50 *msec* (fixed)
Fixed uncertainty	50 *msec* (fixed)	$2x(50 \pm 10)$ *msec* ($\pm$20% uncertainty)
Uniform distribution	50 ± 15 *msec* ($\pm$30% variation)	2*x*50 *msec* (fixed)
Gamma with gap distribution	$[40, 70]$ *msec* ($\approx$ 25% variation)	2*x*50 *msec* (fixed)
Normal distribution	50 ± 15 *msec* ($\pm$30% variation)	2*x*50 *msec* (fixed)

TABLE 3.1: Time-delay values for each circumstances.

Analyzing now the performances of each scenario:

Free motion

Compared to the *ideal* case where the feedback force (viscosity effect) is almost *null*, ($< 0.5\ N$) at high velocities, closed performances are also obtained for the normal and gamma with gap distributed delays, i.e. $< 2\ N$ and $< 4\ N$ respectively, at high velocities. For the other types of delay (uniform distribution and fixed uncertainties)the performances are still consistent with the reality, and the system remains agreeable for manipulation - $< 6\ N$ at high velocities.

An *interesting* phenomenon appears when the delay used in the Smith predictor is larger than the overall system's delay. The predictor is not only compensating the delays' effects, but also the mass and friction of the interface. This can be explained by the phase advance which is the effect of larger delays used in the predictor than the system's delays. This situation appeared in the case of a more important uncertainty ($\tau_S = 120\ ms$) and for the uniform distribution ($\tau_1 = \tau_2 = 35\ ms$). In both cases the stability is guaranteed by the way of modeling the system delays and uncertainties, since the "everything" is taken into account when the control law is *elaborated*. In the case of uncertainties, once the gains are fixed for a constant time-delay, a *simple* way to find the delay' limits is the method proposed in Subsection 2.4.3, which gives a stability dependency between the system's and the Smith predictor delays for fixed PD gains.

Restricted motion

In the first two cases of restricted motion - with fix and moving objects, the contact is reproduced accurately and the impact sensation is naturally provided to the human operator. Depending on the values of the PD gains, the steady state error is *more or less* important. This phenomenon comes naturally since for some situations the stability limitations permit to have higher gains, while for others the gains are more restrictive. More precisely, the impact moment is accurately provided since the force is increasing fast, while the response stiffness depends directly to the K_p gain. For uniform and gamma with gap distributions the force felt represents 60% and 70% respectively from the *ideal* case, while for the fixed uncertainties and normal distribution the force represents 80% and 90% respectively.

In the third case considered for restricted motion - with moving object and perturbed feedback, there exist no problems linked to the impact feeling, the only inconvenient is the vibrating effect in steady state. This vibration can be slightly felt by the human operator in the scenarios considered here. For *higher* perturbations ($> \pm 10$ *mm*) the vibration will also increase, providing an unpleasant manipulation. It is worth mentioning that in *real life* situations, this case has a small probability of occurring, since here the wall position is randomly changing every millisecond. This scenario was presented in order to give an experimental overview of what happens if this feedback is perturbed.

Virtual Box

As mentioned before, these scenarios were elaborated in order to test the method capabilities at high velocities under fast changes from free to restricted motions and viceversa. Another important point of these scenarios was the multi points impacts and contacts, which in our opinion represents a challenging problem. As the results showed, the performances[4] are guaranteed in all tested situations. The impact sensation is clearly provided, even if it is just a *fast* contact or a *long time* steady state. In the case of long time steady states, like in the case of restricted motions, the steady state error is varying depending on the gains' values (for *higher* values of K_p the error is *smaller* and viceversa).

For all three scenarios - *simple*, *moving* and *uncertain* box, the human operator was able to feel that he is limited in all directions. Furthermore, for the *moving* cases, the moves of the walls were also accurately provided and felt by the human operator.

Analyzing now the performances of each time-delay case:

Constant time-delays

This is the most *desired* case, the use of the Smith predictor projects the problem in the free-of delays case. The proposed method for improving the Smith predictor in the case of haptics

[4] high level of transparency and system stability.

solves the problem of switching from free to restricted motion. The performances for this case are *basically* identical with the free-of delays case.

Constant uncertainties

This case illustrated the situations when the delay chosen in the Smith predictor is *higher* or *smaller* than the system's delay. When the delay from the Smith predictor is smaller than the system's delay, the compensation of the delay will be partially reached. On the other hand, if the delay from the predictor is higher, than there will be an effect of *anticipation* which results in a mass and friction compensator. As mentioned, this can be explained by the phase advance which is the effect of larger delays used in the predictor than the system's delays.

In all situations the system's performances and stability are guaranteed (with adequate controller tuning) in both free and restricted motions corresponding to each scenario. In order to avoid the instability situations, the time-delay limits must be correctly included in the elaboration of the stability regions. The method proposed in Subsection 2.4.3 represents a *simple* alternative to find the delay' limits between the system's and the Smith predictor delays for fixed PD gains, previously determined for a given time-delay.

Uncertain delays: uniform distributed time-delays

This distribution covers *basically* all the situations within a given interval. More precisely, if the delay is varying between a minimum and a maximum, the use of this distribution guarantees the system stability. In terms of haptics performances, since the gains are *smaller* than the other cases, the impact moment is accurately provided, but under a more important steady state error compared to other cases. In free motion the viscosity effect is *still low* ($< 6\ N$), guaranteeing a *transparent* manipulation.

Uncertain delays: gamma distributed time-delays with gap

In this case, the shape of the time-delay must be known in order to be characterized. This distribution represents a particular case of the previous one, and this is why the stability region is bigger. According to the literature, this is one of the most *popular* delay distribution for communication networks.

The performances obtained in this case are *better* ($\approx +10\%$) than the previous one, since the gains are closer to the ideal case.

Uncertain delays: normal distributed time-delays

The normal distribution characterizes fast varying delays. As presented in Subsection 2.4.6.4, this distribution comes to the ideal case, and implicitly the performances are slightly identical to the constant time-delay case. Resuming, if the mean value of the delays variation is *closed*

to the value of the delay used in the predictor, and if the variations are *fast* enough, then the system's characteristics will be *closed* to the *ideal* case.

3.5 Conclusions

This chapter presented an original control method, base on Smith predictor, designed for haptics. In haptics, like in teleoperation, the use of *simple* Smith predictor for compensating the time-delays' effects will solve the problem in free motion, but not for the restricted motion case. Everything is caused by the model inconsistency when passing from free into restricted motion. In haptics there is more information available which can be used to improve the system's performances. The virtual reality can provide precise information about the neighborhood of the controlled object in order to anticipate the contacts, i.e. the model switching. The central idea of the proposed method is the use of the additional information available in the virtual scene in order to maintain the accuracy of the model used in the predictor when passing from free to restricted motion or viceversa. By using this method, the system performances are increased, i.e. the transparency level is guaranteed.

The tuning of the controller must be made taking into account the *ideal* case which represents the best performances that can be obtain in terms of end user perception, as well as the stability limitations of each case. Thus, the basic idea is to find the closest gains to the ideal case, while respecting the stability domain for each case.

In Section 3.4, a complete experimental validation is presented starting from constant to uncertain delays tested in different scenarios from free and restricted motion with fix and moving objects to virtual box with moving sides and uncertain feedback. As it was presented in Subsection 3.4.6, the method provides interesting results in all situations, and in most of the cases *closed* to the *ideal* case.

As a general conclusion, in the case of uncertain time-delays, a special attention is required for the *behavior/shape* of the delay. In the case of uncertain time-delays, a special attention is required for the *behavior* of the delay, since it can be modeled in order to choose more efficiently the controller's parameters. The choice of the controller's gains can be more or less important depending on the distribution type. Thus, the largest choice of gain parameters, similar to the ideal case, is obtained for the normal distribution, while for the uniform distribution the choice is the most restrictive one. This comes as a normal consequence, since for the uniform distribution only the limits (minimal and maximal) must be known. A *bridge* solution is represented by the gamma distribution with gap, which may be applied if the delay shape satisfies the necessary conditions (less restrictive then the normal distribution, but more precise than the uniform distribution. Resuming, in the case of varying time-delays, with limited information about the

delay shape, the uniform distribution appears to offer the best solution, while the use of gamma distribution with gap or normal distribution requires more information.

In the next Chapter, based on the conclusions from Chapter 1, a second control strategy for increasing the performances of haptic systems will be presented.

Chapter 4

PD with gain scheduling depending on the distance

4.1 Problem analysis

As previously mentioned, in haptics, like in teleoperation, the problem of time-delays and their effects on stability and transparency remains open. Let us remind that in haptics there are two time-delay sources: the communication channel and the processing time for the virtual reality environment. More precisely, in free motion the delay effect can be felt by the viscosity phenomenon (high force feedback felt at the haptic interface end), and in the case of a hard contacts with the environment, the impact effect will not be stiff, or the most unwanted situation is to loose the system stability due to the delays. Therefore, the delays must be taken into account and included in the control laws.

Chapter 1 highlighted that the first iteration to solve this problem was to *port* the algorithms from teleoperation into haptics. Methods as Proportional Derivative (PD) with local dissipation, PD with passivity observer, PD with passive set-point modulation, etc. were appropriately *adapted* from teleoperation to haptics framework. As pointed out in the previous chapter, in our opinion, the second step will be to use the additional *available data* from the *virtual environment*, i.e. information can be extracted from the virtual environment about the distance between the objects, possible collisions and many other details regarding the system in order to improve the control algorithms. In this chapter,a second method using the additional data from the virtual environment is proposed.

Let us remind the haptic system objectives:

(i) *position tracking error* as small as possible between the haptic interface and the virtual object,

(ii) high degree of *transparency*, i.e. in free motion, the force feedback felt at the haptic interface end must be as small as possible and in case of hard contacts, a stiff response is desired.

This chapter proposes a gain-scheduling PD control approach depending on the distance until a possible collision. To the best of the authors' knowledge, the majority of methods proposed in the literature provide additional guarantees of stability, but in terms of end user perception (transparency), there are no major differences. In order to improve the performances with respect to the end user, different values for the controller gains should be used. Since the *classic* PD control is the *core* of all methods, this one was selected as the starting structure for variable gain implementation. Chapter 1 highlighted that interesting performances in free motion can be obtained using *small* gains, while for *good* performances in restricted motion, *high* gains are needed. The basic idea is to use a *small* gain in free motion in order to obtain a low viscosity movement and a *high* gain in restricted motion for achieving the desired stiff impact. The change between the two gains is achieved depending on the distance between the virtual manipulated object and the other virtual objects present in the scene.

For a clear presentation, some aspects already presented in the previous chapters will be reminded. The equations (1.1)-(1.2), representing the starting point, are given by the *classical* dynamic (nonlinear) equations of motion for two similar robots in the haptics framework:

$$M_h(x_h)\ddot{x}_h(t) + B_h(x_h,\dot{x}_h)\dot{x}_h = -F_h(t) + F_{op}(t), \tag{4.1}$$

$$M_v(x_v)\ddot{x}_v(t) + B_v(x_v,\dot{x}_v)\dot{x}_v = -F_v(t) + F_e(t), \tag{4.2}$$

where x_h, x_v are the haptic interface/virtual object position, F_{op}, F_e are the human/environmental forces, F_h, F_v are the force control signals, M_h, M_v are the symmetric and positive-definite inertia matrices, and B_h, B_v are the Coriolis matrices of the haptic interface and virtual object systems, respectively. The controllers's equations (2.4)-(2.5) of the system, illustrated in Figure 2.2, are given by:

$$F_h(t) = \underbrace{K_{d_h}(\dot{x}_h(t) - \dot{x}_v(t-\tau_2))}_{\text{delayed D-action}} \underbrace{+K_{p_h}(x_h(t) - x_v(t-\tau_2))}_{\text{delayed P-action}}, \tag{4.3}$$

$$F_v(t) = \underbrace{K_{d_v}(\dot{x}_h(t-\tau_1) - \dot{x}_v(t))}_{\text{delayed D-action}} \underbrace{+K_{p_v}(x_h(t-\tau_1) - x_v(t))}_{\text{delayed P-action}}, \tag{4.4}$$

where τ_1, τ_2 are the forward and backward time-delays and $K_{p_{h/v}}, K_{d_{h/v}}$ are the PD control gains, corresponding to the haptic and virtual side respectively.

From Figure 2.2, the equations describing the system response (2.6)-(2.7) are reminded:

$$X_h(s) = P_h(s)\left(F_{op}(s) - C_h(s)\left(X_h(s) - e^{-\tau_2 s}X_v(s)\right)\right), \tag{4.5}$$

$$X_v(s) = P_v(s)\left(-F_e(s) + C_v(s)\left(-X_v(s) + e^{-\tau_1 s}X_h(s)\right)\right), \tag{4.6}$$

where $X_{h/v}(s)$ denotes the Laplace transform of the time signal $x_{h/v}(t)$; similarly for $F_{op}(s)$ and $F_e(s)$.

Based on the equations (2.8)-(2.11), the characteristic equation of the feedback systems - (2.12) is recovered:

$$(1+P(s)C(s))^2 - (P(s)C(s))^2 e^{-(\tau_1+\tau_2)s} = 0. \tag{4.7}$$

Using the stability results for constant time-delays from Section 2.3, Figure 4.1 presents the stability regions for different values of τ_1 and τ_2 (forward and backward time-delays).

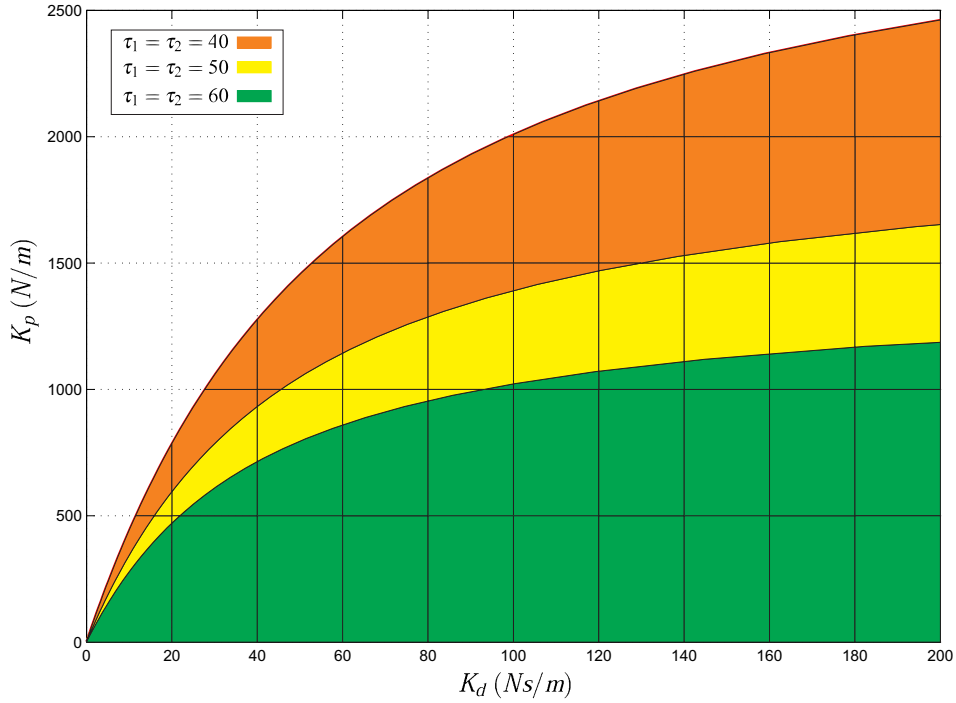


FIGURE 4.1: Stability regions for $\tau_1 = \tau_2 = 40, 50, 60$ *ms*.

In green is the *basic* stability region, and then each color highlights the new area *gained* in addition to the previous one. The most restrictive stability area is the one corresponding to the the biggest time-delay considered - $\tau_1 = \tau_2 = 60$ *ms*, and the largest one corresponds to the smallest delay considered - $\tau_1 = \tau_2 = 40$ *ms*.

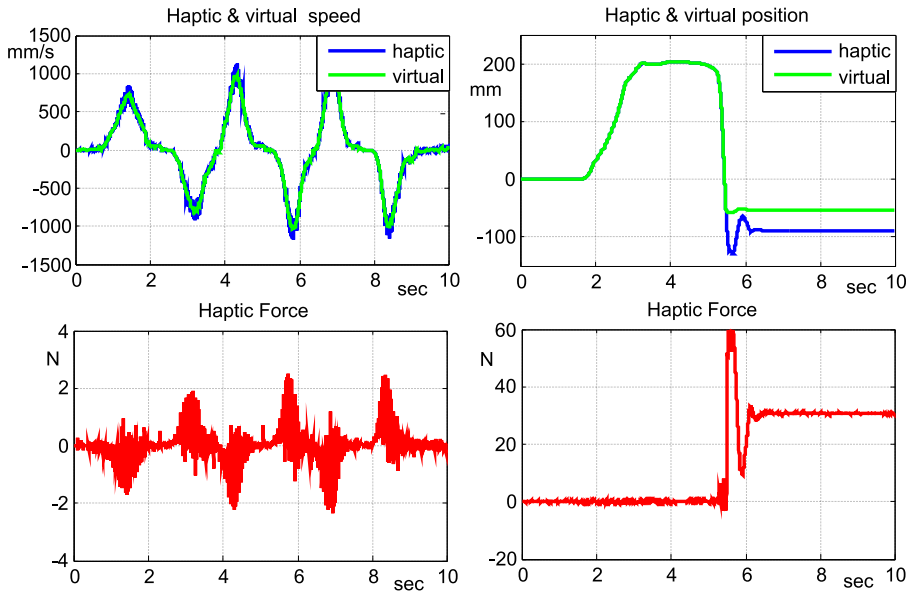
In order to highlight the problem, the cases free of and with time-delay will be discussed in detail.

Case without delays

This case is the starting point of the analysis, i.e. the time-delays are assumed to be equal to zero. In order to illustrate the theoretical development, an experimental platform will be used, which

was presented in Subsection 3.3. Let us remind that for guarantying a full control of the communication delays and processing time, all the control algorithms (for haptic interface/virtual object) and virtual environment simulations will be run on the same computer.

The system (which is considered to be the first degree of freedom of the experimental platform presented) will be analyzed in free and restricted motions. Figure 4.2 depicts such a behavior.



a. Free Motion. b. Restricted Motion.

FIGURE 4.2: Non-delayed Haptic System Behavior ($K_p = 2200\ N/m$ and $K_d = 75\ Ns/m$).

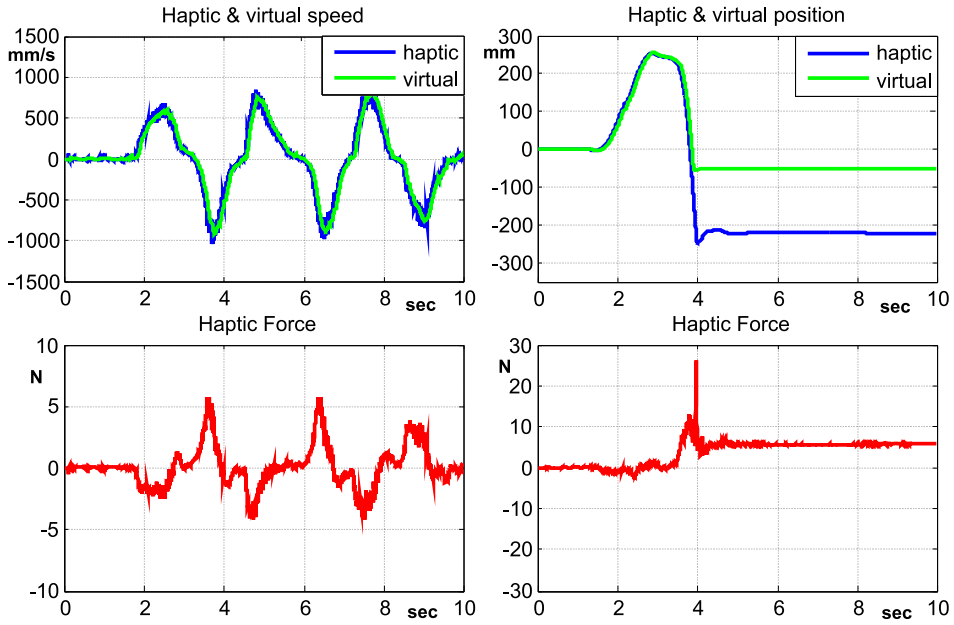
As largely discussed in Section 1.3, in free motion, there is a small feedback force, force which is not felt by the human operator. In case of hard contacts (wall contacts) the force increases very fast providing the contact perception in a very accurate way. In both cases the tracking error is very small.

Delayed Case

Next, the communication time-delays $\tau_1 = \tau_2 = 50ms$ will be considered. Based on the *ideal* behavior, the PD gains were tuned for the two cases respectively as follows, fulfilling stability results summarized in Figure 4.1:

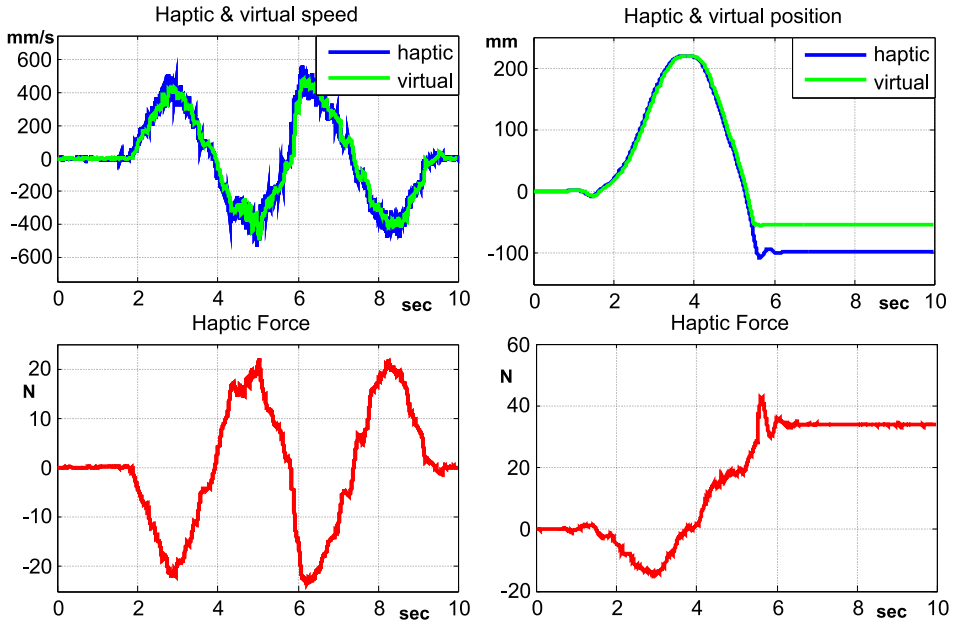
- best performance[1] in free motion case, Figure 4.3, obtained for $K_p = 150\ N/m$ and $K_d = 65\ Ns/m$.
- best performance in restricted motion case, Figure 4.4, obtained for $K_p = 1200\ N/m$ and $K_d = 65\ Ns/m$.

[1]The behavior *closest* to the ideal case.



a. Free Motion. b. Restricted Motion.

FIGURE 4.3: Best performance in free motion case ($K_p = 150$, $K_d = 75$).



a. Free Motion. b. Restricted Motion.

FIGURE 4.4: Best performance in restricted motion case ($K_p = 1200$, $K_d = 75$).

In the first case - *best performance in free motion case*, the performances are recovered for the free motion, while for restricted motion the impact feeling is *poor* and the steady state error is important.

In the second case - *best performance in restricted motion case*, the performances are recovered just partially for the restricted motion, i.e. the steady state error is similar, but the impact feeling

is soft instead of being stiff due to the *progressive* increasing of the force. In free motion, the viscosity effect is important, resulting in an unpleasant manipulation.

Generally, in free motion it is required to have F_h - described by equation (4.3), as small as possible and in restricted motion as high as possible. Here there is a contradiction, it is impossible to obtain the desired performance in both cases with the same gains.

Based on the ideal behavior the gains were tuned in order to obtain the best performance for each specific case (free or restricted motion) with the price of loosing the performance for the other case.

Generally, a compromise is made in order to obtain the *best possible* behavior in both cases using the same gains.

4.2 Proposed solution

In the previous section it was shown that interesting results can be obtained either in free motion, either in restricted motion. In this section a method to switch the gains according to the type of motion is proposed as a gain-scheduling approach.

Many different notions can be viewed or interpreted as *gain-scheduling*. The main idea is to switch or blend the gain values of controllers or models according to different operating conditions, or according to *preset* times. As the terms *switching* and *blending* already indicate, gain-scheduling may involve either continuous or discrete scheduling of controllers or model dynamics [97]. Among the literature many techniques of tuning can be found [18, 21].

When using the gain-scheduling control, the most *difficult* decision is the moment of switching and how to switch. In most cases, all the approaches return to the Linear Parameter Varying (LPV) theory, for more details, see [26, 34, 108]. In other words, based on the value of a parameter, the controllers and/or the models are modified in real time.

Starting from the basic concepts of gain-scheduling control, a control idea for haptic systems will be presented.

4.2.1 Proposed algorithm

As mentioned, the virtual environment offers additional information that can be used in the control algorithm. More precisely, like in the case of the *Smith predictor with distance feedback*, the distance between the controlled virtual object and another virtual object from the scene will be used in order to improve the system performances. The principle is to switch from the gain

used in free motion to the gain used in restricted motion and vice versa. The change must be completely transparent to the human operator. The gain changing will be achieved in two steps:

- first, the PD gain on the virtual side will be changed.
- second, after the time-delay τ_2, the PD gain on the haptic side will be updated.

Figure 4.5 resumes the approach above. The main idea is to update the gain on the virtual side based on the distance between the virtual object and the virtual environment (i.e. when the controlled virtual object is *close* to an impact, the gain will be switch from *small* to *high* gain, while if the object is further the a given limit, the gains will be switched back). Based on the update made on the virtual side, the controller gains on the haptic side are changed after the communication delay.

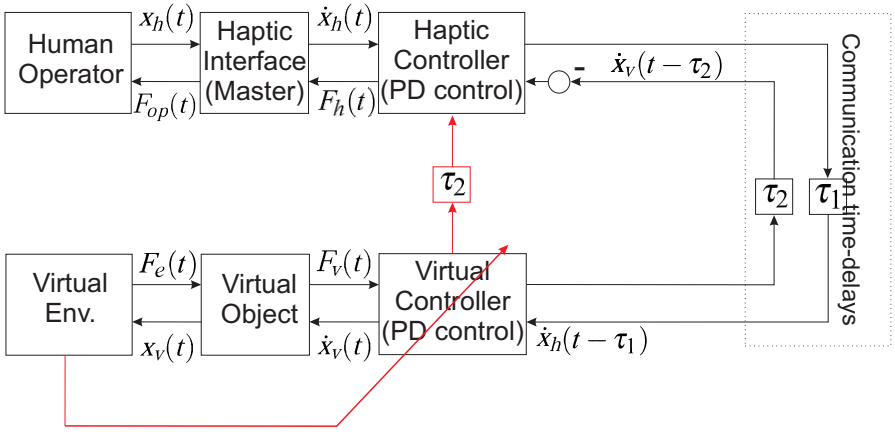


FIGURE 4.5: Scheme of PD control with gain-scheduling depending on the distance between the virtual object and the closest collision object from the virtual scene.

Figure 4.6 presents the workflow of the proposed algorithm. The increase factor $K_{p_{mim}} + (K_{p_{max}} - K_{p_{min}}) * 0.1^{x_v(k)-x_{wall}}$ was chosen in order to obtain an exponential growth of the gain.

Remark 4.1. The theory developed here is made under the assumption that the virtual wall represent the positive *limit* ob the virtual object movement. More precisely, before the impact $x_{wall} > x_v > 0$, and after impact $x_v > x_{wall} > 0$. For negative values the theory is applied viceversa.

Figure 4.7 illustrates the dependency between K_p and $x_v - x_{wall}$ under the above remark. This why the *limit* is considered to be negative. The key of the algorithm is to choose the *limit* of switching. More precisely the *limit* represents the distance from which begins the gain changing. During this position interval the K_p gain will have different values between $K_{p_{min}}$ and $K_{p_{max}}$.

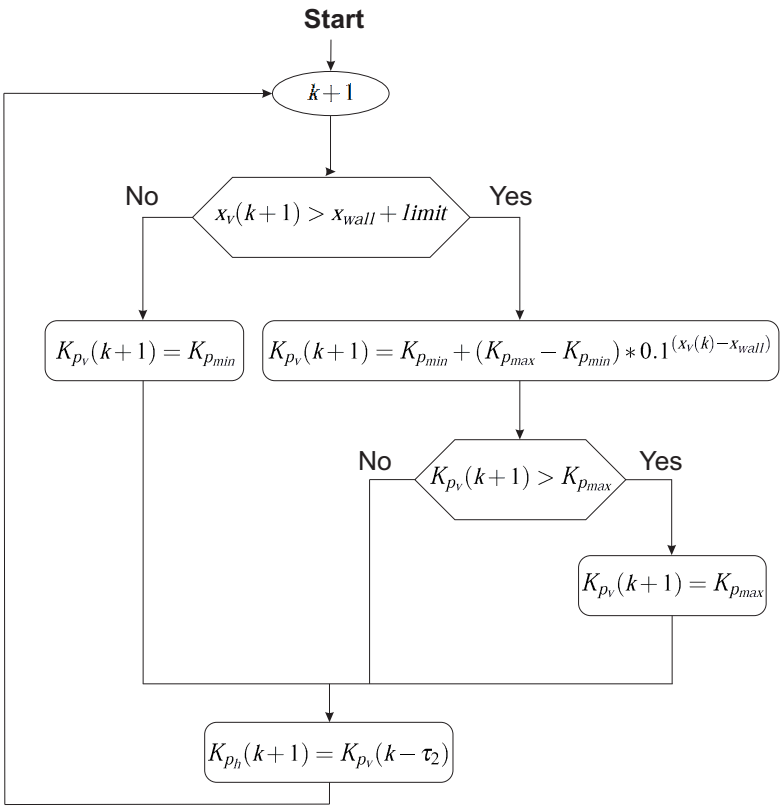


FIGURE 4.6: Algorithm workflow.

4.2.2 Discussions

For the free motion case, a *small* K_p gain is used in order to guarantee low viscosity effect without loosing the position and speed coordination between the haptic interface and the virtual object. For the restricted motion, a *high* K_p gain is considered in order to confer a stiff response to the human operator at the haptic end.

As already mentioned, the tuning regarding the *small* gain ($K_{p_{min}}$) must be chosen as small as possible in order to obtain the desired tracking error and respecting the stability zone based on the time-delay type (fixed or uncertain). In the sequel, the tuning of the *high* gain ($K_{p_{max}}$) must be chosen as big as possible respecting the stability constrains. If needed, in order to fulfill the stability conditions also the K_d gain could be switched between the two values corresponding to free and restricted motion.

The gain tuning, i.e. *small* and *high* gains is made in order to be *close* to the *ideal* case (free-of delay) in free and restricted motion respectively. More precisely, the *small* gain ($K_{p_{min}}$) should be chosen as small as possible in order to guarantee the desired tracking error in free motion, while respecting the stability constrains. As for the *high* gain ($K_{p_{max}}$), it should be chosen as big as possible with respect to the stability conditions.

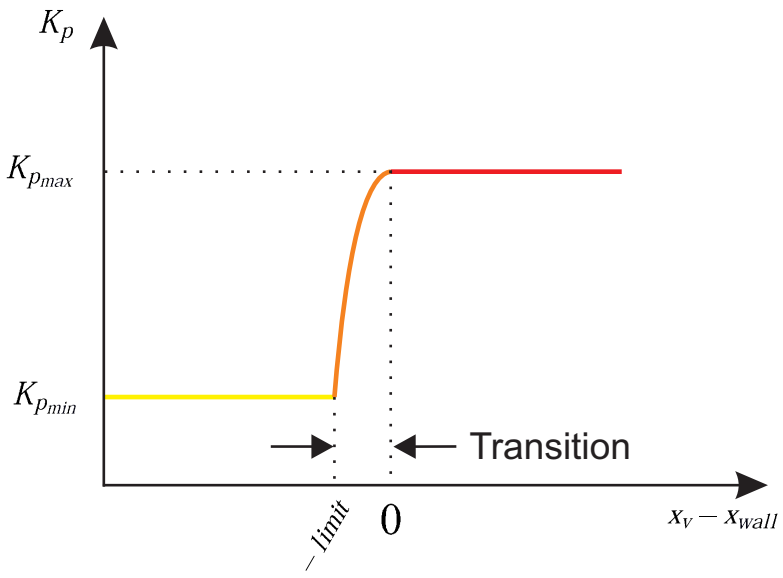


FIGURE 4.7: Dependency between K_p and $x_v - x_{wall}$.

With these considerations, both gains are chosen based on the end user perception and with respect to the theoretical results presented in Figure 4.1 according to the theory developed in Section 2.3. The following gain values are finally selected:

$$\begin{array}{ccccccccc} K_{px_{min}} & = & K_{py_{min}} & = & K_{pz_{min}} & = & 150 & N/m, \\ K_{px_{max}} & = & K_{py_{max}} & = & K_{pz_{max}} & = & 1200 & N/m, \\ K_{dx} & = & K_{dy} & = & K_{dz} & = & 65 & Ns/m, \end{array}$$

and the time-delays: $\tau_1 = \tau_2 = 0.050sec$.

The most challenging part of this solution is the switching part, there are several conditions that must be met simultaneously:

- When passing from *small* to *high* ($K_{p_{max}} \rightarrow K_{p_{min}}$): a smooth transition must be reached which also provides the desired stiff response.
- When passing from *high* to *small* ($K_{p_{min}} \rightarrow K_{p_{max}}$): the *sticking*[2] effect must be avoided.
- In case of changing the direction[3] during the transition phase, the system must be able to react in a smooth and transparent way.

All these conditions will be now detailed. The first situation is switching from free motion to restricted motion, i.e. from the *small* gain to the *high* one. According to the experimental

[2]The so-called *sticking* effect is the resistance felt in free motion until the *high* gain is switched to the *small* one. More precisely when passing from restricted motion to free motion, the resistance is suddenly changed due to the gain switching and then the effect is felt like a release, similar to a peel off (detachment).

[3]A direction change defines the reverse move just before or in the moment of impact in the case of free to restricted motion, while in the case of restricted to free motion the reverse move is made viceversa

results, in order to obtain a "*pleasant*" manipulation in terms of end user, the transition must be made when the impact moment is *near*, but far enough to ensure the gain changing at the haptic side at the impact moment. If the passing is too early, the user will feel the viscosity effect and the collision will be smooth instead of stiff, see Figure 4.8.a (F-R[4]), where the dashed zone represents the moment when the system becomes viscous and the impact is not stiff. If the passing is too late, the feedback will be very *aggressive* due to the high force ($\approx 95\ N$, double than a normal feedback), giving an unreal way of feeling the collision, see Figure 4.8.b. Furthermore, if the passing is *extremely* late, the system will strongly oscillate risking to become unstable, see Figure 4.8.c. Even if the conditions seem restrictive, it still remains a large range of possibilities depending on the time-delay.

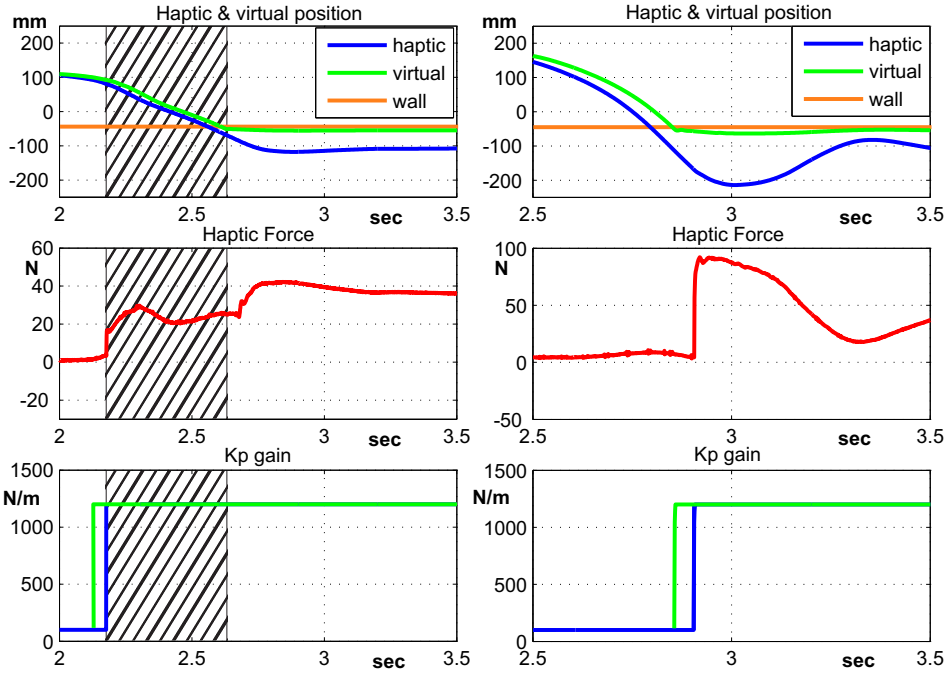
Figure 4.10.a depicts the desired behavior (close to the ideal case) of the system when switching from free motion to restricted motion, i.e. from the *small* gain to the *high* one. This behavior was obtained based on the conditions depicted above.

The second situation considers switching from restricted motion to free motion, i.e. from the *high* gain to the *small* one. This is a more complicated problem because in the first case there is a chronological order, but in this case the user must begin the reverse movement, the switching will be first made on the virtual end, and then will be also made on the haptic side. During this time the user will feel the viscosity effect and at changing the gain will feel a *sticking* effect, see Figure 4.8.d (R-F[5]). More precisely, while the user passes in free motion, the viscosity effect is still important until the haptic controller gains are updated from *high* to *small* values. This period depends directly on time-delay and the viscosity effect increase with the velocity, i.e. for slow motion the viscosity effect is lower, while at high velocities is more important. For the moment this effect cannot be completely eliminated but it can be reduced by using a fast and smooth gain changing, see Figure 4.10.b.

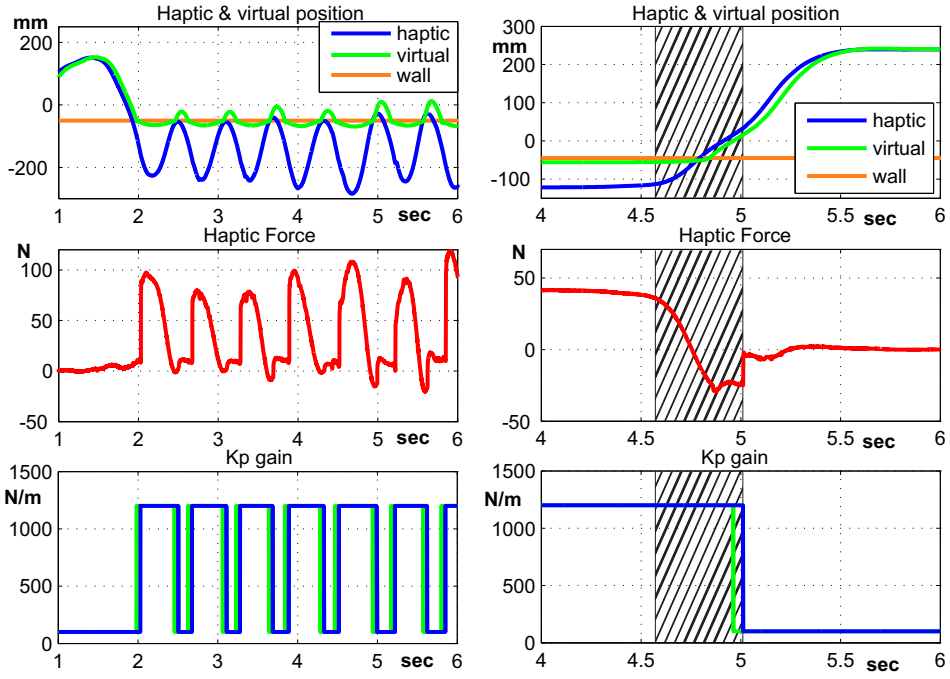
Finally the *last moment* changes during the transition must be taken into account. More precisely, the system must be able to switch back to the previous situation in case of a change during transition without loosing the transparency. This condition is linked to the previous ones (smooth transition and no sticking effect), generally if the transitions from free to restricted motion and viceversa are correctly achieved, then the last moment changes condition comes as a consequence.

Figure 4.9.a illustrates the situation when the user decides to switch from restricted to free motion, and in the last moment it returns in restricted motion. On the K_p graphic from Figure 4.9.a, the gain is switched from *high* to *small* for a short period of time ($\approx 0.2\ sec$ - the dashed zone) and then restored. Next, Figure 4.9.b illustrates the last minute change from free to restricted motion. Actually, the user touches the wall, fact that triggers out the switching process

[4] F-R - Free to restricted motion
[5] R-F - Restricted to free motion



a. F-R: Early Switch. b. F-R: Late Switch.

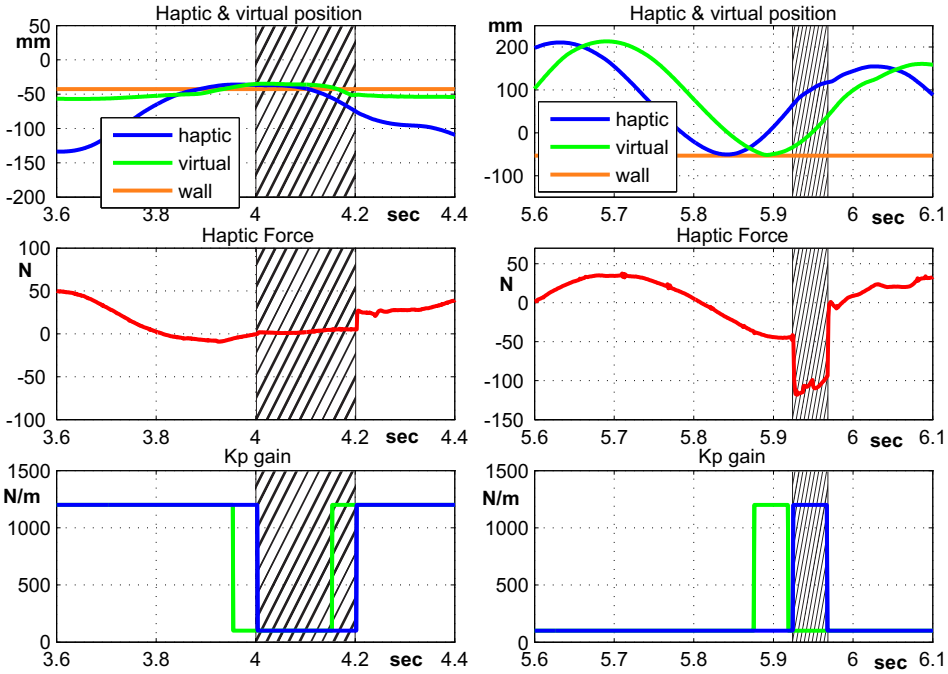


c. F-R: Too Late Switch. d. R-F *Sticking* Effect.

FIGURE 4.8: Different effects linked to gain switching.

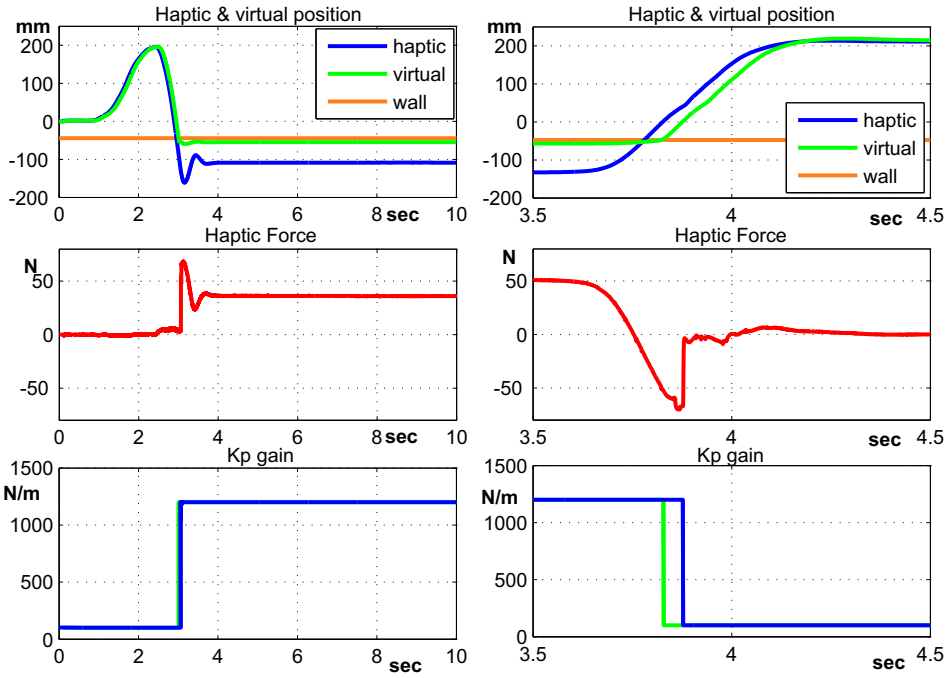
from *small* to *high* gain (see Figure 4.9.b - the dashed zone). In both cases there appears an unexpected gain switch. In the first case (last moment change from restricted to free motion - Figure 4.9.a), the effect felt by the end user is almost insignificant, since the force variation due to the gain switching is not important. In the second case (last moment change from free to

restricted motion - 4.9.b), the force variation is more important resulting in a damping effect of short duration (≈ 0.05 *sec* - time needed for update).



a. R-F: Last moment *change*. b. F-R: Last moment *change*.

FIGURE 4.9: Last moment changes.



a.F-R transition. b.R-F transition.

FIGURE 4.10: Optimum transitions.

Based on these conditions the switching parameter *limit* was tuned experimentally in order to obtain the best performance in all situations. The gains are updated on each axis independently. For example if there is a collision on X axis, a *high* gain will be used only for this axis and for the others the *small* gains will be maintained. Section 4.3 presents a complete experimental validation of the method and further discussions.

4.3 Results and analysis

In this section, the gain-scheduling approach will be experimentally tested and analyzed for different scenarios and under different types of time-delay.

The experimental platform that will be used, was previously described in Section 3.4. Let us remind that the main objective of this platform is to guarantee a full control of the communication time-delays.

Like in the case of the *Smith predictor with distance feedback*, all the experiments are carried out by a human operator, explaining why the conditions will not be exactly the same in all experiments.

In the same spirit as in the previous chapter, in order to create an efficient way of testing the proposed method, multiple scenarios were defined:

1. free motion (random motions on each axis),
2. maintained restricted motion (continuous wall contacts on each axis for more than 5 *sec*) and returning in free motion,
3. random motions inside a virtual box i.e. random motions with or without contacts on each axis and multi-point contacts.

The first two scenarios will point out the basic functionalities of haptic systems, using the proposed control method under different types of delays. The last scenario proposes a more *realistic* experiment, since the complexity level of the environment has been increased and fast gain changes are required. The purpose is to test the proposed method under fast changes switches from free to restricted motions and viceversa in *small* spaces.

Based on the analysis of the PD control used for haptic systems, presented in Section 2.3, and considering the analyzed types of delays, three experimental time-delay cases will be used in each scenario:

- constant time-delays,

- uncertain delays: uniform distributed time-delays,
- uncertain delays: gamma with gap distributed time-delays.

4.3.1 Constant time-delays

In this case, the time-delays are considered to be $\tau_1 = \tau_2 = 50\ ms$ and the controller's gains used for all the experiments presented in this subsection, corresponding to the constant time-delays case, are given by:

$$\begin{array}{rcrcrcll} K_{px_{min}} & = & K_{py_{min}} & = & K_{pz_{min}} & = & 150 & N/m, \\ K_{px_{max}} & = & K_{py_{max}} & = & K_{pz_{max}} & = & 1200 & N/m, \\ K_{dx} & = & K_{dy} & = & K_{dz} & = & 65 & Ns/m. \end{array}$$

Scenario 1. Figure 4.11 presents the free motion case under constant time-delays.

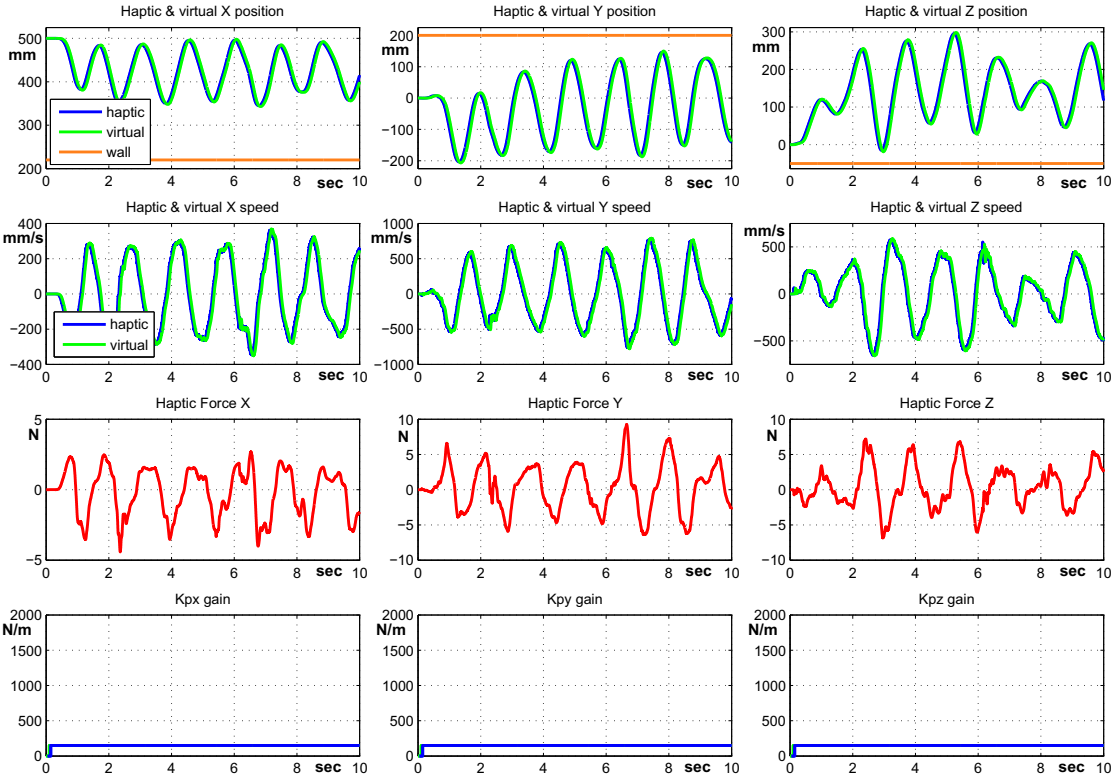


FIGURE 4.11: Free motion case under constant time-delays.

From Figure 4.11 it can be seen that the viscosity effect is low ($< 5\ N$, except some peaks due to direction changes at high velocities), providing an agreeable manipulation. Since there are no motion changes, the gain is maintained for all the duration of the experiment.

Scenario 2. In Figure 4.12 the restricted motion case is illustrated for th constant time-delay case.

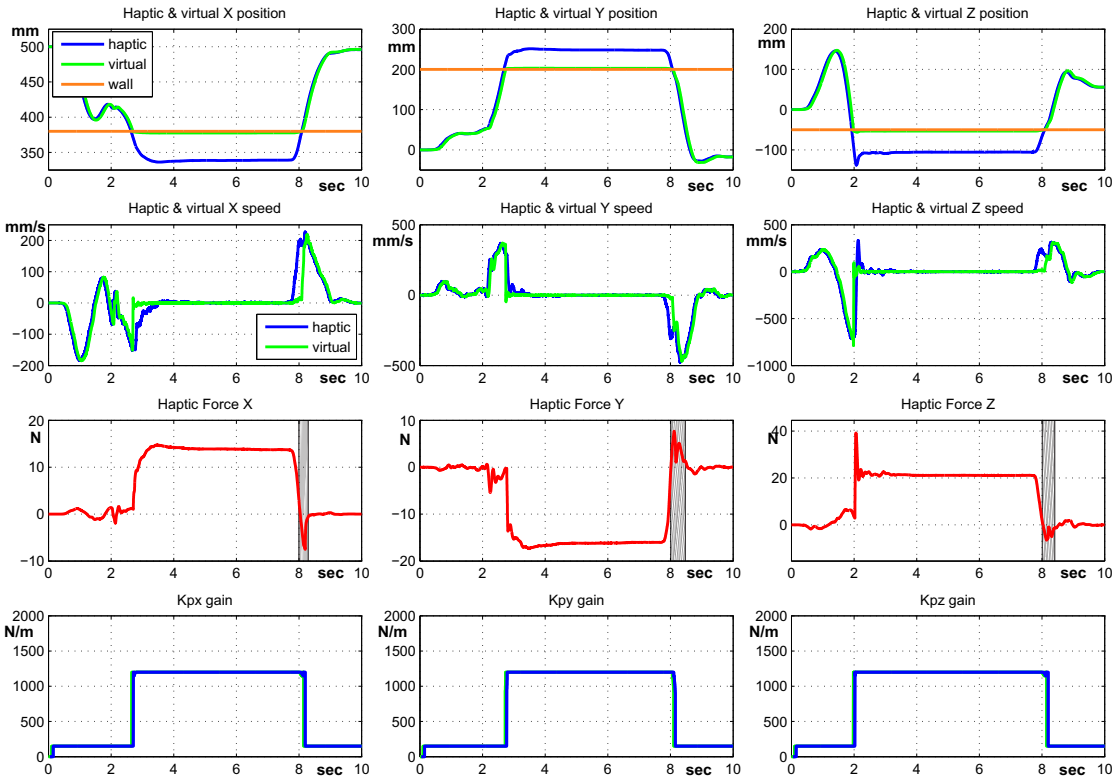


FIGURE 4.12: Restricted motion case under constant time-delays.

Here, the K_p gains are switched based on the type of motion - free or restricted. As the figure shows, the impact feeling is correctly provided (the force increases fast), and the end user is able to feel the contact in a stiff way. When *returning* from restricted to free motion, the sticking effect is felt for a *small* period time (the dashed zone of the force graphics on each axes), without any particular inconveniences. The overall behavior of the system provides performances a realistic feeling as well a pleasant manipulation.

Scenario 3. For this example, a virtual box - Figure 3.4, will be used as virtual environment. Figure 4.13 presents the results for this case.

As mentioned, the purpose of this scenario is to test the proposed method under fast switches. The restricted motion situations can be seen on the graphics by the exceeded of the virtual wall limits on each axes. As it can be seen there exist many changes on the K_p gains, especially *last moment* changes, which are not perfectly managed. More precisely, due to these fast switches, the effects felt by the human operator are not very clear. Occasionally there are some unnatural increases of viscosity effects, while other times the impact is provided in a *soft* way.

In this subsection, under constant time-delays, the three scenarios were presented in order to highlight the performances of the proposed method. In *clear* situations, free motions or long time (> 1 *sec*) restricted motions, the systems provides accurate perceptions. Also, singular last moment changes can be *managed* correctly. The insufficiency appears when fast successive

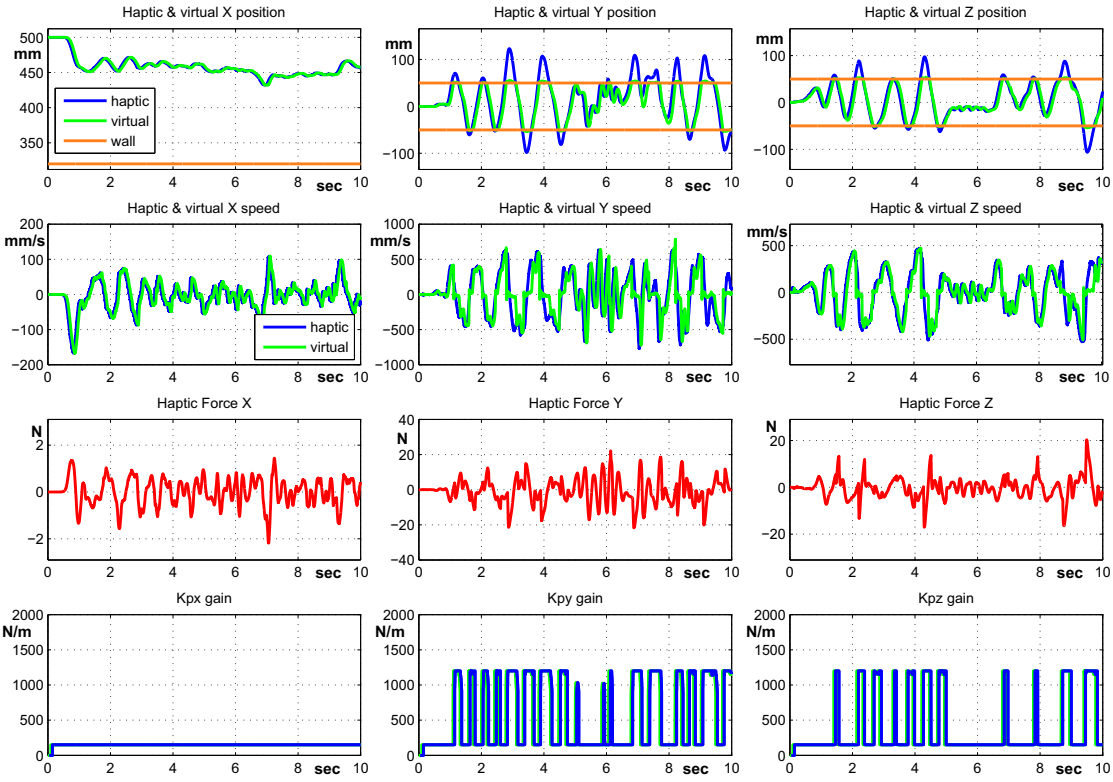


FIGURE 4.13: Random moves inside a virtual box under constant time-delays.

changes appear, as illustrated in the last scenario, when due to time-delays the system is not able to update the gains fast enough in order to provide a realistic situation.

4.3.2 Uncertain time-delays: uniform distribution

In the sequel, the time-delays are considered to have a totally random variation between a minimum and maximum value; here $\tau_1, \tau_2 \in [35\ ms, 65\ ms]$. The controller's gains used for all the experiments presented in this subsection are given by:

$$\begin{array}{ccccccc} K_{px_{min}} & = & K_{py_{min}} & = & K_{pz_{min}} & = & 150 \quad N/m, \\ K_{px_{max}} & = & K_{py_{max}} & = & K_{pz_{max}} & = & 1500 \quad N/m, \\ K_{dx} & = & K_{dy} & = & K_{dz} & = & 65 \quad Ns/m. \end{array}$$

These gains were tunned in order to obtain the best performances in terms of end user perceptions, while respecting the stability constrains corresponding to the uniform distributed time-delays case, illustrated in Figure 2.11.

Scenario 1. Figure 4.14 presents the free motion case for uniform distributed time-delays.

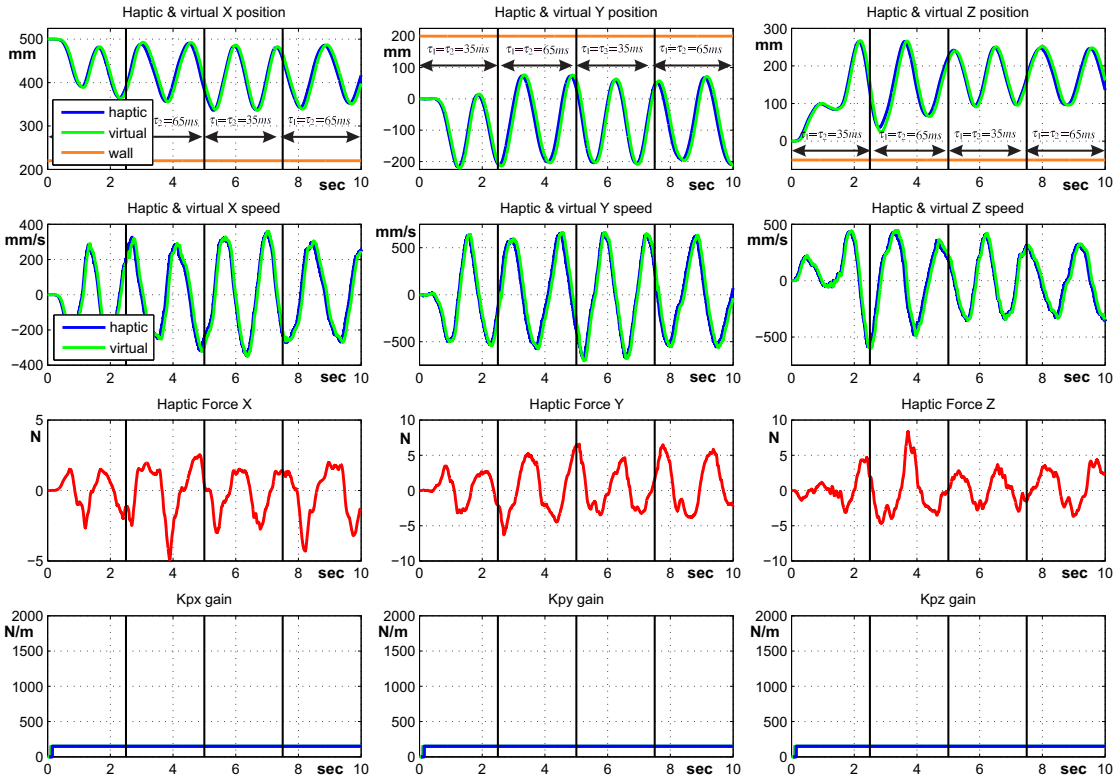


FIGURE 4.14: Free motion case under uniform distributed time-delays.

Compared to the previous case, here the time-delays are changing (as illustrated in Figure 4.14), but the viscosity effect remains low ($< 4\ N$) in all situations, providing an agreeable manipulation.

Scenario 2. Next, in Figure 4.15 the restricted motion case is highlighted.

In addition to the previous case, since the $K_{p_{max}}$ value is bigger, the impact feeling is provided in a stiffer way ($\approx +15\%$ force increase for the same steady state error). On the other hand, when passing from restricted to free motion, the sticking effect is felt for a similar time period, but with a *small* increase in terms of force due to the bigger value of the $K_{p_{max}}$ ($\approx +10\%$).

Scenario 3. The last example of this subsection proposes a virtual box - Figure 3.4, as virtual environment. Figure 4.16 presents the results for this case.

As expected, similar to the constant time-delay case, due to the fast switches, the human operator perceptions are not clear due to the often switches between the small and high gains. More precisely, the fast changes are causing a frequent gain switching resulting in an ambiguous perception since sometimes the *small* gain ($K_{p_{min}}$) is used in restricted motions, while other times the *high* gain $K_{p_{max}}$ is used in free motions.

In this subsection, the three scenarios were presented under uniform distributed time-delays, case which corresponds to a wide range of possible delays shapes and uncertainties. Similar

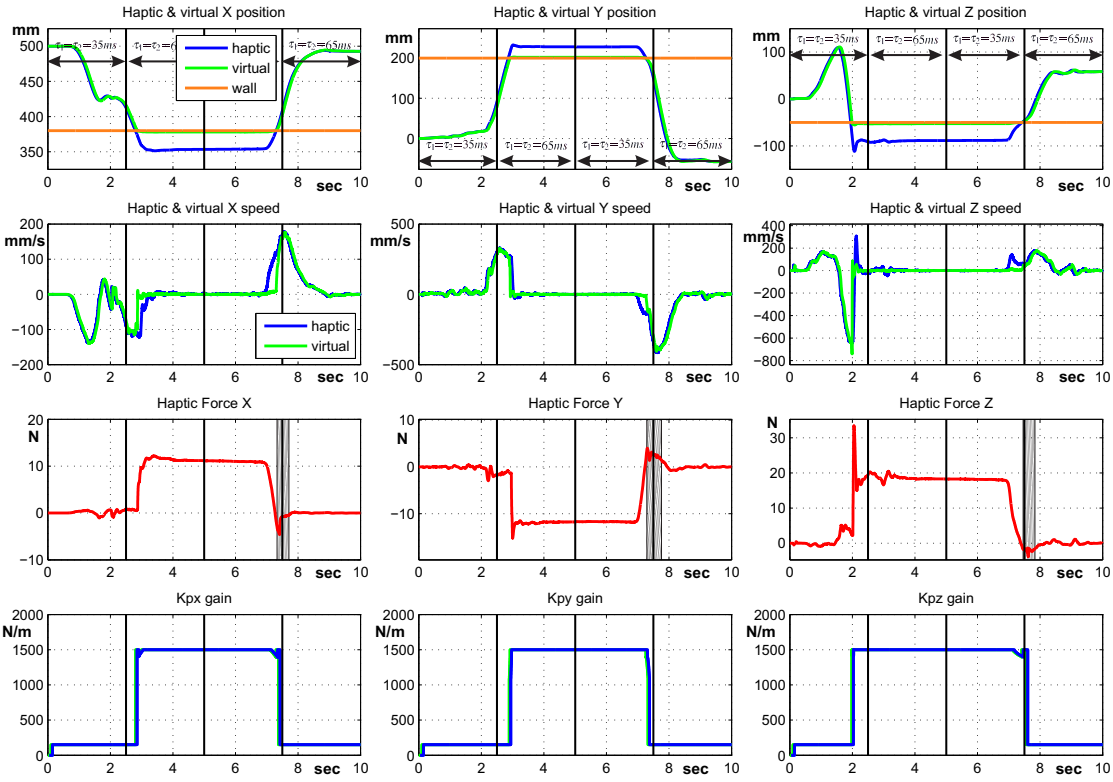


FIGURE 4.15: Restricted motion case under uniform distributed time-delays.

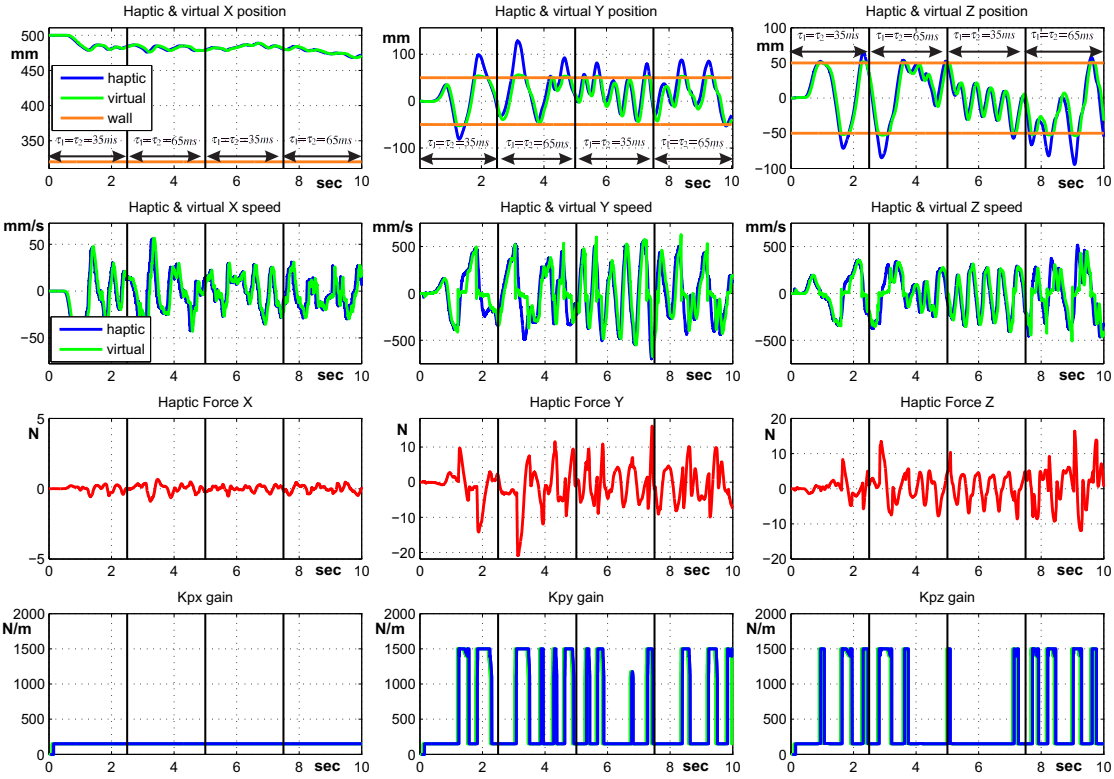


FIGURE 4.16: Random moves inside a virtual box under uniform distributed time-delays.

to the constant time-delays, the performances in *clear* situations (free or restricted motion for more than 1 *sec* and even singular last minute switches) are respecting the desired performances, providing a realistic manipulation. The loss of performances comes with the fast changes in short time periods (more than 2 per second), since the strategy is directly linked to delays.

4.3.3 Uncertain time-delays: gamma with gap distribution

In this last subsection, the time-delays are considered to have a variation shape as described in Figure 2.10, with $\tau_1 = \tau_2 = \hat{\tau} + nT$, where $\hat{\tau} = 40\ ms$, $n = 1$ and $T = 10\ ms$ (as described in Subsection 2.3.3.2). The controller's gains used for all the experiments presented in this subsection, corresponding to the gamma with gap distributed time-delays case - Figure 2.12, are chosen as follows:

$$\begin{array}{ccccccc} K_{px_{min}} & = & K_{py_{min}} & = & K_{pz_{min}} & = & 150 \quad N/m, \\ K_{px_{max}} & = & K_{py_{max}} & = & K_{pz_{max}} & = & 1800 \quad N/m, \\ K_{dx} & = & K_{dy} & = & K_{dz} & = & 65 \quad Ns/m. \end{array}$$

Similar to the previous cases, these gains were tunned in order to obtain the best performances in terms of end user perceptions, while respecting the stability constrains corresponding to the gamma with gap distributed delays, presented in subsection 2.3.3.2.

Scenario 1. The free motion case for uniform distributed time-delays is illustrated in Figure 4.17.

In this example, the time-delays are changing randomly, respecting the gamma with distribution and having a similar shape as the one presented in Figure 2.10. Similar to the previous case, the viscosity effect remains low ($< 5\ N$) (as illustrated in Figure 4.17) providing the desired performances in free motion.

Scenario 2. In this example, the restricted motion case is presented in Figure 4.18, following the second scenario presented in the beginning of this section.

The use of a *high* value for $K_{p_{max}}$ provides a *powerful* impact feeling, i.e. $\approx 25\%$ and $\approx 10\%$ compared to the constant delays and uniform distributed delays respectively. As in the other two examples, when passing from restricted to free motion, the sticking effect is felt for a similar time period, but with an increased force compared to the previous ones due to the increase of the $K_{p_{max}}$ gain ($\approx 17\%$ compared to the constant delays and $\approx 10\%$ compared to uniform distributed delays).

Scenario 3. In the last example, a virtual box - Figure 3.4, is used as virtual environment. The results for this case are presented in Figure 4.19.

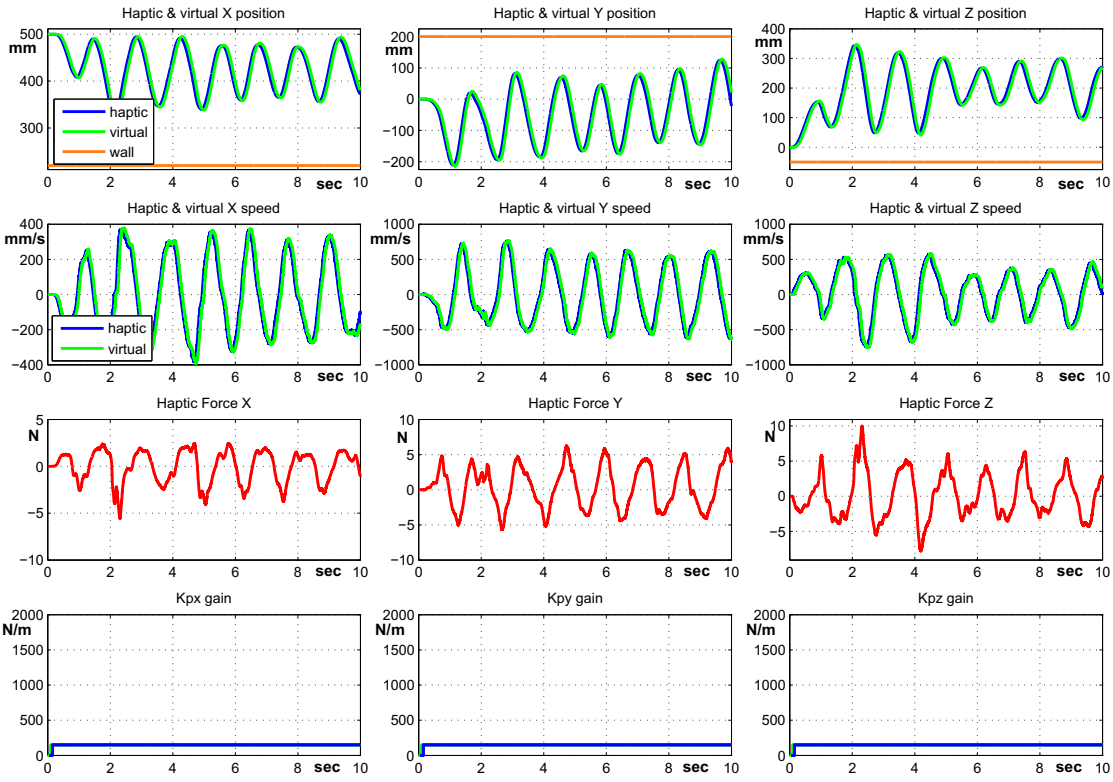


FIGURE 4.17: Free motion case under gamma with gap distributed time-delays.

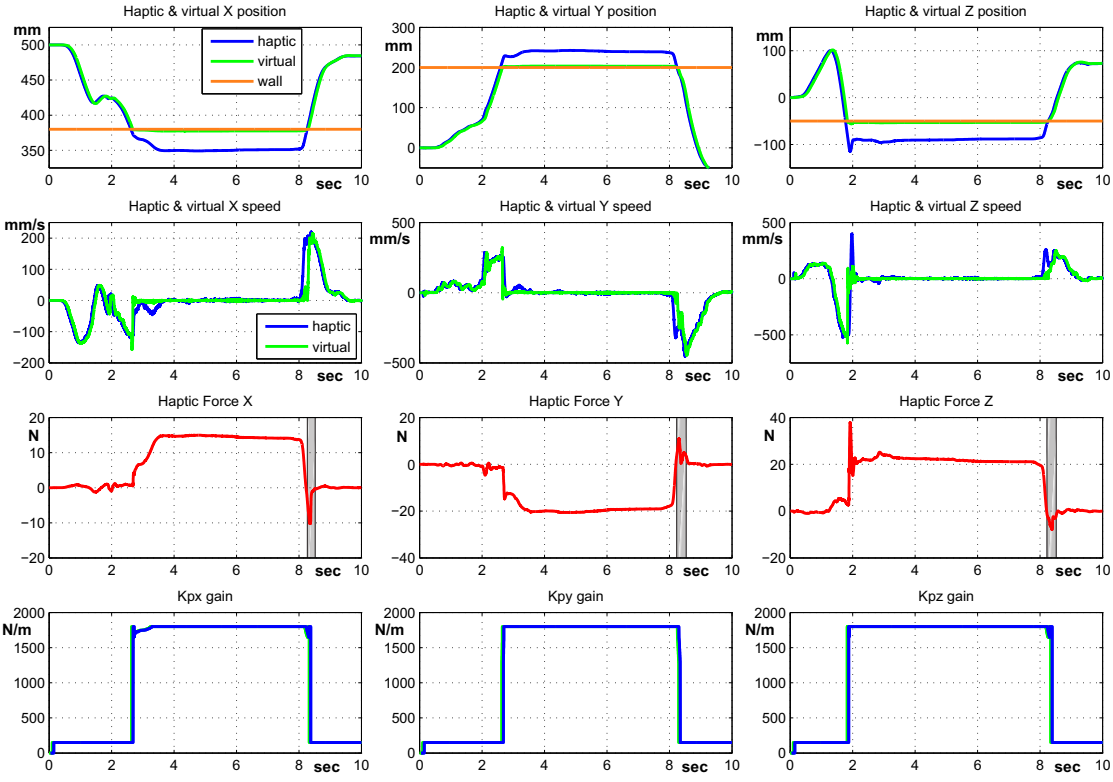


FIGURE 4.18: Restricted motion case under gamma with gap distributed time-delays.

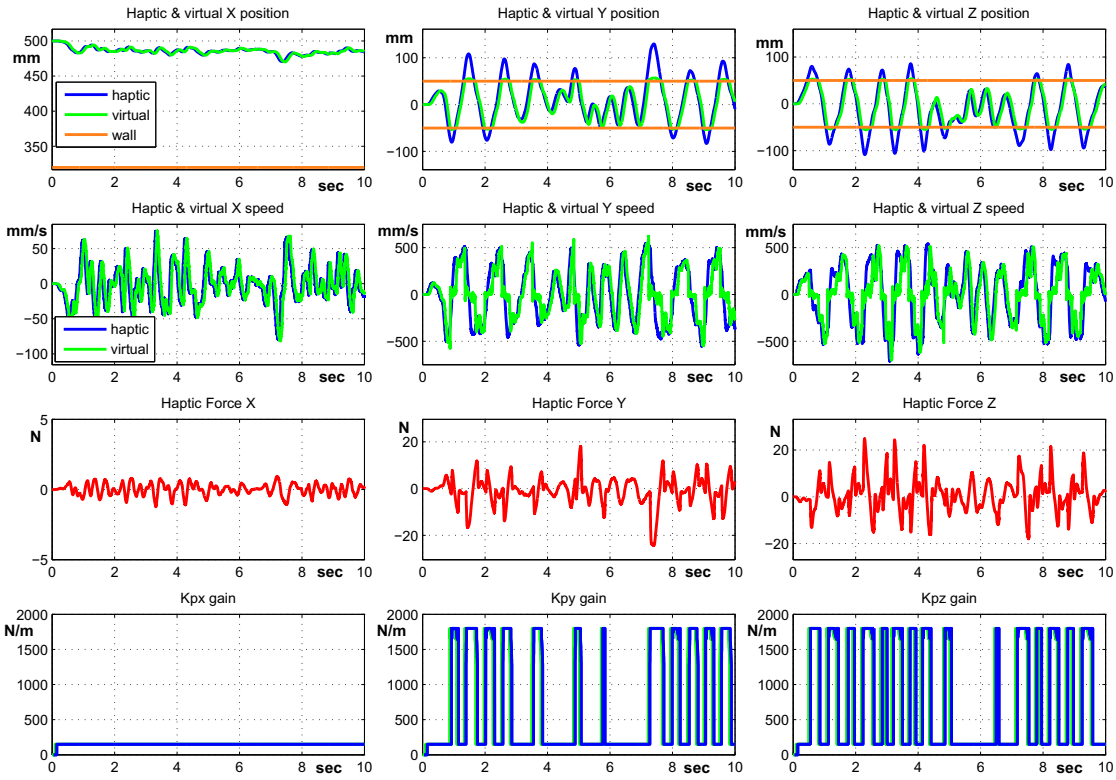


FIGURE 4.19: Random moves inside a virtual box under gamma with gap distributed time-delays.

Similarly to the previous cases, the human operator perceptions are not clear due to the fast switches.

This subsection presented the three scenarios under gamma with gap distributed time-delays, a common distribution for communication networks. The main objective was to test the method capabilities under random variations of time-delays. The next sections will largely discuss the obtained results.

4.3.4 Discussions

For all three time-delays circumstances considered, the performances in free motion are consistent with the reality, providing a correct manipulation, under *low* viscosity effect ($< 5\ N$). It is worth mentioning that, the performances of *simple* free motion case are not influenced by delay variations. Basically in all situations, the small gain ($K_{p_{min}}$) must be chosen as small as possible in order to fulfill the stability conditions and to assure the desired tracking error between the haptic interface and virtual object.

In restricted motion (when passing from free to restricted), the impact feeling is provided in realistic way in all cases. It is worth pointing out that, depending on the time-delay, the maximum

value of the $K_{p_{max}}$ can be increased, providing a stiffer impact sensation and a lower steady state error. The best results are obtained for the gamma with gap distribution, while for the constant time-delay, the results are the *poorest*. On the other hand, when switching back from restricted to free motion, the *sticking* effect is also directly proportional with the values of the $K_{p_{max}}$. More precisely, the lowest sticking effect is obtained in the case of constant delays and the biggest one is obtained for gamma with gap distribution. The uniform distribution presents the best compromise between the three cases analyzed, from the point of view of the transitions' effects. With these considerations, the performances in restricted motion (more precisely when passing from free to restricted motion and maintained the force for at least 1 *sec*) are directly influenced by the delay variation type which gives the maximum value of the *high* gain $K_{p_{max}}$. Furthermore, the *sticking* effect which appears when switching back from restricted to free motion is also directly influenced by the $K_{p_{max}}$ value, i.e. for bigger values of $K_{p_{max}}$ the *sticking* effect will also increase.

The last scenario - the virtual box, pointed out some inconveniences of the proposed approach. Due to the *tiny* size of the virtual box, the last moment changes were highlighted. The situations are similar for all three time-delays considered. The unwanted increase of viscosity in free motion, as well as the decrease of stiffness in restricted motion, conduct to an overall decrease of performances under fast or last moment changes. The problem is linked to the algorithm of switch, which is based on a fixed distance limit for switching between the gains. In *normal* conditions[6] the method is working correctly, but for fast switches the gain *updating* is cannot assure the desired performances.

4.4 Conclusions

This chapter presented a study of the possibility of using a gain scheduling approach in haptics. Since there exist two functioning situations, and implicitly two corresponding tunings, the gain scheduling approach seems to be appropriate for such a situation. The switching method proposed here improves the overall performances in *clear* situations, i.e. transition from free to restricted motions or viceversa at *low frequencies*. The experiments carried out, for an overall time-delay of 100 *ms*, pointed out the improvements, as well as the limitations of the method in case of *fast* switches. It is worth mentioning that the choice of the switching *limit* was made in order to cover the velocities of *common* manipulations.

As already mentioned, the tuning regarding the *small* gain ($K_{p_{min}}$) must be chosen as small as possible in order to obtain the desired tracking error and respecting the stability zone based on the time-delay type (fixed or uncertain). In the sequel, the tuning of the *high* gain ($K_{p_{max}}$) must be chosen as big as possible respecting the stability constrains. If needed, in order to fulfill the

[6] Here, *normal conditions* refer to *simple* free to restricted motion while the force is maintained for at least 1 *sec* or vice versa case, as well as for singular last moment changes (not more than one in 1 *sec*)

stability conditions also the K_d gain could be switched between the two values corresponding to free and restricted motion.

The method works correctly for *relatively* small communication delays (< 20 *ms*), while for bigger ones (> 25 *ms*) there appears a loss of performances, especially under fast or last moment changes. A possible solution for the method improvement is the use of a more complex gain changing algorithm based on velocity, and time-delay values. More precisely, for a *high* velocity the limit of switching can be increased in order to change the gain faster, while for a smaller velocity the limit should be decreased in order to complete the updating process on both sides by the impact moment. This could be formalized also as a optimization problem in finding the optimum *limit* depending on velocity and time-delay in order to complete the updating process in the impact moment. Another point which can be added in order to decrease the *sticking* effect which appears when switching from free to restricted motion, is to take into account that once the user is moving in the opposite side of the wall, the gain on the haptic side may be increased before receiving the information from the virtual controller. Resuming, a dynamical *limit* for each axis depending on the velocity may reduce substantially the limitations of the method.

Compared to the method presented in the previous chapter - Smith predictor with distance feedback, the performance obtained here are less interesting, since the PD with gain scheduling method is strongly related to velocity and time-delay. Another aspect is represented by the loss of performances under fast switches (from free to restricted motion and viceversa). The reason which makes the difference between the two methods is that the Smith predictor with distance feedback acts like a predictor (as the name says), while the gain scheduling acts based on the present actions without any anticipation. The improvement perspectives for the gain-scheduling approach presented above result also in a sort of predictor, since the *limit* is tunned dynamically in order to have the updating process achieved in the impact moment. Resuming, among the two methods discussed in their present form, the Smith predictor-based approach gives better results. It is worth mentioning that with further improvements also the PD with gain scheduling could provide better results.

This chapter ends the work presented in this thesis. In the last chapter, some general conclusions and discussions are presented in order to complete the overall work.

Conclusions and Perspectives

The main objective of this thesis was to provide new solutions in order to improve the performances in terms of end user perception for haptic systems affected by communication time-delays. As previously mentioned, the main problems of haptic systems in the presence of time-delays are: on one hand, the viscosity effects in free motion, and on the other hand, the non-stiff response in case of hard contacts.

The first step of this work was to analyze and experimentally test the most common control methods used in haptics (classic Proportional Derivative (PD) controllers, PD controllers with local dissipation, PD controllers with passivity observer, PD controllers with passive set-point modulation, wave scattering transform and Smith predictor). The experimental results carried out on one degree of freedom (one degree-of-freedom) haptic platform presented in subsection 1.5.2.2, revealed that the main objectives of haptic system (low viscosity in free motion and stiff response in case of hard contacts) cannot be ensured simultaneously by any of the presented methods. The results showed that good performances may be obtained either in free or in restricted motion only with the price of loosing the other case performances.

Next, the theoretical tools for analyzing the stability of the delayed systems in different configurations, as well as the physical limitations and tuning clues from a practical point of view were presented in Chapter 1. Since the PD control is the most common used controller in haptics and teleoperation, a complete study of stability, as well as the fragility of such controllers for systems affected by time-delays were presented. The study was made for fixed time-delays as well as for uncertain time-delays, modeled by the uniform and gamma with gap distributions. It is worth mentioning that, in the case of uncertain time-delays, a special attention is required for the behavior of the delay in order to choose more efficiently the controller's gains. The specific results for the haptic case, as well as some general examples were presented using a geometrical representation in order to give a clear overview about the stability region in the PD gains parameter space. Furthermore, such a geometric approach allowed giving an unitary treatment for the thesis.

The use of the Smith predictor was further analyzed in two distinct directions: stability regions in controller's gain parameters-space and stability in delay parameters-space. In the first case,

for fixed or uncertain time-delays, an allowable stability region for the tuning gains of the controller ($K_p,\ K_d$) was drawn, similar to the *simple* PD control. For the second case, once the tuning parameters of the PD controller ($K_p,\ K_d$) are set-up, a study of the variation dependency between the system's delay and the Smith predictor's delay was proposed. A specific analysis of the Smith predictor-based control in the case of uncertain delays was presented in detail starting from the case when the uncertainty is fixed, then uniform distribution and next gamma distribution with gap were analyzed, and finally, using a mean stability notion, the normal distribution case was presented. Illustrative examples from haptics are inserted during the presentation in order to point out the theory.

In both cases (*simple* PD and Smith predictor-based control) a special attention is required for the behavior of the uncertain delay, since the controller's parameters can be chosen more efficiently in terms of end user perception. As it was presented, depending on the considered distribution type, the choice of controller's gains can be more or less restrictive.

Once the stability limits are driven, the tuning must be done in order to maximize the end user perceptions in free and restricted motions. In this sense, some tuning clues inspired from our practical experience have been presented taking into account also the motors dynamics as well as the saturation limits.

Based on the analysis from Chapter 1 and with respect to the theoretical results from Chapter 2, two new approaches were proposed and tested on a three degree-of-freedom haptic system in order to improve the performances in terms of end user perception:

- The first method - Smith predictor with distance feedback, uses the information from the virtual environment regarding the distance until a possible collision as feedback in order to create an accurate model for the predictor. It is worth mentioning that the classic Smith predictor compensates the delay only in free motion, while for restricted motion the model is no longer accurate, resulting in an oscillating system. This method was tested from *simple* situations (free motion and restricted motion) to more complicated in which impacts with moving objects and tiny virtual boxes were considered under fixed and uncertain time-delays. In all situations, the Smith predictor with distance feedback approach provided the appropriate performances.
- The second method proposed - PD with gain scheduling, comes *naturally* since good performances may be obtained in one case using adequate gains with the price of completely loosing the other case performances. As mentioned, the method works *well* for *relatively* small communication delays (< 20 *ms*), while for bigger ones (> 25 *ms*) there appears a loss of performances, especially under fast or last moment changes.

Among the two proposed methods in their present form, the best results are obtained for the Smith predictor with distance feedback. The difference comes from the strategy of the methods since the first one acts like a predictor (as its name says), while the PD with gain scheduling acts based on the present actions without any explicit anticipation.

Perspectives

For further improvement of the Smith predictor-based control with distance feedback, a passivity observer or set-point modulation method may be added in order to provide an additional guarantee of stability in case of fast oscillations of the virtual environment or violent perturbations on the distance feedback.

Furthermore, this new strategy could easily extended for haptic systems with six degree-of-freedom and for more complicated virtual environments i.e. concave or non-regular geometric shapes.

The improvement perspectives for the PD with gain-scheduling approach are connected to the use of a more complex gain changing algorithm based on velocity, and time-delay values. More precisely, for a *high* velocity, the switching limit can be increased in order to change the gain faster, while for a smaller velocity the limit should be decreased in order to complete the updating process on both sides by the impact moment. As mentioned previously, this could be formalized also as a optimization problem in finding the optimum *limit* depending on velocity and time-delay in order to complete the updating process in the impact moment. Another point which can be added in order to decrease the *sticking* effect which appears when switching from free to restricted motion, is to take into account that once the user is moving in the opposite side of the wall, the gain on the haptic side may be increased before receiving the information from the virtual controller. Resuming, a dynamical *limit* for each axis depending on the velocity may reduce substantially the limitations of the method. It is worth mentioning that, with further improvements, also the PD with gain scheduling could provide better results.

In terms of applications, the next step may be the implementation of the control algorithms for collaborative systems. More precisely, in this work the haptic system was designed for only one human operator, while in future developments, the algorithms will be implemented for systems with more users interacting in the same virtual environment. Some ideas can be found in [36] in the context of a *simple* Smith predictor case study.

Next, the algorithms presented here can be also implemented for supervision applications used in teleoperation, systems which are also affected by time-delays resulting in similar problems to the ones dealt in this thesis. The super vision applications represent a virtual assistance for the human operator in completing difficult tasks. Such systems use a virtual replica or analyze in real time the remote environment where the slave robot is working and provide additional help in completing the tasks [43, 56].

The time-delays variations analyzed here are similar to the Internet® ones, and thus, future applications will be tested directly on Internet®. In order to improve the algorithms, packet-loss and network *crashes* will be taken into account and implemented in the control strategies. Such approaches are necessary when using the Internet® as a communication line.

Appendix A

Results' Analysis for 10ms time-delay

This appendix completes the presentation in Chapter 1, where some comparison between several methods is proposed. Two delay configurations have been considered $\tau_1 = \tau_2 = 10$ *ms* and $\tau_1 = \tau_2 = 50$ *ms*. The discussions concerning the first case study are given below. A constant human operator force $F_h = 7\,N$ for the restricted motion cases is considered. The Smith predictor case is not analyzed in this study.

The delays related to virtual reality simulations were not taken into account. The basic idea is to set up an optimal tuning for restricted motion in order to have a minimal position tracking error (close to the *ideal* case) and then to analyze it for free motion case. Next, the gains will be set up in order to obtain a reduced viscosity (near to the *ideal* case) and then the restricted motion case will be analyzed. Based on the *ideal* behavior, the gains were tuned for the two cases as follows:

- best performance for position tracking error in the case of restricted motion, figures A.1 and A.2 (free and restricted motion), with:
  - for the first four methods:

$$K_p = 3200, \qquad K_d = 100,$$

  - for the wave-scattering method:

$$K_p = 2500, \qquad K_d = 70, \qquad b = 0.2,$$

- best performance for viscosity in the case of free motion, figures 1.22 and 1.23 (free and restricted motion), with:

– for the first four methods:

$$K_p = 1000, \qquad K_d = 40,$$

– for the wave-scattering method:

$$K_p = 500, \qquad K_d = 60, \qquad b = 0.3,$$

Table A.1 summarizes the controller gains' values for each method and case.

Method	**Optimal - Restricted motion**		**Optimal - Free motion**	
	$K_p\ (N/m)$	$K_p\ (Ns/m)$	$K_p\ (N/m)$	$K_p\ (Ns/m)$
Classic Proportional-Derivative (PD)	3200	100	1000	40
Proportional-Derivative control with local dissipation				
Proportional-Derivative control with passivity observer				
Proportional-Derivative control with Passive Set-Position Modulation				
Wave-Scattering Transform	2500	70	500	60

TABLE A.1: Controller gains' values.

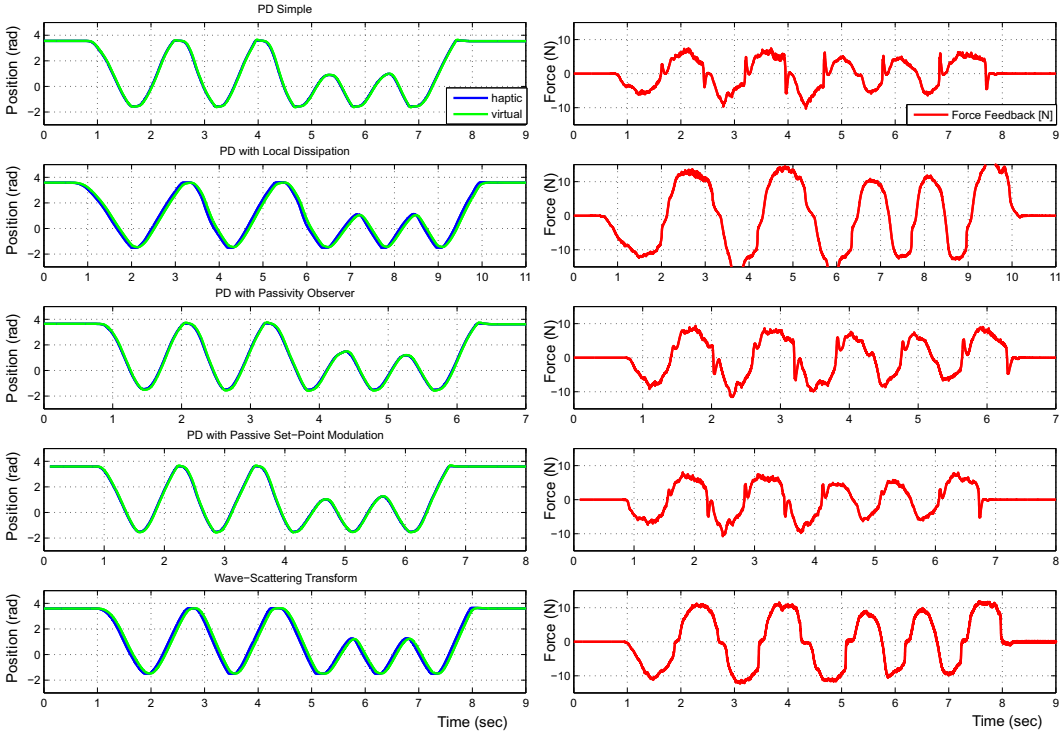


FIGURE A.1: Optimal error tracking - free motion, 10 ms delay.

In figure A.5 the maximum position tracking error and the average force feedback (measured at a speed of 8 rad/sec) is presented for each method and each case.

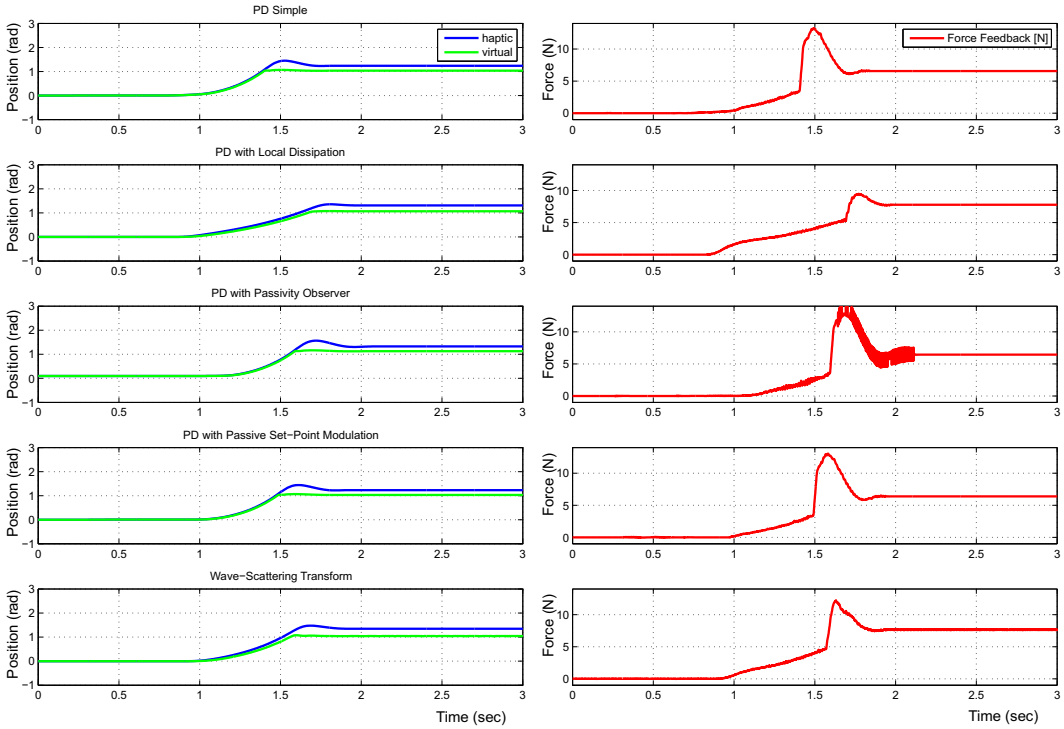


FIGURE A.2: Optimal error tracking - restricted motion, 10 ms delay.

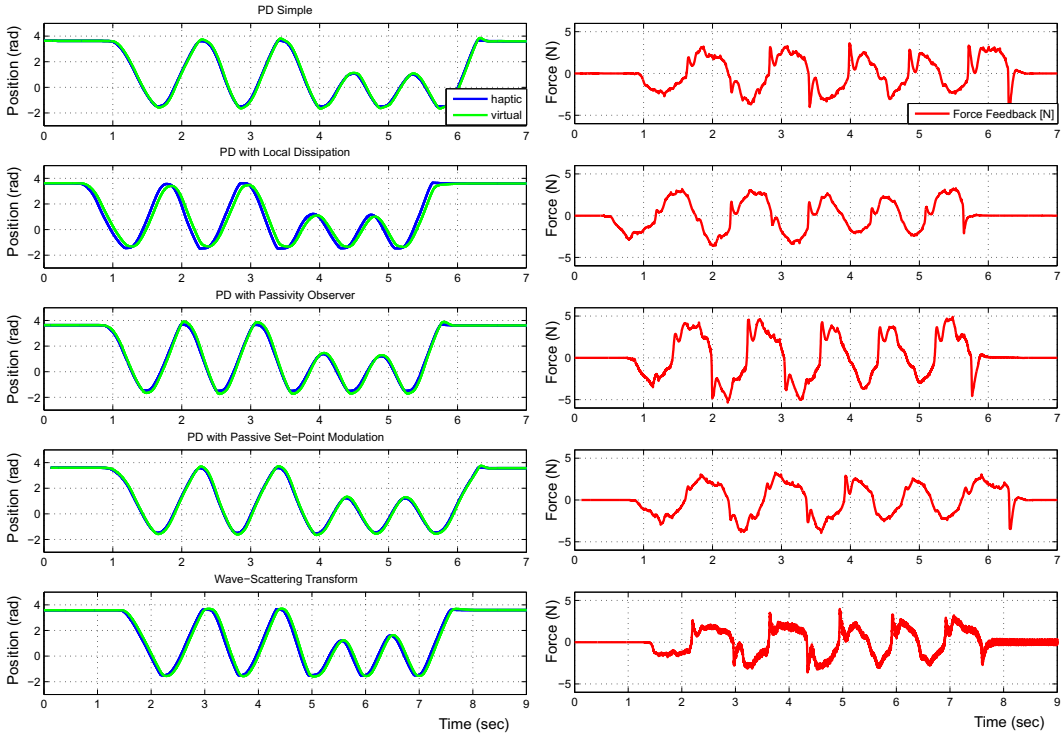


FIGURE A.3: Optimal viscosity effect - free motion, 10 ms delay.

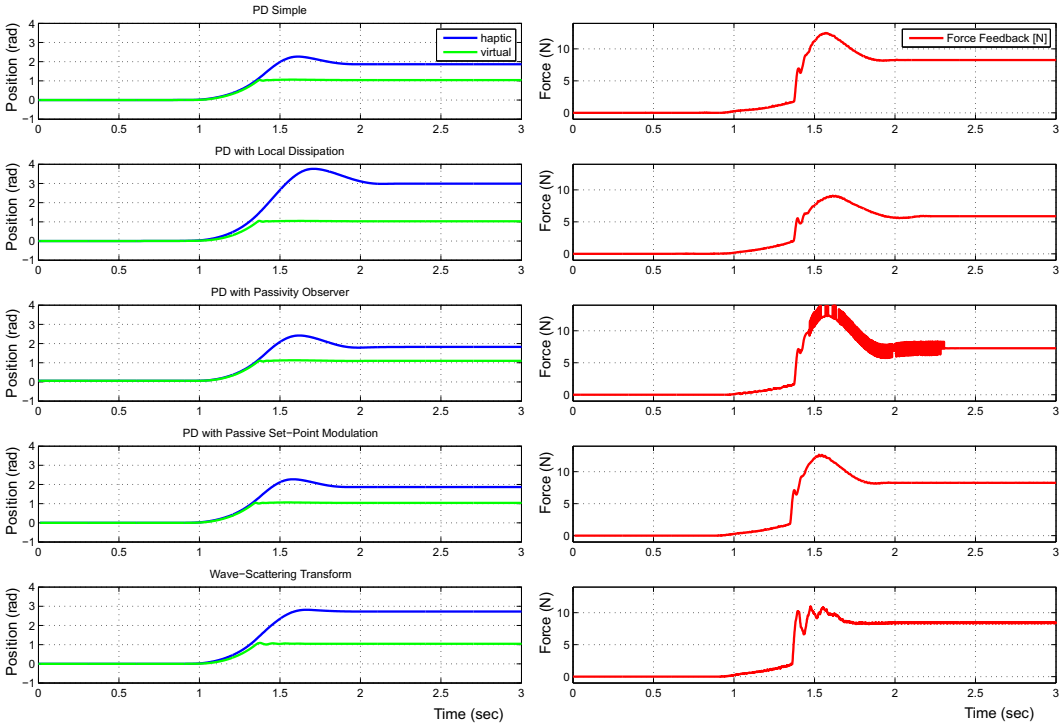


FIGURE A.4: Optimal viscosity effect - restricted motion, 10 ms delay.

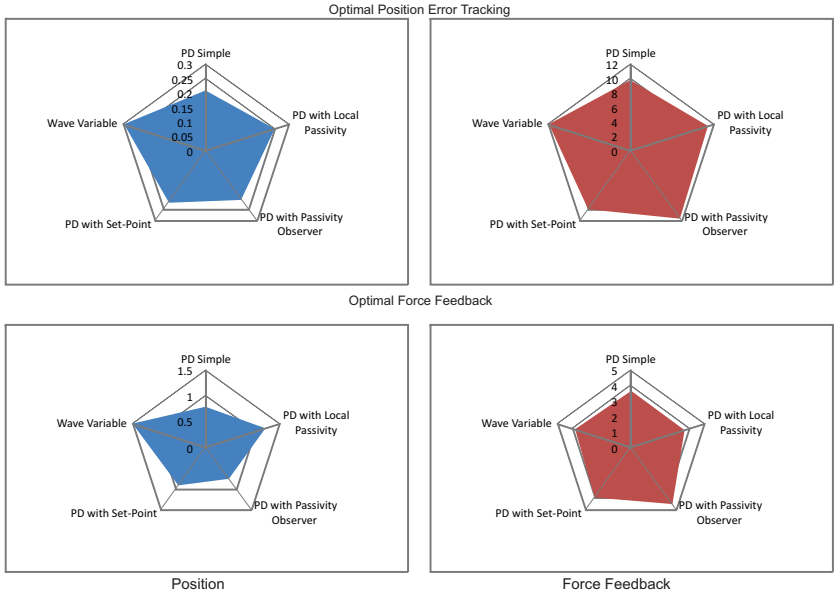


FIGURE A.5: Position tracking error and force feedback performance in the case of 50 ms delay.

According to the experimental results, the differences between the methods remain very slight. It is obvious, that for the first five methods the stability is assured, furthermore the passivity observer and the set-point methods provide an additional theoretical guarantee. The Smith predictor method is more sensible in terms of stability. On the other hand, none of these methods can provide high degree of transparency and small position tracking error in the same time. The compromise between transparency and position tracking error is obvious. The presence of time delays deteriorates the system's performances including disturbing effects like viscosity, which is directly linked to time delays.

For the first case - optimal tracking error, the best method in terms of position tracking error is the PD control with local dissipation with an error of 0.512 rad, but in free motion the viscosity effect is significant (17.2 N). The best method, from the transparency point of view, is the Smith Predictor, with a force feedback of 7.5 N, but in terms of position tracking error the result is not very good (0.832 rad).

The *best* compromise between transparency and position tracking error is assured by the PD control with set-point method with a position tracking error equal to 0.569 rad and a force feedback of 11.2 N.

For the second case - optimal viscosity effect, which is more significant in the case of haptics (the interest to have a low viscosity effect in free motion and a stiff response in case of hard contact is bigger than to have small position tracking error), the best performance, i.e. the smallest force feedback, is obtained for the PD control with set-point modulation method (1.9 N) and a position tracking error of 2.336 rad. The smallest position tracking error is assured by the PD control with local dissipation (2.271 rad) like in the first case.

The *best* compromise between transparency and position tracking error is assured by the classic PD control with a position tracking error equal to 2.298 rad and a force feedback of 2 N.

It is worth mentioning that some of the methods (wave variables in the second case for free and restricted motion, PD with passivity observer in first case for restricted motion and Smith predictor in the second case for restricted motion) induce significant noise on the responses.

Generally speaking, even if one is able to see small differences between the methods, the optimal tuning for one case will deteriorate the performance for the other one.

Appendix B

Stability and transparency for teleoperated systems

This section addresses the "trade-off' between transparency and stability of some specific bilateral teleoperation systems including communication time-delays. Using a geometric approach, a simple method will be derived for studying the *fragility* of the proposed controller for a general 4-channel architecture for bilateral teleoperation with time-delays such that the closed-loop stability as well as the transparency are guaranteed for the overall scheme.

B.1 Preliminaries

A teleoperated system indicates operation of a robot/machine at a distance. The first master-slave teleoperator was build in the mid 1940s by Goertz [54], and since then the systems have been improved over and over in accordance with the new control techniques and hardware. Roughly speaking, the goal of teleoperation systems is to replace the direct human manipulation in order to prevent accidents, contaminations (like radioactivity or dust, see [88]) or just to get in inaccessible places for humans, like, for example, the depth of the oceans [50]. A particular interest is granted to military and space applications [171, 192]. During the last years, more teleoperated robots are used in surgery due to the needs of a high level of precision and small incisions [182]. In figure B.1 an example of a teleoperated humanoid robot is presented [92].

The *ideal* teleoperation system must be perfectly *transparent* and must have *a real-time response*. Transparency can be defined as telepresence sense between the operator and the environment, see, for instance [76]. More precisely, in case of contact on the slave side, the master robot must reproduce exactly the same circumstances in order to provide to the human operator the feeling that he is directly realizing the contact and no intermediary system is used. The

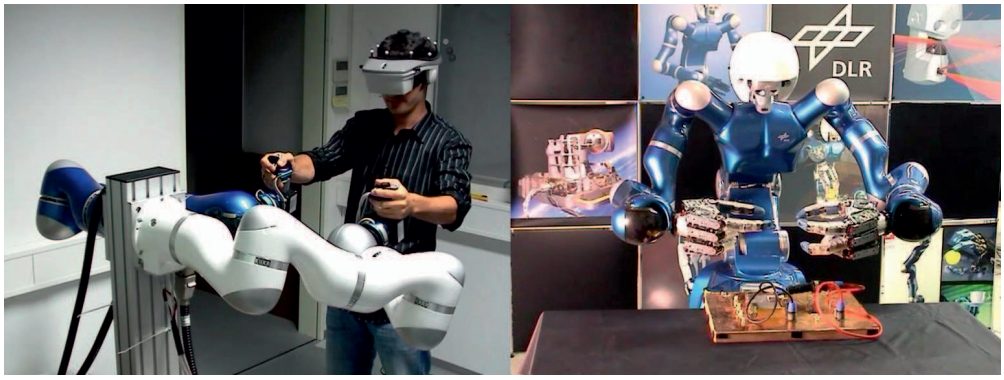


FIGURE B.1: Telepresent control of DLR's Rollin' JUSTIN

system stability must be assured for the free motion and contact cases and also the transitory state from one cases to another must be carefully analyzed because at this point the system is the most vulnerable from the stability point of view. These two goals - transparency and stability, are generally conflicting and always a trade-off is needed. To the best author's knowledge, there are two approaches for handling transparency: a correspondence between the master and slave position and forces (the so-called *kinematic* correspondence) [163, 194], or a match between the impedances of operator and environment [96], respectively.

An important factor which affects directly the stability and also the transparency are the time-delays. Modern teleoperated systems can have a long distance between master and slave, they can have different geographical locations. The communication medium introduces time delays that must be taken into account when the control law is elaborated.

For the sake of brevity, consider the block diagram of a general teleoperation system [67] as depicted in Figure B.2, where the concept of two port network is used to represent the interactions between the operator and the environment.

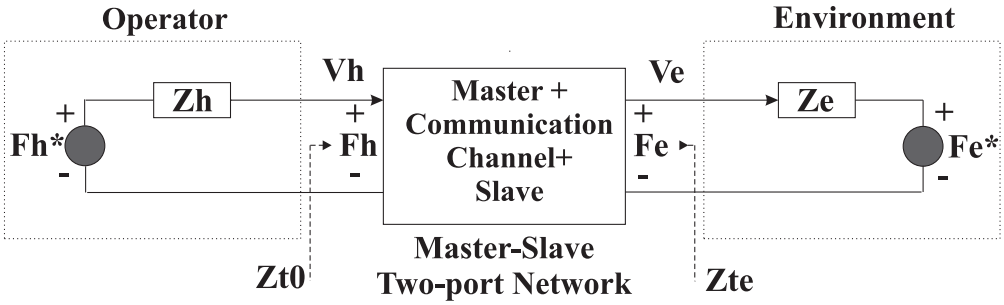


FIGURE B.2: Teleoperation Block Scheme System

The operator and the environment are assumed to be in contact with the master and slave, and they are modeled around their contact operating point in Laplace domain by the lumped under time-invariant (LTI) dynamics [63, 149] as follows:

$$F_h = F_h^* - Z_h V_h, \tag{B.1}$$

$$F_e = F_e^* + Z_e V_e, \tag{B.2}$$

where Z_h, Z_e, V_h, V_e, F_h, F_e, F_h^* and F_e^* are the master and slave impedances and velocities, the operator force on the master, the slave force on the environment, and the exogenous force inputs generated by the operator and the environment, respectively.

There exist several possibilities to represent the network matrices depending on the choice of input/output of the network: I/O, impedance Z, admittance hybrid H and inverse hybrid G as defined in [1, 69]:

$$\begin{bmatrix} F_h \\ F_e \end{bmatrix} = O_Z := ZI_Z = \begin{bmatrix} z_{11} & z_{12} \\ z_{21} & z_{22} \end{bmatrix} \begin{bmatrix} V_h \\ -V_e \end{bmatrix}, \tag{B.3}$$

$$\begin{bmatrix} V_h \\ -V_e \end{bmatrix} = O_Y := YI_Y = \begin{bmatrix} y_{11} & y_{12} \\ y_{21} & y_{22} \end{bmatrix} \begin{bmatrix} F_h \\ F_e \end{bmatrix}, \tag{B.4}$$

$$\begin{bmatrix} F_h \\ -V_e \end{bmatrix} = O_H := HI_H = \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{bmatrix} \begin{bmatrix} V_h \\ F_e \end{bmatrix}, \tag{B.5}$$

$$\begin{bmatrix} V_h \\ F_e \end{bmatrix} = O_G := GI_G = \begin{bmatrix} g_{11} & g_{12} \\ g_{21} & g_{22} \end{bmatrix} \begin{bmatrix} F_h \\ F_e \end{bmatrix}. \tag{B.6}$$

Here, F_h and $-V_e$ will be considered as inputs and V_h and F_e as outputs. In this case, equation (B.5) will describe the considered configuration, where h_{ij}, i, j = 1, 2 are functions of the master and slave dynamics and their corresponding control parameters.

According to [67], the conditions for kinematic correspondence $V_h \equiv V_e$, as well as for impedance matching $Z_{t0} \equiv Z_{te}$ or $Z_{te} \equiv Z_h$ must be examined. Consider Z_{t0} and Z_{te} defined as follows [67]:

$$Z_{te} := \frac{F_e}{-V_e}|_{F_h^*=0} = \frac{h_{11} + Z_h}{(h_{11}h_{22} - h_{12}h_{21}) + h_{22}Z_h}, \tag{B.7}$$

$$Z_{t0} := \frac{F_h}{V_h}|_{F_e^*=0} = \frac{h_{11} + (h_{11}h_{22} - h_{12}h_{21})Z_e}{1 + h_{22}Z_e}. \tag{B.8}$$

If the network parameters are not function of Z_h and Z_e, the impedance matching can be obtained if and only if [67]:

$$\begin{cases} h_{11} = h_{22} = 0 & \text{Impedance Matching} \\ h_{12} = h_{21} = -1 & \text{Conditions.} \end{cases} \tag{B.9}$$

Figure B.3 presents a general 4-channel teleoperation control system which incorporates communication delays $e^{-\tau s}$ and appropriate feedback compensators K_5F_e, K_6F_h, where:

$Z_m := M_m s$	M_m, M_s linear mass models of the
$Z_s := M_s s$	force actuated master and slave,
$K_m := B_m + k_m/s$	local position
$K_s := B_s + k_s/s$	controllers,
K_1, K_2, K_3 and K_4	rational control transfer functions.

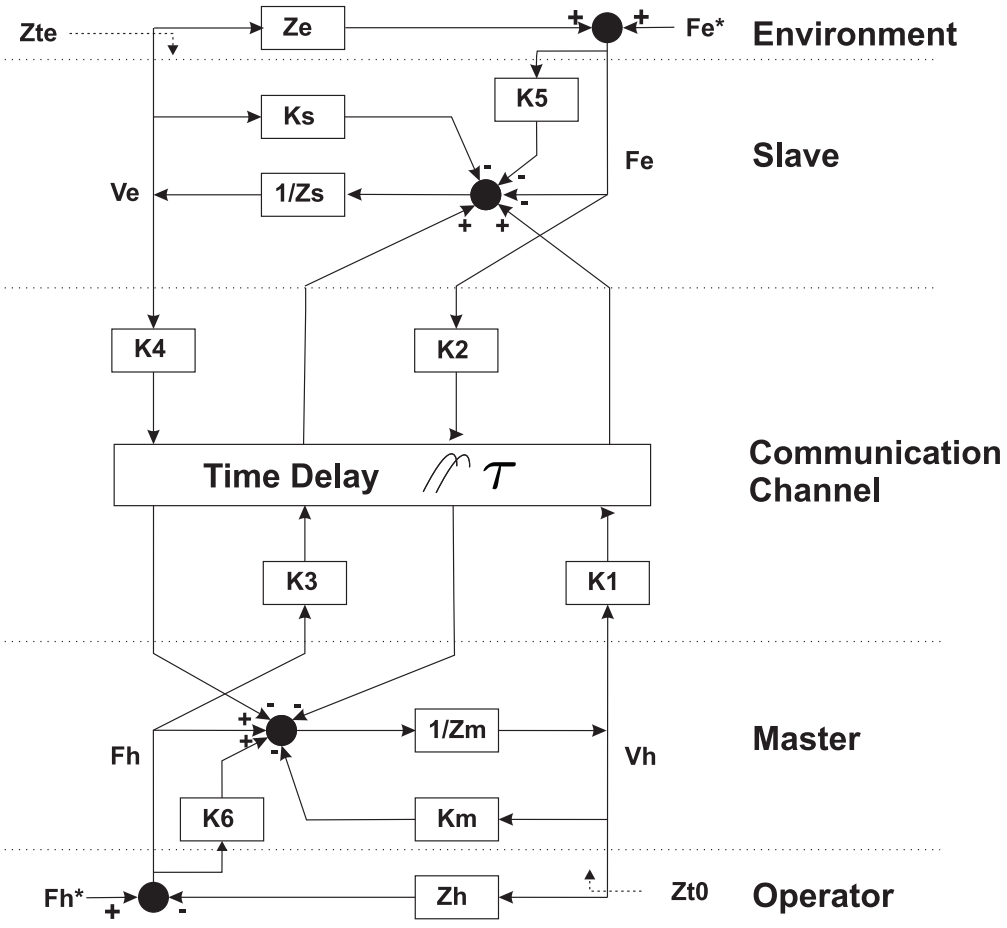


FIGURE B.3: Block diagram of a general bilateral controller used in teleoperation

According to [67], the closed-loop equations from Figure B.3 can be expressed as follows :

$$Z_{cm}V_h + K_4 e^{-\tau s}V_e = (1+K_6)F_h - K_2 e^{-\tau s}F_e, \tag{B.10}$$

$$K_1 e^{-\tau s}V_h - Z_{cs}V_e = -K_3 e^{-\tau s}F_h + (1+K_5)F_e, \tag{B.11}$$

with:

$$Z_{cm} := Z_m + K_m, \quad Z_{cs} := Z_s + K_s.$$

According to [96], by considering that $K_1 ... K_6$ are not functions of Z_h and Z_e, the transparency is achieved, if and only if the following conditions holds simultaneously:

$$\begin{cases} K_1 = Z_{cs} & \\ K_2 = 1 + K_6 & \text{Perfect} \\ K_3 = 1 + K_5 & \text{Transparency} \\ K_4 = -Z_{cm} & \text{Condition-set,} \end{cases} \tag{B.12}$$

and (K_2, K_3) $\neq$ (0,0) holds. This corresponds to the case when (K_5, K_6) $\neq$ (-1, -1).

Consider now the equations (B.10)-(B.11) and the transparency conditions (B.12). Then the characteristic equation in the free-of-delay case can be written as follows:

$$\Delta_0 = (K_2 Z_{cs} + K_3 Z_{cm})(Z_h + Z_e) = 0. \tag{B.13}$$

This case was presented and largely discussed by Salcudean et al. [163]. Now, in the case of communication delays, the characteristic equation rewrites as follows:

$$\Delta_\tau \quad = (Z_{cs} + K_3 Z_e)(Z_{cm} + K_2 Z_h) - (Z_{cs} - K_3 Z_h)(Z_{cm} - K_2 Z_e) e^{-2\tau s} = 0. \tag{B.14}$$

In our opinion, the notion of *controller fragility*[1] is more appropriate for such a study, see, for instance, [4, 85, 111]. Roughly speaking, the fragility describes the deterioration of closed-loop stability due to small variations of the controller parameters. The purpose is to detect non-fragile controllers by appropriate construction of the closed-loop stability regions in the corresponding controller parameter-space. A deeper discussion on the effects induced by the system's parameters on the (closed-loop) stability of delayed systems can be found in [116, 129]. A simple geometric argument, inspired by the ideas suggested in [123], will allow to conclude on the best controller's choice. It is worth mentioning that such an approach can be adapted to other controllers' configurations. However, in the sequel, only the "simple" gains case will be considered, that is:

$$(K_5, K_6) = (\alpha, \beta), \quad \text{with } (\alpha, \beta) \neq (-1, -1), \tag{B.15}$$

as mentioned above.

[1] Here, by *fragility*, it is simply understood the deterioration of closed-loop stability due to small variations of the controller parameters (see, for instance, [4, 85, 111] for further details on such topics).

B.2 Scalar gains

It is well known that the stability of a closed-loop linear system is given by the location of the zeros of the characteristic equation. Here Δ_τ from equation (B.14) can be rewritten as:

$$\Delta_\tau = Q(s,\alpha,\beta) + P(s,\alpha,\beta)e^{-2\tau s}, \tag{B.16}$$

where α and β are the controller parameters, τ is the communication delay and Q, P are defined as follows:

$$\begin{aligned} Q(s,\alpha,\beta) &= q_1 s^4 + q_2 s^4 + q_3 s^2 + q_4 s + q_5, \\ P(s,\alpha,\beta) &= p_1 s^4 + p_2 s^3 + p_3 s^2 + p_4 s + p_5, \end{aligned} \tag{B.17}$$

with:

$$\begin{aligned} q_i(\alpha,\beta) &= q_{i1} + q_{i2}\alpha + q_{i3}\beta + q_{i4}\alpha\beta, \\ p_i(\alpha,\beta) &= p_{i1} + p_{i2}\alpha + p_{i3}\beta + p_{i4}\alpha\beta, \end{aligned} \tag{B.18}$$

and q_i, p_i: $\mathbb{R}$ x $\mathbb{R} \to \mathbb{R}$, as in equation (B.14).

According to [115], a controller law (α^*, β^*) must be derived, and a positive value d such that the control law stabilizes the given system for any α and β satisfying:

$$\sqrt{(\alpha-\alpha^*)^2 + (\beta-\beta^*)^2} < d. \tag{B.19}$$

In such a case the controller (α^*, β^*) define the *best non-fragile controller* guaranteeing simultaneously robust stability in closed-loop and transparency in the sense defined by [96].

According to the continuity of zeros with respect to the system's parameters [98], the number of roots in the right half plane (RHP) can change only when some zeros appear and cross the imaginary axis ($j\mathbb{R}$). In such a frame, an useful concept is the *frequency crossing set* [123], defined as the set of all real positive frequencies ω for which there exist at least a pair (α, β) such that there exists at least one characteristic root on the imaginary axis:

$$H(j\omega;\alpha,\beta,\tau) = Q(j\omega;\alpha,\beta) + P(j\omega;\alpha,\beta)e^{-2\tau j\omega} = 0. \tag{B.20}$$

The following proposition can be written:

Proposition B.1. *For a given $\tau \in \mathbb{R}_+^*$ and $\omega \in \Omega \subset \mathbb{R}_+^*$ a crossing point (α, β) is given by the solutions of the following system:*

$$\begin{cases} \gamma_1^R\alpha + \gamma_2^R\beta + \gamma_3^R\alpha\beta &= -\gamma_4^R, \\ \gamma_1^I\alpha + \gamma_2^I\beta + \gamma_3^I\alpha\beta &= -\gamma_4^I. \end{cases} \tag{B.21}$$

where γ_i^R and γ_i^I, $\overline{i=1...4}$ are the corresponding coefficients for real and imaginary parts including the free term of the equation.

Proof. First, remark that the stability is given by the localization of the zeros of the closed-loop characteristic equation Δ_τ. Next, since the zeros are continuous with respect to the parameters α and β (see, for instance, [98]), this implies that the crossing points are characterized by the solutions of the following system:

$$\Re\left(\Delta_\tau|_{s=j\omega}\right) = 0, \tag{B.22}$$

$$\Im\left(\Delta_\tau|_{s=j\omega}\right) = 0, \tag{B.23}$$

which is precisely (B.21).

To guarantee the existence of α and $\beta \in \mathbb{R}$, the following constrains must be satisfied:

$$\alpha \neq -\gamma_2^R/\gamma_3^R, \quad \alpha \neq -\gamma_2^I/\gamma_3^I, \quad \beta \neq -\gamma_1^R/\gamma_3^R, \quad \beta \neq -\gamma_1^I/\gamma_3^I,$$

$$(\gamma_2^R\gamma_1^I+\gamma_3^R\gamma_4^I-\gamma_1^R\gamma_2^I-\gamma_4^R\gamma_1^I)^2-4(\gamma_3^R\gamma_1^I-\gamma_1^R\gamma_3^I)(\gamma_2^R\gamma_4^I-\gamma_4^R\gamma_2^I)\geqslant 0,$$

$$(\gamma_3^R\gamma_4^I+\gamma_1^R\gamma_2^I-\gamma_2^R\gamma_1^I-\gamma_4^R\gamma_3^I)^2-4(\gamma_3^R\gamma_2^I-\gamma_2^R\gamma_3^I)(\gamma_1^R\gamma_4^I-\gamma_4^R\gamma_1^I)\geqslant 0,$$

$$(\gamma_2^I\gamma_1^R+\gamma_3^I\gamma_4^R-\gamma_1^I\gamma_2^R-\gamma_4^I\gamma_1^R)^2-4(\gamma_3^I\gamma_1^R-\gamma_1^I\gamma_3^R)(\gamma_2^I\gamma_4^R-\gamma_4^I\gamma_2^R)\geqslant 0,$$

$$(\gamma_3^I\gamma_4^R+\gamma_1^I\gamma_2^R-\gamma_2^I\gamma_1^R-\gamma_4^I\gamma_3^R)^2-4(\gamma_3^I\gamma_2^R-\gamma_2^I\gamma_3^R)(\gamma_1^I\gamma_4^R-\gamma_4^I\gamma_1^I)\geqslant 0.$$

Proposition B.2. *Let $\alpha^*,\ \beta^* > 0$ be given. Let $\Omega_{\alpha^*,\beta^*}$ denotes the set of all frequencies $\omega > 0$ satisfying equation (B.14) for at least one pair of (α, β) in the rectangle $|\alpha| \leqslant |\alpha^*|$, $|\beta| \leqslant |\beta^*|$. Then $\Omega_{\alpha^*,\beta^*}$ consists of a finite number of intervals of finite length.*

Proof. Using the modulus, equation (B.14) implies that:

$$|(Z_{cs}+K_3Z_e)(Z_{cm}+K_2Z_h)|_{s=j\omega} = |(Z_{cs}-K_3Z_h)(Z_{cm}-K_2Z_e)|_{s=j\omega}.$$

Next, using the fact that $|\alpha| < |\alpha^*|$ and $|\beta| < |\beta^*|$, the above equality can be written as a polynomial inequality, implying that there exists always a finite number of real solutions and, as a consequence, $\Omega_{\alpha^*,\beta^*}$ consists of a finite number of intervals of finite length.

When ω varies within some interval Ω_l, (B.22) and (B.23) define a continuous curve (see, for instance, [115]). Denote this curve by $\mathcal{T}_l$ and consider the following decompositions:

$$R_0 + jI_0 = j\frac{\partial H(j\omega;\alpha,\beta,\tau)}{\partial s}\Big|_{s=j\omega},$$

$$R_1 + jI_1 = -\frac{\partial H(j\omega;\alpha,\beta,\tau)}{\partial \alpha}\Big|_{s=j\omega},$$

$$R_2 + jI_2 = -\frac{\partial H(j\omega;\alpha,\beta,\tau)}{\partial \beta}\Big|_{s=j\omega}.$$

The following notations will be used:

$$\mathcal{T} = \bigcup_{l=1}^{N} \mathcal{T}_l, \quad \mathcal{T}_l = \{(\alpha,\beta)|\omega\in\Omega_l\},$$

$$\overrightarrow{k(\omega)} = (\alpha(\omega),\beta(\omega))^T, \quad \overrightarrow{k^*} = (\alpha^*,\beta^*)^T.$$

In the same spirit as [115] the following proposition is written:

Proposition B.3. *The maximum parameter deviation from (α^*, β^*), without changing the number of unstable roots of the close-loop equation* (B.16) *can be expressed as follows:*

$$d = min_{\omega\in\Omega_f}\left\{\left\|\overrightarrow{k(\omega)} - \overrightarrow{k^*}\right\|\right\} \tag{B.24}$$

where Ω_f is the set of roots of the function $f : \mathbb{R}_+ \to \mathbb{R}$,

$$f(\omega) \triangleq \left(\overrightarrow{k(\omega)} - \overrightarrow{k^*}\right) \cdot \frac{\overrightarrow{dk(\omega)}}{d\omega}, \tag{B.25}$$

where "$\cdot$" means the inner product.

The explicit computation of the maximum parameter deviation d can be summarized by the following algorithm:

STEP 1: Compute the "degenerate" points of each curve $\mathcal{T}_l$:

- the roots of $R_1I_2 - R_2I_1 = 0$;
- multiple solutions of (B.14);

STEP 2: Compute the set of Ω_f defined by Proposition B.3 (i.e. the roots of equation $f(\omega) = 0$, where f is given by (B.25));

STEP 3: The corresponding maximum parameter deviation d_l given by (B.24).

Since this is the case of controller gains, it seems that the simplest choice is to take: $K_5 = \beta$ and $K_6 = \alpha$.

Considering now equations (B.12) and (B.15), then the control law becomes:

$$\begin{aligned} K_1 &= Z_{cs}, & K_4 &= -Z_{cm}, \\ K_2 &= 1+\alpha, & K_3 &= 1+\beta, \end{aligned} \tag{B.26}$$

with:

$$\begin{aligned} Z_{cm} &= a_{cm} + b_{cm}s + \tfrac{c_{cm}}{s}, \\ Z_{cs} &= a_{cs} + b_{cs}s + \tfrac{c_{cs}}{s}, \\ Z_e &= a_e + b_e s + \tfrac{c_e}{s}, \\ Z_h &= a_h + b_h s + \tfrac{c_h}{s}, \\ K_2 &= (1+\alpha), \\ K_3 &= (1+\beta), \end{aligned} \tag{B.27}$$

where $a_{cm}(a_{cs},\ a_e,\ a_h)$, $b_{cm}(b_{cs},\ b_e,\ b_h)$ and $c_{cm}(c_{cs},\ c_e,\ c_h)$ are constant, specific to the given system and environment. For a given τ and for a set of positive frequencies, the stability zone given by controller parameters $(\alpha,\ \beta)$ can be represented. Based on this, it can be decided if a given controller is *far or not* from the boundary of the stability regions, as pointed out by some of the examples proposed in the sequel.

B.3 Illustrative Examples

In this section four numerical examples will be presented. A significant simplification of the environment dynamics can be obtained using linear impedance model.The dynamic interaction can be described by the convenient and well-studied linear system model (impedance causality):

$$M_e\ddot{x} + B_e\dot{x} + C_e x = -F,$$

where: M_e represents a positive environment mass or inertia, B_e is the environment damping, and finally C_e is the environment stiffness.

The operator's hand is assumed to be a constant-mass spring and damper system with impedance model, see, for instance, [84]:

$$Z_h = 70 + 0.5s + \frac{2000}{s}. \tag{B.28}$$

According to [162], the dynamic model for master and slave can be expressed as:

$$Z_m = Z_s = 0.62s, \tag{B.29}$$

and the local controller K_m and K_s chosen as:

$$K_m = K_s = 10 + 0.1/s. \tag{B.30}$$

Example B.1. *Consider a soft environment impedance:*

$$Z_e = 100 + 10s + \frac{200}{s}. \tag{B.31}$$

It appears that there exist two solutions for α and β. In Figure B.4, the stability region for $\alpha^{(1)}$, $\beta^{(1)}$ and $\alpha^{(2)}$, $\beta^{(2)}$ is presented. The stability zones correspond to frequency $\omega \in [0, 100]$.

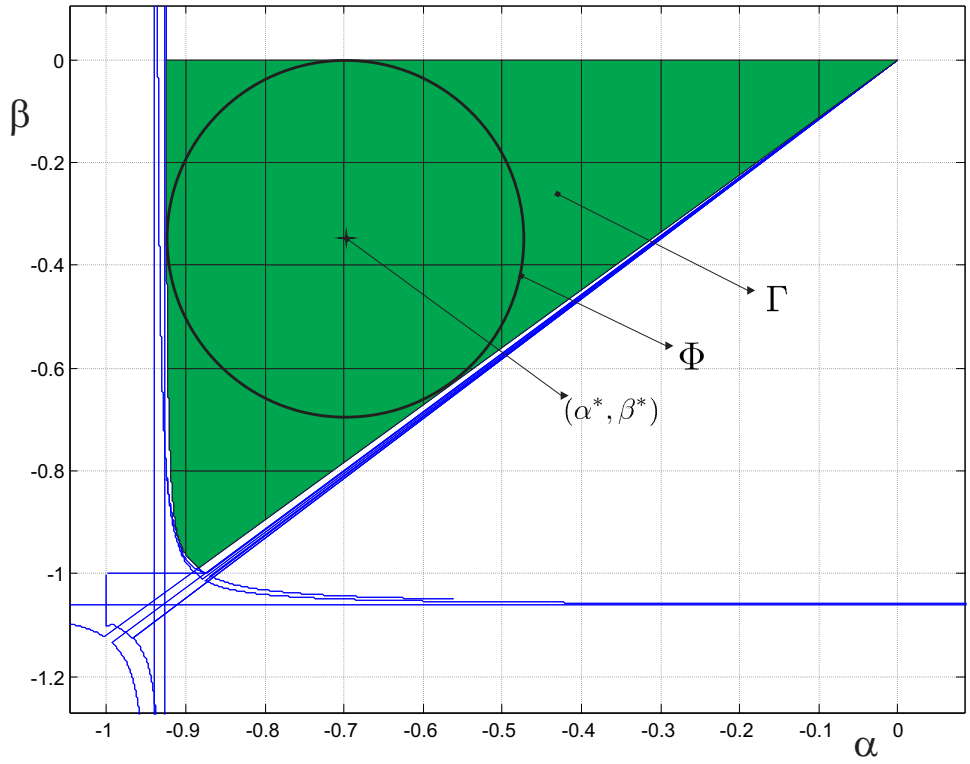


FIGURE B.4: Stability area Γ for α and β - soft environment, α^* = -0.683 , β^* = -0.352 , d = 0.463, with: Φ representing the non-fragile disc centered in (α^*, β^*) and having the radius d.

Example B.2. *Consider now a stiff environment [162]:*

$$Z_e = 10000 + 1000s + \frac{200000}{s}. \tag{B.32}$$

Similarly with the previous example, Figure B.5 depicts the stability regions for $\alpha^{(1)}$, $\beta^{(1)}$ and $\alpha^{(2)}$, $\beta^{(2)}$ corresponding to the stiff environment.

Example B.3. *Next, as suggested by [67] consider a different local controller K_m and K_s:*

$$K_m = K_s = 80 + 1600/s. \tag{B.33}$$

Figure B.6, presents the stability regions for $\alpha^{(1)}$, $\beta^{(1)}$ and $\alpha^{(2)}$, $\beta^{(2)}$ with the new local controllers and a soft environment.

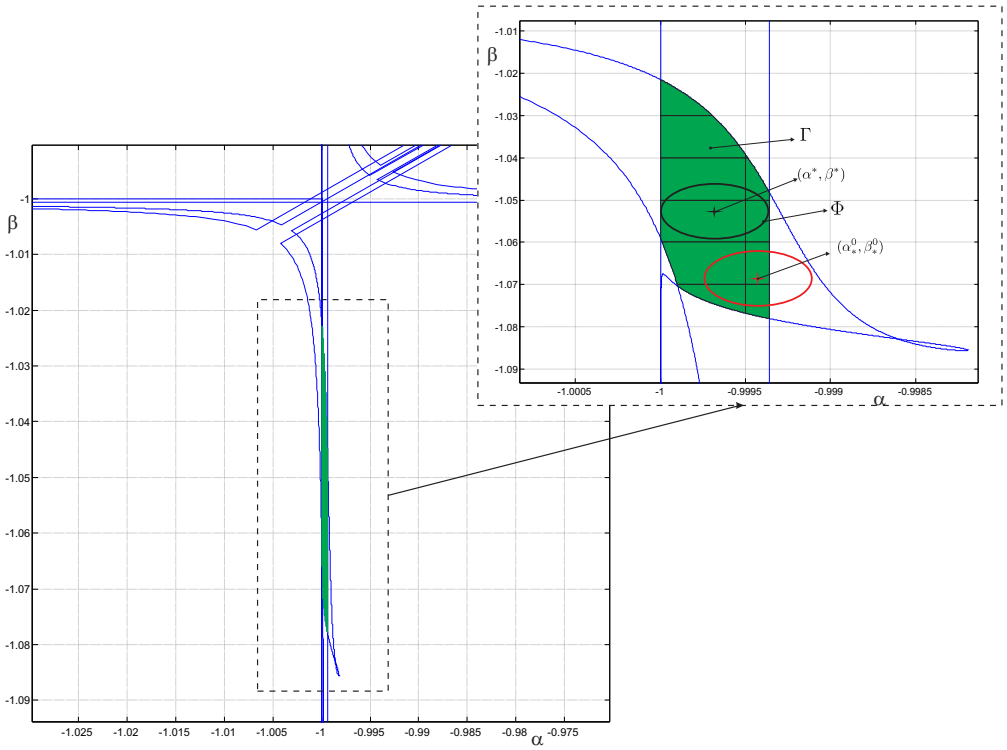


FIGURE B.5: Stability area Γ for α and β - stiff environment, α^* = -0.99955 , β^* = -1.0543 , d = 0.00064, with: Φ representing the non-fragile disc centered in (α^*, β^*) and having the radius d.

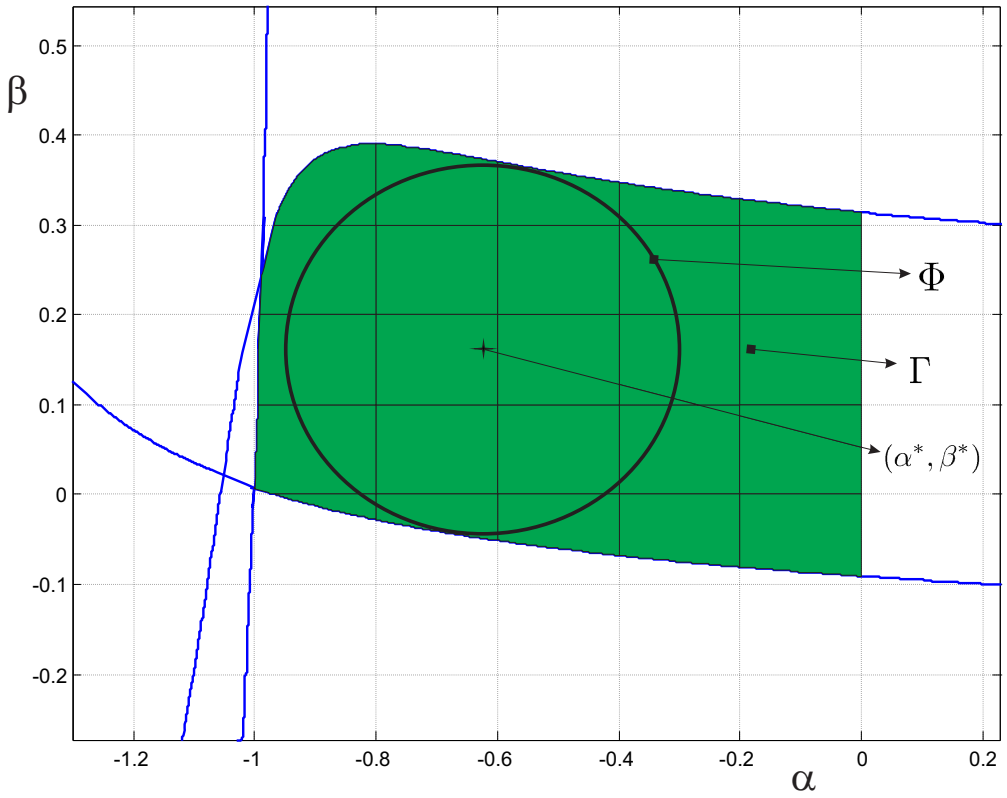


FIGURE B.6: Stability area Γ for α and β - soft environment, α^* = 0.183, β^* = -0.482, d = 0.55, with: Φ representing the non-fragile disc centered in (α^*, β^*) and having the radius d.

Example B.4. *Finally, a soft environment and two identical PHANTOM OMNI (for more details please refer to [165]) haptics devices (master and slave) were considered. According to [172], the PHANTOM OMNI is modeled as a mass-damping-spring system as follows:*

$$G(s) = \frac{1}{ms^2 + bs + k}, \tag{B.34}$$

with $m = 50g$, $b = 3Ns/m$ *and* $k = 20N/m$.

In Figure B.7, the stability regions for $\alpha^{(1)}$, $\beta^{(1)}$ *and* $\alpha^{(2)}$, $\beta^{(2)}$ *are drawn, corresponding to the configuration mentioned above.*

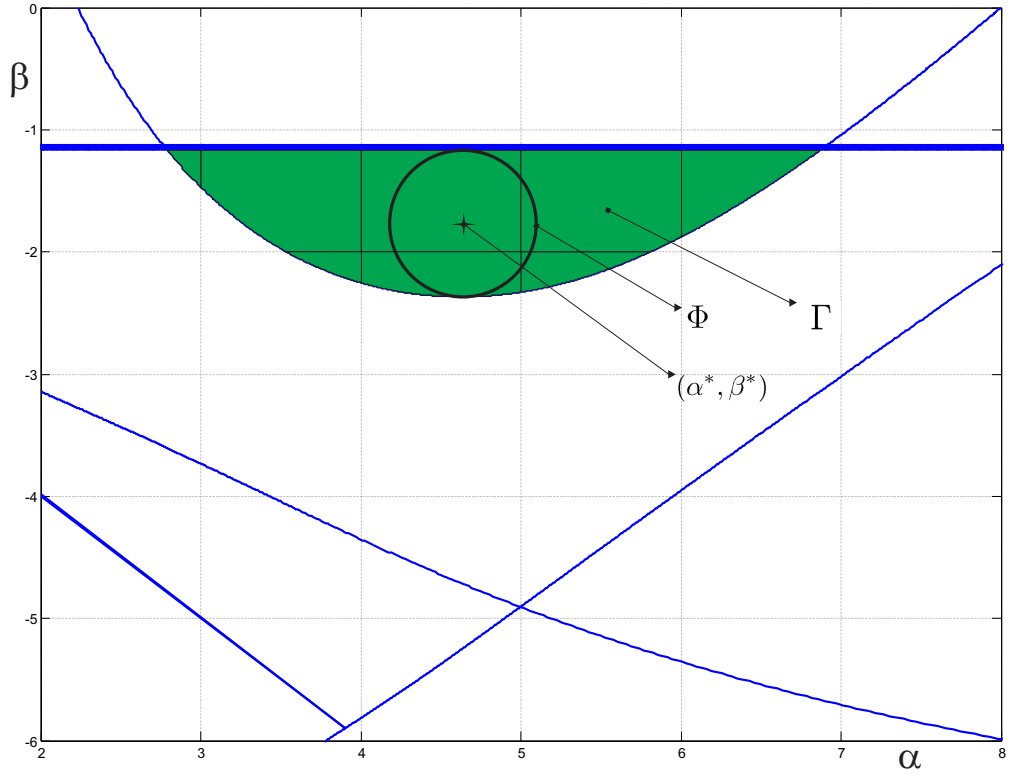


FIGURE B.7: Stability area Γ for α and β (PHANTOM OMNI master and slave and soft environment) α^* = 4.7415, β^* = -1.712, d = 0.935. Φ represents the non-fragile disc centered in (α^*, β^*) and having the radius d.

Remark B.4. It appears that in most of the examples considered (in particular, the first, the third and the last one) there is a large choice of non-fragile controllers and here only the *"best" non-fragile ones* were chosen to be represented. However, the second example proves the existence of some practical situations when corresponding stabilizing controllers are *fragile*, i.e. extremely sensitive to perturbations on the parameters.

More precisely, for any control parameter (α^*, β^*) chosen inside the *green* (stable) zone Γ, there exists a perturbation ($\Delta\alpha$, $\Delta\beta$) satisfying any constraint :

$$\sqrt{\Delta\alpha^2 + \Delta\beta^2} > 0.00064,$$

leading to an unstable closed-loop behavior.

In general, for bilateral teleoperation systems it is very important to have a high level of transparency. Furthermore, for end-users this is the most important characteristic. Next, in order to obtain good performances for closed-loop systems, the stability must be carefully analyzed.

The method proposed here handles both concepts in the corresponding controller parameter space. Furthermore, the choice of non-fragile controller is proposed by using some simple geometric arguments.

Appendix C

Reduction of the stability conditions

Since $(1+PC)^{-1}$ is a stable transfer function, from (2.13) it is worth mentioning that the feedback system is stable *if and only if* the following two equations do not have zeros in $\mathbb{C}_+$:

$$1+G(s)\left(\frac{1-e^{-\tau s}}{s}\right)=0, \quad \text{where} \quad G(s)=\frac{K_p+K_d s}{ms+b}, \tag{C.1}$$

$$1+T(s)\ e^{-\tau s}=0, \quad \text{where} \quad T(s)=\frac{K_p+K_d s}{s(ms+b)+K_p+K_d s}. \tag{C.2}$$

Now define:

$$K:=\frac{K_p}{b}, \quad \tau_c:=\frac{K_d}{K_p}, \quad \tau_p:=\frac{m}{b},$$

then $G(s)$ and $T(s)$ can be re-written as:

$$G(s)=K\,\frac{1+\tau_c s}{1+\tau_p s}, \tag{C.3}$$

$$T(s)=\frac{K(1+\tau_c s)}{\tau_p s^2+(1+\tau_c K)s+K}. \tag{C.4}$$

Further, a frequency normalization can be made:

$$\widehat{s}=\tau_p s, \tag{C.5}$$

and introduce new definitions:

$$L:=\frac{1}{K\tau_p}=\frac{b^2}{m\,K_p}, \quad \alpha:=\frac{\tau_c}{\tau_p}=\frac{b\,K_d}{m\,K_p}, \quad h:=\frac{\tau}{\tau_p}=\frac{(\tau_1+\tau_2)\,b}{2\,m}, \tag{C.6}$$

so that the characteristic equations (C.1) and (C.2) can be re-written as:

$$1+\frac{1}{L}\,\frac{(1+\alpha\widehat{s})}{(1+\widehat{s})}\,\left(\frac{1-e^{-h\widehat{s}}}{\widehat{s}}\right)=0, \tag{C.7}$$

$$1+\frac{(1+\alpha\widehat{s})}{(L\widehat{s}^2+(L+\alpha)\widehat{s}+1)}\,e^{-h\widehat{s}}=0. \tag{C.8}$$

The next step is to find the controller parameters L and α (which define K_p and K_d), as functions of h, that place all the roots of (C.7) and (C.8) in $\mathbb{C}_-$. In what follows, without any lack of generality, only the case where K_p and K_d are positive, i.e., $L>0$ and $\alpha>0$ is considered. It is worth mentioning that, in practice, such a situation occurs in most cases.

Analysis of stability conditions of transfer functions (C.7) and (C.8) are based on Nyquist Stability Criterion. Let us consider (C.7) first. Since $|e^{-jh\omega}|=1$ for all $\omega\in\mathbb{R}$, the phase of $(1-e^{-jh\omega})$ is between $+\pi/2$ and $-\pi/2$ for all $\omega\geq 0$ and:

$$\lim_{\omega\searrow 0^+}\angle(1-e^{-jh\omega})=+\frac{\pi}{2}.$$

Therefore,

$$0\leq\angle f(j\omega)\leq-\pi \qquad \forall\ \omega\in\mathbb{R} \qquad \text{where} \qquad f(\widehat{s}):=\frac{1-e^{-h\widehat{s}}}{\widehat{s}}. \tag{C.9}$$

This means that if $\alpha>1$, the phase of $\frac{(1+j\alpha\omega)}{(1+j\omega)}f(j\omega)$ is always strictly grater than $(-\pi)$ for all $\omega\geq 0$. Thus, all the roots of (C.7) are in $\mathbb{C}_-$ when $\alpha>1$, independent of L and h. Furthermore, when $\alpha=1$ the equation (C.7) reduces to

$$1+\frac{1}{L}f(\widehat{s})=0.$$

Note that whenever $\angle f(j\omega)=-\pi$ the following condition holds $|f(j\omega)|=0$. This fact, together with (C.9), implies that when $\alpha=1$ all the roots of (C.7) are in $\mathbb{C}_-$, independent of L and h. In conclusion, the analysis of (C.7) becomes interesting only if $\alpha<1$. In this case, all the roots of (C.7) are in $\mathbb{C}_-$ if and only if the following condition is met:

$$L>\frac{2(1-\alpha)}{\omega_p^2+1}, \tag{C.10}$$

where ω_p is the smallest $\omega>0$ satisfying:

$$\tan^{-1}(\alpha\omega)-\tan^{-1}(\omega)-\frac{h\omega}{2}=-\pi. \tag{C.11}$$

The condition (C.10) gives an allowable region in the (α,L)-plane for all the roots of (C.7) to be in $\mathbb{C}_-$ when $\alpha<1$. Since,

$$f(j\omega)=\frac{1-e^{-j\omega h}}{j\omega}=\frac{sin(\omega h)}{\omega}-j\frac{(1-cos(\omega h))}{\omega}$$

the following identity is derived, used in (C.11):

$$\angle f(j\omega) = tan^{-1}\left(\frac{cos(\omega h)-1}{sin(\omega h)}\right) = -\frac{h\omega}{2}, \qquad \forall\ \omega \in [0\ , \frac{2\pi}{h}],$$

by using half-angle formulas, followed by simplification using the trigonometric identity $cos^2(\omega h/2) = 1 - sin^2(\omega h/2)$.

Re-arranging the equation (C.11) for $\alpha < 1$ as:

$$\pi - \left(\tan^{-1}(\omega_p) - \tan^{-1}(\alpha\omega_p)\right) = \frac{h\omega_p}{2}. \tag{C.12}$$

It is easy to show that:

$$|f(j\omega_p)| = \frac{\sin(h\omega_p/2)}{\omega_p/2} = \frac{2(1-\alpha)}{\sqrt{(1-\alpha)^2\omega_p^2 + (1+\alpha\omega_p^2)^2}}.$$

Using this identity, after algebraic manipulations and for $\alpha < 1$, (C.10) is now equivalent to:

$$L > \frac{2(1-\alpha)}{\omega_p^2+1}, \tag{C.13}$$

where ω_p is determined from (C.12).

Now consider (C.8). The cross-over frequency $\omega_c > 0$ where:

$$\left|\frac{1+j\alpha\omega_c}{1-L\omega_c^2 + j(L+\alpha)\omega_c}\right| = 1$$

can be found as the feasible root of:

$$L^2\omega_c^2\left(\omega_c^2 + 1 - \frac{2(1-\alpha)}{L}\right) = 0.$$

Clearly, this has a non-zero real solution if an only if:

$$2(1-\alpha) > L \tag{C.14}$$

in which case:

$$\omega_c = \sqrt{\frac{2(1-\alpha)}{L} - 1}. \tag{C.15}$$

This means that if (C.14) is not satisfied, then $|T(j\omega)|$ is a uniformly decreasing function with $T(0) = 1 = \|T\|_\infty$ which, by the small gain theorem, implies that all the roots of (C.8) are in $\mathbb{C}_-$ independent of h. Since ω_p is a positive real number, the condition (C.13) also holds irregardless of delay value h when (C.14) is not satisfied. To complete the analysis, now assume (C.14) is satisfied and ω_c is as defined by (C.15). In this case, by the Nyquist criterion, all the roots of

(C.8) are in $\mathbb{C}_-$ if and only if:

$$\tan^{-1}(\alpha\omega_c) - atan2\left[(L+\alpha)\omega_c,\ 1-L\omega_c^2\right] - h\omega_c > -\pi, \tag{C.16}$$

where atan2(y,x)=Pr arg(x+*i*y)=Arg(x+*i*y).

To recapitulate, with the parameter definitions of (C.6), the feedback system described by (2.11) is stable independent of h if $\alpha \geq 1$ and it is stable depending on h if $2(1-\alpha) > L > 0$. For every fixed $h > 0$ the region of delay-dependent stabilizing $\{(\alpha, L) : 2(1-\alpha) > L > 0\}$ is determined from the intersection of the conditions (C.10) and (C.16).

Since (C.15) implies:

$$L = \frac{2(1-\alpha)}{1+\omega_c^2}$$

for $\alpha < 0$ and $2(1-\alpha) > L$, the condition (C.13) can be re-written as:

$$\omega_p > \omega_c. \tag{C.17}$$

Let us now re-consider (C.16). Using the notation $L = 2(1-\alpha)/(1+\omega_c^2)$, then, after simple algebraic manipulations, it is easy to see that:

$$\begin{aligned}\tan^{-1}(\alpha\omega_c) - atan2\left[(L+\alpha)\omega_c,\ 1-L\omega_c^2\right] &= -\tan^{-1}\left(\frac{2(1-\alpha)\omega_c(1+\alpha\omega_c^2)}{(1+\alpha\omega_c^2)^2-(2(1-\alpha)\omega_c)^2}\right)\\ &= -2\tan^{-1}\left(\frac{(1-\alpha)\omega_c}{(1+\alpha\omega_c^2)}\right) = -2\left(\tan^{-1}(\omega_c) - \tan^{-1}(\alpha\omega_c)\right).\end{aligned}$$

Thus the condition (C.16) is equivalent to:

$$\frac{\pi - 2\left(\tan^{-1}(\omega_c) - \tan^{-1}(\alpha\omega_c)\right)}{\omega_c} > h. \tag{C.18}$$

Recall from (C.17) and (C.12) that ω_c is restricted to satisfy $\omega_p > \omega_c$, where ω_p is defined from:

$$\frac{2\left(\pi - \left(\tan^{-1}(\omega_p) - \tan^{-1}(\alpha\omega_p)\right)\right)}{\omega_p} = h. \tag{C.19}$$

Resuming, the system is stable independent of delay h if $\alpha \geq 1$; or if $\alpha < 1$ and $L > 2(1-\alpha)$. Furthermore, the analysis for the case $\alpha < 1$ and $2(1-\alpha) > L > 0$ reduces to the following. Define:

$$\begin{aligned} g_c(x) &= \frac{\pi - 2\left(\tan^{-1}(x) - \tan^{-1}(\alpha x)\right)}{x},\\ g_p(x) &= \frac{2\left(\pi - \left(\tan^{-1}(x) - \tan^{-1}(\alpha x)\right)\right)}{x}. \end{aligned}$$

Clearly, g_p and g_c are uniformly decreasing functions and $g_p(x) > g_c(x)$ for all $x > 0$. So, if ω_p is defined as the solution of the equation $g_p(x) = h$ and ω_o as the solution of the equation $g_c(x) = h$, then $\omega_o < \omega_p$ and hence, for $\alpha < 1$ and $2(1-\alpha) > L > 0$, the feedback system shown in Figure 2.2 is stable if and only if $\omega_c < \omega_o$, which is equivalent to:

$$L > \frac{2(1-\alpha)}{1+\omega_o^2}, \qquad \text{where} \quad \omega_o > 0 \ \text{ is the solution of } \quad g_c(x) = h\,. \tag{C.20}$$

Appendix D

PD-like Controller

In [136] a *PD-like* controller is proposed, having the block scheme presented in Figure D.1.

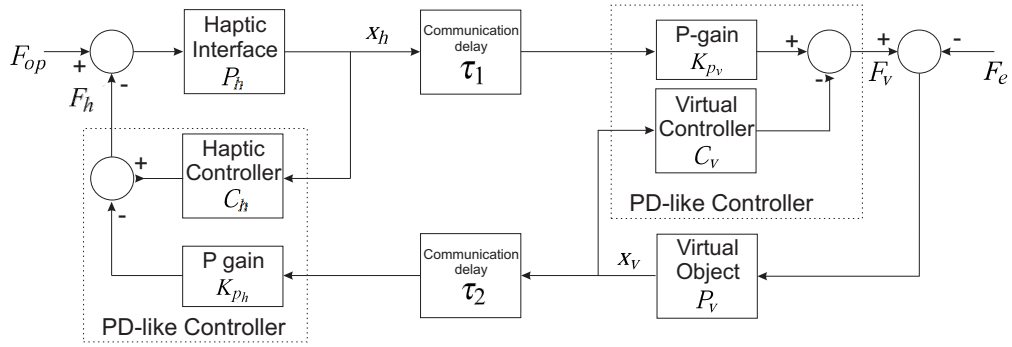


FIGURE D.1: Bilateral Haptic System using a *PD-like* Controller.

Here, instead of using both - position and velocity errors, only the position error will be used in the controller in order to guarantee the passivity of the system. The velocity of the haptic interface/virtual object will be used with the D-gain for computing the commands for the haptic and virtual side, respectively. With this assumption, equations (2.4)-(2.5) are rewritten as follows:

$$F_h(t) = \underbrace{K_{d_h}\dot{x}_h(t)}_{\text{D-action}} \underbrace{+K_{p_h}(x_h(t) - x_v(t-\tau_2))}_{\text{delayed P-action}}, \tag{D.1}$$

$$F_v(t) = \underbrace{-K_{d_v}\dot{x}_v(t)}_{\text{D-action}} \underbrace{+K_{p_v}(x_h(t-\tau_1) - x_v(t))}_{\text{delayed P-action}}, \tag{D.2}$$

Next, considering (2.8)-(2.10), equations (2.6)-(2.7) become:

$$X_h(s) = P_h(s)\left(F_{op}(s) - C_h(s)X_h(s) - K_{p_h}e^{-\tau_2 s}X_v(s)\right), \tag{D.3}$$

$$X_v(s) = P_v(s)\left(-F_e(s) - C_v(s)X_v(s) + K_{p_v}e^{-\tau_1 s}X_h(s)\right). \tag{D.4}$$

Rearranging (D.3)-(D.4), it follows:

$$\begin{bmatrix} 1+P_h(s)C_h(s) & -K_{p_h}P_h(s)e^{-\tau_2 s} \\ -P_v(s)K_{p_v}e^{-\tau_1 s} & 1+P_v(s)C_v(s) \end{bmatrix} \begin{bmatrix} X_h(s) \\ X_v(s) \end{bmatrix} = \begin{bmatrix} P_h(s)F_{op}(s) \\ -P_v(s)F_e(s) \end{bmatrix}. \tag{D.5}$$

Therefore, with the definitions (2.8)-(2.10), the new characteristic equation of the feedback system becomes:

$$(1+P(s)C(s))^2 - K_p^2 P(s)^2 e^{-(\tau_1+\tau_2)s} = 0, \tag{D.6}$$

which is equivalent to:

$$(1+P(s)C(s)+K_pP(s)e^{-\tau s})(1+P(s)C(s)-K_pP(s)e^{-\tau s}) = 1+K_p\, s\, P(s)\left(\frac{K_d}{K_p}+\frac{1}{s}\pm\frac{e^{-\tau s}}{s}\right)=0. \tag{D.7}$$

with $\tau = \frac{\tau_1+\tau_2}{2}$.

Since $(K_p sP(s))$ is positive real, in order to guarantee the stability, it is needed to guarantee:

$$\Re\left(\frac{K_d}{K_p}+\frac{1\pm e^{-\tau s}}{s}\right) > 0, \quad \forall s \in \bar{\mathbb{C}}_+.$$

Knowing that:

$$Re\left\{\frac{K_d}{K_p}+\frac{1+e^{-j\tau\omega}}{j\omega}\right\} = \frac{K_d}{K_p} - \frac{\sin(\tau\omega)}{\omega} \geq \frac{K_d}{K_p} - \tau$$

and:

$$\left|\frac{1-e^{-\tau j\omega}}{j\omega}\right| \leq \tau, \quad \forall \omega \in \mathbb{R}_+,$$

the stability is guaranteed if the following condition holds:

$$\frac{K_d}{K_p} > \tau \quad \Leftrightarrow \quad K_d > K_p\tau. \tag{D.8}$$

The result obtained in [136], by using a different argument:

$$K_{d_h}\, K_{d_v} > K_{p_h}K_{p_v}\tau_1\tau_2,$$

is exactly the same with (D.8), under the assumption (2.9) and $\tau_1 = \tau_2 = \tau$.

Appendix E

Fragility analysis of PD controllers

Let us define:

$$k_{p_0} := |k_p^* - k_p(0)|$$

In table E.1, the corresponding results obtained for example 2.1 are summarized after applying the proposed algorithm in order to analyze the fragility for the controller $(k_p^*, k_d^*) = (3.25, 1.65)$.

Frequency	$d_{\mathcal{T}_l}$	$k_{d\infty}$	k_{p_0}	$\min\{d_T, k_{d\infty}, k_{p_0}\}$
$\omega_1 = 0.9017$	2.3459			
$\omega_2 = 2.7292$	23.0540			
$\omega_3 = 2.8228$	23.0158			
$\omega_4 = 3.1625$	203.161			
$\omega_5 = 3.5744$	2.8603			
$\omega_6 = 4.1134$	10.6317	3.35	3.75	2.34598084836201
$\omega_7 = 4.6386$	2.4229			
$\omega_8 = 5.5736$	28.3525			
$\omega_9 = 6.3485$	6.2916			
$\omega_{10} = 7.1127$	30.5030			
$\omega_{11} = 7.8169$	3.13656			

TABLE E.1: Parameters deviation results without losing the stability for system (2.47).

The fragility analysis for the PD-controller for the system (2.48) are summarized in table E.2.

Table E.3 summarizes the fragility analysis for the controller $(k_p^*, k_d^*) = (1.1021, 0.2514)$ for the system (2.49).

Frequency	$d_{\mathcal{T}_l}$	$k_{d\infty}$	k_{p_0}	$\min\{d_T, k_{d\infty}, k_{p_0}\}$
$\omega_1 = 0.9352$	8.2917			
$\omega_2 = 1.9489$	14.3176			
$\omega_3 = 3.9725$	9.8893			
$\omega_4 = 23.2533$	288.344		17.85	8.291787839
$\omega_5 = 33.7137$	27.1850			
$\omega_6 = 73.7534$	4792.91			
$\omega_7 = 95.1040$	101.249			

TABLE E.2: Parameters deviation results without losing the stability for system (2.48).

Frequency	$d_{\mathcal{T}_l}$	$k_{d\infty}$	k_{p_0}	$\min\{d_T, k_{d\infty}, k_{p_0}\}$
$\omega_1 = 1.0388$	0.562			
$\omega_2 = 1.8131$	0.804			
$\omega_3 = 2.7121$	0.491			
$\omega_4 = 4.7540$	2.164	0.7486	0.5229	0.4910436777
$\omega_5 = 6.3137$	0.609			
$\omega_6 = 12.9703$	1.217			
$\omega_7 = 16.1440$	14.602			

TABLE E.3: Parameters deviation results without losing the stability for system (2.49).

Bibliography

[1] R. Adams and B. Hannaford. Stable haptic interaction with virtual environments. *Robotics and Automation, IEEE Transactions on*, 15(3):465 –474, jun 1999.

[2] M. Agus, A. Giachetti, E. Gobbetti, G. Zanetti, and A. Zorcolo. Real-time haptic and visual simulation of bone dissection. *Presence: Teleoperators & Virtual Environments*, 12(1):110–122, 2003.

[3] S. Aiguo, Z. Qingjun, and H. Weiyi. Identification and control of bilateral telerobot with time delay. In *Proceedings of the International Conference on Intelligent Robots and Systems*, volume 3, pages 1353–1358, nov 1996.

[4] V. M. Alfaro. PID controllers fragility. *ISA Transactions*, 46(4):555 – 559, 2007.

[5] R. Anderson and M. Spong. Bilateral control of teleoperators with time delay. In *Proceedings of the 27th IEEE Conference on Decision and Control*, volume 1, pages 167–173, dec 1988.

[6] R. J. Anderson and M. W. Spong. Asymptotic stability for force reflecting teleoperators with time delay. *The International Journal of Robotics Research*, 11(2):135–149, 1992.

[7] C. Andriot and J. Perret. Immersive virtual prototyping with haptic feedback and virtual manikins. In *Proceedings of IDMME - Virtual Concept, Beijing, China*, oct. 2008.

[8] C. Andriot and J. Perret. Immersive virtual prototyping with haptic feedback and virtual manikins. In *Proceedings of IDMME - Virtual Concept, Beijing, China*, 2008.

[9] K. H. Ang, G. Chong, and Y. Li. PID control system analysis, design, and technology. *IEEE Transactions on Control Systems Technology*, 13(4):559 – 576, july 2005.

[10] P. Arcara and C. Melchiorri. A comparison of control schemes for teleoperation with time delay. In *1st IFAC Conference Telematics Applications in Automation and Robotics*, page 1505Ű510, jul 2001.

[11] P. Arcara and C. Melchiorri. Control schemes for teleoperation with time delay: A comparative study. *Robotics and Autonomous Systems*, 38(1):49–64, 2002.

[12] H. Arioui, S. Mammar, and T. Hamel. A smith-prediction based haptic feedback controller for time delayed virtual environments systems. In *Proceedings of the 2002 American Control Conference*, volume 5, pages 4303–4308, 2002.

[13] J. Artigas, J.-H. Ryu, and C. Preusche. Position drift compensation in time domain passivity based teleoperation. *International Conference on Intelligent Robots and Systems (IROS)*, pages 4250 –4256, oct 2010.

[14] J. Artigas, J. Vilanova, C. Preusche, and G. Hirzinger. Time domain passivity control-based telepresence with time delay. *International Conference on Intelligent Robots and Systems (IROS)*, 21(4):4205–4210, oct 2006.

[15] K. Astrom, C. Hang, and B. Lim. A new smith predictor for controlling a process with an integrator and long dead-time. *Automatic Control, IEEE Transactions on*, 39(2):343–345, 1994.

[16] K. Astrom and H. T. *PID Controllers: Theory, Design and Tuning.* ISA; 2nd Revised edition edition, 1995.

[17] K. Astrom and H. T. *Advanced PID Control.* ISA, 2006.

[18] K. Astrom and B. Wittenmark. *Adaptive Control.* Addison-Wesley, 1989.

[19] A. Aziminejad, M. Tavakoli, R. Patel, and M. Moallem. Stability and performance in delayed bilateral teleoperation: Theory and experiments. *Control Engineering Practice*, 16(11):1329–1343, 2008.

[20] N. Bajcinca. In W. Michiels(Ed.), Proc. of 5th IFAC workshop on time-delay systems,Oxford:Elsevier. In *Computation of stable regions in PID parameter space for time-delay systems*, 2005.

[21] A. Banach and W. Baumann. Gain-scheduled control of nonlinear partial differential equations. In *Proceedings of the 29th IEEE Conference on Decision and Control*, volume 2, pages 387 –392, dec 1990.

[22] C. Basdogan, C.-H. Ho, M. A. Srinivasan, and M. Slater. An experimental study on the role of touch in shared virtual environments. *ACM Transactions on Computer-Human Interactions*, 7(4):443–460, 2000.

[23] P. Bernstein, V. Hadzilacos, and N. Goodman. *Concurrency Control and Recovery in Database Systems*. Addison-Wesley Publishing Company, 1987.

[24] S. P. Bhattacharyya, H. Chapellat, and K. L. H. *Robust Control: The Parametric Approach.* Prentice Hall, 1995.

[25] R. Biernacki, H. Hwang, and S. Bhattacharyya. Robust stability with structured real parameter perturbations. *Automatic Control, IEEE Transactions on*, 32(6):495–506, 1987.

[26] F. Blanchini, D. Casagrande, S. Miani, and U. Viaro. Stable LPV realization of parametric transfer functions and its application to gain-scheduling control design. *IEEE Transactions on Automatic Control*, 55(10):2271–2281, oct. 2010.

[27] D. Borro, J. Savall, A. Amundarain, J. Gil, A. Garcia-Alonso, and L. Matey. A large haptic device for aircraft engine maintainability. *Computer Graphics and Applications, IEEE*, 24(6):70 – 74, nov.-dec. 2004.

[28] S. Butner and M. Ghodoussi. Transforming a surgical robot for human telesurgery. *Robotics and Automation, IEEE Transactions on*, 19(5):818–824, oct. 2003.

[29] B. A. C., C. V. Hollot, and H. Lin. Location of an entire polytope of polynomials: It suffices to check the edges. *Mathematics of Controls, Signals and Systems*, 1:61–71, 1988.

[30] J. Calvin, A. Dicken, B. Gaines, P. Metzger, and D. Miller, D.and Owen. The simnet virtual world architecture. In *Proc. of IEEE Virtual Reality Annual International Symposium*, pages 450–455, 1993.

[31] F. Cepolina and R. Michelini. Review of robotic fixtures for minimally invasive surgery. *The international Journal of Medical Robotics and Computer Assisted Surgery*, 1(1):43–63, 2004.

[32] H. Chapellat and S. Bhattacharyya. A generalization of kharitonov's theorem; robust stability of interval plants. *Automatic Control, IEEE Transactions on*, 34(3):306–311, 1989.

[33] N. Chebotarev and N. Meiman. The routh-hurwitz problem for polynomials and entire functions (in russian). In *Trudy Mat. Inst. Steklov, 26*, 1949.

[34] J.-J. Chen and Z.-C. Ji. The gain scheduling control for wind energy conversion system based on LPV model. In *International Conference on Networking, Sensing and Control (ICNSC)*, pages 653 –657, april 2010.

[35] J. Cheong, S.-I. Niculescu, A. Annaswamy, and M. Srinivasan. Motion synchronization in virtual environments with shared haptics and large time delays. In *Eurohaptics Conference, 2005 and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems, 2005. World Haptics 2005. First Joint*, pages 277 – 282, march 2005.

[36] J. Cheong, S.-I. Niculescu, and C. Kim. Motion synchronization control of distributed multisubsystems with invariant local natural dynamics. *IEEE Transactions on Robotics*, 25(2):382 –398, Apr. 2009.

[37] H. Ching and W. Book. Internet-based bilateral teleoperation based on wave variable with adaptive predictor and direct drift control. *Journal of Dynamic Systems, Measurement, and Control*, 128:86–93, 2006.

[38] N. Chopra, M. Spong, R. Ortega, and N. Barabanov. On tracking performance in bilateral teleoperation. *IEEE Transactions on Robotics*, 22(4):861–866, 2006.

[39] S. Chung and J.-J. Slotine. Cooperative robot control and concurrent synchronization of lagrangian systems. *IEEE Transactions on Robotics*, 25(3):2504–2509, 2009.

[40] J. Colgate, M. Stanley, and J. Brown. Issues in the haptic display of tool use. In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems 95 - "Human Robot Interaction and Cooperative Robots"*, volume 3, pages 140–145, aug 1995.

[41] P. Cominos and N. Munro. PID controllers: recent tuning methods and design to specification. *IEEE Proceedings of Control Theory and Applications*, 149(1):46 –53, jan 2002.

[42] J. Cushing and J. Cushing. *Integrodifferential equations and delay models in population dynamics*. Lecture notes in biomathematics. Springer, 1977.

[43] O. David, Y.Measson, C. Bidard, C. Rotinat-Libersa, and F. X. Russotto. Maestro: a hydraulic manipulator for maintenance and decomissionning. *European Nuclear Conference (ENC), Bruxelles, Belgium*, 2007.

[44] L. De Cicco, S. Mascolo, and S.-I. Niculescu. Brief paper: Robust stability analysis of smith predictor-based congestion control algorithms for computer networks. *Automatica*, 47(8):1685–1692, aug 2011.

[45] G. Dogangil, B. Davies, and F. y Baena. A review of medical robotics for minimally invasive soft tissue surgery. *Proceedings of the Institution of Mechanical Engineers, Part H: Journal of Engineering in Medicine*, 224(5):653–679, 2010.

[46] F. Du, W. Du, Y. Du, and B. Chen. Wireless networked control systems with gpc and new smith predictor. In *Chinese Control and Decision Conference (CCDC)*, pages 1920 –1924, may 2010.

[47] C. Duriez, F. Dubois, A. Kheddar, and C. Andriot. Realistic haptic rendering of interacting deformable objects in virtual environments. *Visualization and Computer Graphics, IEEE Transactions on*, 12(1):36–47, 2006.

[48] L. El'sgol'ts and S. Norkin. *Introduction to Theory and Applications of Differential Equations with Deviating Arguments*. Academic Press: New York, 2006.

[49] A. Fernandez, A. Barreiro, A. Banos, and J. Carrasco. Reset control for passive teleoperation. In *Industrial Electronics, 2008. IECON 2008. 34th Annual Conference of IEEE*, pages 2935 –2940, nov. 2008.

[50] M. Ferre, M. Buss, R. Aracil, C. Melchiorri, and C. Balaguer. Introduction to advances in telerobotics. In M. Ferre, M. Buss, R. Aracil, C. Melchiorri, and C. Balaguer, editors, *Advances in Telerobotics*, volume 31 of *Springer Tracts in Advanced Robotics*, pages 1–7. Springer Berlin / Heidelberg, 2007.

[51] W. R. Ferrell. Delayed force feedback. *Human factors*, 8:449–455, 1966.

[52] W. R. Ferrell and T. B. Sheridan. Supervisory control of remote manipulation. *Spectrum, IEEE*, 4(10):81–88, oct. 1967.

[53] P. Garcia-Robledo, P. Garcia-Borras, J. Barrio, M. Ferre, and R. Aracil. Teleoperation platform for multimodal experimentation. In *IEEE International Conference on Mechatronics (ICM)*, pages 1 –6, april 2009.

[54] R. Goertz. Mechanical master-slave manipulator. *Nucleonics (US) Ceased publication*, 12, 1954.

[55] H. Górecki, S. Fuksa, P. Grabowski, and A. Korytowski. *Analysis and synthesis of time delay systems*. Wiley New York, 1989.

[56] F. Gosselin, C. Megard, S. Bouchigny, F. Ferlay, F. Taha, P. Delcampe, and C. D'Hauthuille. A VR training platform for Maxillo Facial Surgery. *Applied Human Factors and Ergonomics (AHFE) International Conference, Miami, Florida, USA, Advances in Cognitive Ergonomics*, 2010.

[57] K. Gu, V. Kharitonov, and J. Chen. *Stability of time-delay systems*. Control engineering. Birkhäuser, 2003.

[58] K. Gu, S.-I. Niculescu, and J. Chen. On stability crossing curves for general systems with two delays. *Journal of Mathematical Analysis and Applications*, 311(1):231 – 253, 2005.

[59] H. W. Guggenheimer. *Differential Geometry*. Dover Books on Mathematics. Dover Publications, 1977.

[60] K. Guru, A. Hussain, R. Chandrasekhar, P. Piacente, M. Bienko, M. Glasgow, W. Underwood, G. Wilding, J. Mohler, M. Menon, et al. Current status of robot-assisted surgery in urology: a multi-national survey of 297 urologic surgeons. *The Canadian journal of urology*, 16(4):4736, 2009.

[61] G. Guthart and J. Salisbury, J.K. The intuitivetm telesurgery system: overview and application. In *Robotics and Automation, 2000. Proceedings. ICRA '00. IEEE International Conference on*, volume 1, pages 618–621, 2000.

[62] J. Hale and S. Lunel. *Introduction to functional differential equations*. Applied mathematical sciences. Springer-Verlag, 1993.

[63] B. Hannaford. A design framework for teleoperators with kinesthetic feedback. *Robotics and Automation, IEEE Transactions on*, 5(4):426 –434, aug 1989.

[64] B. Hannaford and J.-H. Ryu. Time-domain passivity control of haptic interfaces. *IEEE Transaction on Robotics and Automation*, 18(1):1 –10, 2002.

[65] B. Hannaford and J.-H. Ryu. Time domain passivity control of haptic interfaces. *U.S. Patent No. 7,027,965 B2, Apr. 11*, 2006.

[66] K. Hashtrudi-Zaad and S. Salcudean. Adaptive transparent impedance reflecting teleoperation. In *Proceedings of IEEE International Conference on Robotics and Automation*, volume 2, pages 1369–1374, apr 1996.

[67] K. Hashtrudi-Zaad and S. Salcudean. Transparency in time-delayed systems and the effect of local force feedback for transparent teleoperation. *Robotics and Automation, IEEE Transactions on*, 18(1):108 –114, feb 2002.

[68] K. Hashtrudi-Zaad and S. E. Salcudean. Analysis of control architectures for teleoperation systems with impedance/admittance master and slave manipulators. *The International Journal of Robotics Research*, 20(6):419–445, 2001.

[69] S. Haykin. *Active network theory*. Addison-Wesley series in electrical engineering. Addison-Wesley Pub. Co., 1970.

[70] K. Hertkorn, T. Hulin, P. Kremer, C. Preusche, and G. Hirzinger. Time domain passivity control for multi-degree of freedom haptic devices with time delay. *International Conference on Robotics and Automation (ICRA)*, pages 1313 –1319, may 2010.

[71] F. Herzog. *Lecture on Stochastic Systems*. ETH Zurich, 2010.

[72] S. Hirche, A. Bauer, and M. Buss. Transparency of haptic telepresence systems with constant time delay. In *Control Applications, 2005. CCA 2005. Proceedings of 2005 IEEE Conference on*, pages 328–333. IEEE, 2005.

[73] S. Hirche and M. Buss. Packet loss effects in passive telepresence systems. In *43rd IEEE Conference on Decision and Control*, volume 4, pages 4010–4015, dec. 2004.

[74] M.-T. Ho. Non-fragile PID controller design. In *Proceedings of the 39th IEEE Conference on Decision and Control*, volume 5, pages 4903–4908, 2000.

[75] W. K. Ho, C. C. Hang, and L. S. Cao. Tuning of PID controllers based on gain and phase margin specifications. *Automatica*, 31(3):497–502, Mar. 1995.

[76] P. F. Hokayem and M. W. Spong. Bilateral teleoperation: An historical survey. *Automatica*, 42(12):2035 – 2057, 2006.

[77] K.-S. Hong, D.-H. Kang, and J.-G. Kim. Robust smith predictor design via uncertainty quantification: Application to a reclaimer, jun. 2000.

[78] Z. Hu, S. E. Salcudean, and P. D. Loewen. Optimization-based teleoperation controller design. *13th IFAC Triennial World Congress, San Francisco, USA*, pages 405–410, 1996.

[79] T. Hudson, A. Helser, D. Sonnenwald, and M. Whitton. Managing collaboration in the nanomanipulator. *Presence*, 13(2):193–210, 2004.

[80] W. Iida and K. Ohnishi. Reproducibility and operationality in bilateral teleoperation. In *The 8th IEEE International Workshop on Advanced Motion Control. AMC'04*, pages 217–222. IEEE, 2004.

[81] T. Imaida, Y. Yokokohji, T. Doi, M. Oda, and T. Yoshikawa. Ground-space bilateral teleoperation of ets-vii robot arm by direct bilateral coupling under 7-s time delay condition. *IEEE Transactions on Robotics and Automation*, 20(3):499–511, june 2004.

[82] C. Jay, M. Glencross, and R. Hubbold. Modeling the effects of delayed haptic and visual feedback in a collaborative virtual environment. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 14(2):8, 2007.

[83] Y. Jing, T. Ren, Y. Zhou, and X. Zheng. Abr traffic control over atm networks with time-varying multiple time-delays using fuzzy-immune controller. In *Decision and Control, 2007 46th IEEE Conference on*, pages 5851 –5856, dec. 2007.

[84] H. Kazerooni, T.-I. Tsay, and K. Hollerbach. A controller design framework for telerobotic systems. *Control Systems Technology, IEEE Transactions on*, 1(1):50 –62, mar 1993.

[85] L. Keel and S. Bhattacharyya. Robust, fragile, or optimal? *Automatic Control, IEEE Transactions on*, 42(8):1098 –1105, aug 1997.

[86] A. A. Keller. *Time-Delay Systems: With Applications to Economic Dynamics and Control*. LAP LAMBERT Academic Publishing, 2011.

[87] G. Kim. *Designing virtual reality systems: the structured approach*. Springer-Verlag New York Incorporated, 2005.

[88] K. Kim, J. Park, H. Lee, and K. Song. Teleoperated cleaning robots for use in a highly radioactive environment of the DFDF. In *SICE-ICASE, 2006. International Joint Conference*, pages 3094 –3099, oct. 2006.

[89] H. Koivo and J. T. Tanttu. Tuning of PID controllers: Survey of siso and mimo techniques. *Preprints of the IFAC International Symposium on Intelligent Tuning and Adaptive Control, ITACŠ91, Singapore*, jan. 1991.

[90] V. Kolmanovskii and A. Myshkis. Applied theory of functional differential equations. 1992.

[91] V. Kolmanovskii and V. Nosov. *Stability of functional differential equations*, volume 180. Academic press, 1986.

[92] P. Kremer, T. Wimbock, J. Artigas, S. Schatzle, K. Johl, F. Schmidt, C. Preusche, and G. Hirzinger. Multimodal telepresent control of DLR's Rollin' JUSTIN. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 1601 –1602, may 2009.

[93] K. Kuchenbecker, W. Provancher, G. Niemeyer, and M. Cutkosky. Haptic display of contact location. In *Proceedings of the 12th International Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, pages 40–47, march 2004.

[94] T. Kuhlen, I. Assenmacher, and L. Jeřábková. Interacting in virtual reality. *Advanced Man-Machine Interaction*, pages 263–313, 2006.

[95] I. Landau. Robust digital control of systems with time delay (the smith predictor revisited). *International Journal of Control*, 62(2):325–347, 1995.

[96] D. Lawrence. Stability and transparency in bilateral teleoperation. *Robotics and Automation, IEEE Transactions on*, 9(5):624 –637, oct 1993.

[97] D. Lawrence. Gain scheduling dynamic linear controllers for a nonlinear plant. *Automatica*, 31(3):381–390, 1995.

[98] E. L.E. and S. Norkin. *Introduction to the theory and application of differential equations with deviating arguments*. Mathematics in science and engineering. Academic Press, 1973.

[99] A. Lecuyer, C. Andriot, and A. Crosnier. Interfaces haptiques et pseudo-haptiques. *Journees Nationales de la Recherche en Robotique*, 2003.

[100] D. Lee and K. Huang. Passive position feedback over packet-switching communication network with varying-delay and packet-loss. *Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems. Haptics.*, pages 335 –342, march 2008.

[101] D. Lee and M. W. Spong. Passive bilateral teleoperation with constant time delay. *IEEE Transactions on Robotics*, 22(2):269–281, 2006.

[102] D. Lee and D. Xu. Feedback R-passivity of Lagrangian systems for mobile robot teleoperation. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 2118 –2123, may 2011.

[103] S. Lee, J. Kim, and Y. Ishibashi. Transparency improvement of force-reflecting teleoperation over time-varying network delays. In *IEEE International Conference on Multimedia and Expo (ICME)*, pages 1010 –1015, july 2010.

[104] B. Liacu, C. Andriot, D. Dumur, F. Colledani, S.-I. Niculescu, and P. Boucher. Experimental comparative study of control architectures for haptic interfaces including communication delays. In *Submitted at Mediterranean Conference on Control and Automation*, 2012.

[105] B. Liacu, C. Mendez-Barrios, S.-I. Niculescu, and S. Olaru. Some Remarks on the Fragility of PD Controllers for SISO Systems with I/O delays. In *14th International Conference on System Theory and Control, Sinaia, Romania*, 2010.

[106] B. Liacu, I. Morarescu, S. Niculescu, C. Andriot, D. Dumur, F. Colledani, and P. Boucher. Control of a haptic system using smith predictor. In *Accepted for presentation at The 43rd International Symposium on Robotics (ISR)*, aug 2012.

[107] B. Liacu, I.-C. Morarescu, C. Andriot, S.-I. Niculescu, D. Dumur, P. Boucher, and F. Colledani. Some remarks on the fragility of smith predictors used in haptics. In *11th International Conference on Control, Automation and Systems (ICCAS)*, pages 1851 –1856, oct. 2011.

[108] H. Luanddders, H. Abbas, D. Doberstein, F. Thielecke, and H. Werner. LPV gain-scheduling control of an electromechanically driven landing gear for a commercial aircraft. In *American Control Conference ACC*, pages 4659 –4664, 2010.

[109] N. MacDonald. *Biological Delay Systems: Linear Stability Theory*. Cambridge University Press, 1989.

[110] M. Macedonia, M. Zyda, D. Pratt, P. Barham, and S. Zeswitz. A network software architecture for large-scale virtual environments. *Presence*, 3(4):265–287, 1994.

[111] P. Makila, L. Keel, and S. Bhattacharyya. Comments on "robust, fragile, or optimal?" [with reply]. *Automatic Control, IEEE Transactions on*, 43(9):1265 –1268, sept. 1998.

[112] N. Marcassus, A. Chriette, and M. Gautier. Theoretical and experimental overview of bilateral teleoperation control laws. *14th Mediterranean Conference on Control and Automation, 2006. MED '06.*, pages 1 –6, jun 2006.

[113] J. Marshall. *Control of time-delay systems*. P. Peregrinus, 1979.

[114] V. Mendez and M. Tavakoli. A passivity criterion for N-port multilateral haptic systems. In *49th IEEE Conference on Decision and Control (CDC)*, pages 274 –279, dec. 2010.

[115] C. Mendez-Barrios, S.-I. Niculescu, C.-I. Morarescu, and K. Gu. On the fragility of PI controllers for time-delay SISO systems. In *16th Mediterranean Conference on Control and Automation*, pages 529 –534, june 2008.

[116] W. Michiels and S. Niculescu. *Stability and stabilization of time-delay systems: an Eigenvalue-based approach*. Advances in design and control. Society for Industrial and Applied Mathematics, 2007.

[117] S. Mitra and T. Acharya. Gesture recognition: A survey. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, 37(3):311–324, 2007.

[118] C. Morarescu and S. Niculescu. Stability crossing curves of siso systems controlled by delayed output feedback. *Dynamics of Continuous Discrete and Impulsive Systems Series B*, 14(5):659, 2007.

[119] C.-I. Morarescu and B. Brogliato. Passivity-based tracking control of multiconstraint complementarity Lagrangian systems. In *47th IEEE Conference on Decision and Control*, pages 292–297, dec. 2008.

[120] I.-C. Morarescu, C.-F. Mendez-Barrios, S.-I. Niculescu, and K. Gu. Stability crossing boundaries and fragility characterization of PID controllers for siso systems with I/O delays. In *American Control Conference (ACC)*, pages 4988 –4993, jul 2011.

[121] I.-C. Morarescu, S.-I. Niculescu, and K. Gu. Stability crossing curves of shifted gamma-distributed delay systems. *SIAM Journal on Applied Dynamical Systems*, 6(2):475–493, 2007.

[122] I.-C. Morărescu. *Qualitative analysis of distributed delay systems: Methodology and algorithms*. Ph.D. thesis, University of Bucharest/Université de Technologie de Compiègne, September, 2006.

[123] I.-C. Morărescu, S.-I. Niculescu, and K. Gu. On the Geometry of PI Controllers for SISO Systems with Input Delays. *Proceedings of IFAC Time Delay Systems, Nantes, France*, 2007.

[124] I.-C. Morărescu, S.-I. Niculescu, and K. Gu. On the geometry of stability regions of smith predictors subject to delay uncertainty,. *IMA Journal of Mathematical Control and Information*, 24(3):411 – 423, 2007.

[125] I.-C. Morărescu, S.-I. Niculescu, and K. Gu. On the stability crossing curves of some distributed delay systems. *SIAM J. Appl. Dyn. System*, 6:475 – 493, 2007.

[126] I.-C. Morărescu, S.-I. Niculescu, and K. Gu. The geometry of stability crossing curves of PI controllers for siso systems with I/O delays. *Revue Roumaine MATH. PURES APPL.*, 55(4):297–313, 2010.

[127] J. Neimark. D-subdivisions and spaces of quasi-polynomials. *Prikl. Math. Mech.*, 13:349–380, 1949.

[128] S. Niakosari and S. Sirouspour. Improving transparency in network-based haptics. In *Third Joint EuroHaptics conference, and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems. World Haptics 2009.*, pages 547 –552, march 2009.

[129] S. Niculescu. *Delay effects on stability: a robust control approach*. Springer, 2001.

[130] G. Niemeyer. Using wave variables in time delayed force reflecting teleoperation. *Ph.D. dissertation, Massachussetts Institute of Technology*, 1996.

[131] G. Niemeyer and J.-J. Slotine. Designing force reflecting teleoperators with large time delays to appear as virtual tools. *Proceedings. IEEE International Conference on Robotics and Automation*, 3:2212 –2218, Apr. 1997.

[132] G. Niemeyer and J.-J. Slotine. Towards force-reflecting teleoperation over the internet. In *Proceedings of IEEE International Conference on Robotics and Automation*, volume 3, pages 1909 –1915 vol.3, may 1998.

[133] G. Niemeyer and J.-J. E. Slotine. Using wave variables for system analysis and robot control. *Proceedings of the IEEE International Conference on Robotics and Automation Albuquerque, New Mexico*, 2:1619–1625, apr 1997.

[134] G. Niemeyer and J.-J. E. Slotine. Telemanipulation with time delays. *The International Journal of Robotics Research*, 23(9):873–890, sep 2004.

[135] G. Niemeyer and J.-J. E. Slotine. Reducing wave-based teleoperator reflections for unknown environments. *IEEE Transaction on Industrial Electronics*, 58(2):392–397, feb 2011.

[136] E. Nuno, R. Ortega, N. Barabanov, and L. Basanez. A globally stable pd controller for bilateral teleoperators. *Robotics, IEEE Transactions on*, 24(3):753 –758, june 2008.

[137] A. O'Dwyer. *Compensation of processes with time delays by using an appropriately modified Smith predictor compensator*. Technical Report AOD.98.05, Dublin Institute of Technology, Kevin St., Dublin, 1998.

[138] A. O'Dwyer. PI and PID controller tuning rules for time delay process: a summary. In *Proceedings of the Irish Signals and Systems Conference,Dublin, Ireland*, pages 5–12, jun 2000.

[139] A. Okamura. Methods for haptic feedback in teleoperated robot-assisted surgery. *Industrial Robot: An International Journal*, 31(6):499–508, 2004.

[140] R. Ortega. *Passivity-based control of Euler-Lagrange systems: mechanical, electrical, and electromechanical applications*. Communications and control engineering. Springer, 1998.

[141] Z. Palmor and D. Powers. Improved dead-time compensator controllers. *AIChE journal*, 31(2):215–221, 1985.

[142] J. Park and G. Niemeyer. Haptic rendering with predictive representation of local geometry. In *Haptic Interfaces for Virtual Environment and Teleoperator Systems, 2004. HAPTICS'04. Proceedings. 12th International Symposium on*, pages 331–338. IEEE, 2004.

[143] J. Park and G. Niemeyer. Haptic rendering with predictive representation of local geometry. In *Proceedings of 12th International Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, pages 331–338, march 2004.

[144] L. Pontryagin. On the zeros of some elementary transcendental functions. In *(in Russian) Izv. Akad. Nauk SSSR 6 (1942) 115-134 (English translation in American Math. Soc. Transl., 95-110)*, 1955.

[145] L. Pontryagin. On the zeros of some transcendental functions. In *Doklady Akad. Nauk. SSSR 91 (1953) 1279-1280 (English translation in American Math. Soc. Transl., 19-20)*, 1958.

[146] A. Pressman, L. Welty, A. Karniel, and F. Mussa-Ivaldi. Perception of delayed stiffness. *The International Journal of Robotics Research*, 26(11-12):1191–1203, 2007.

[147] C. Preusche, T. Ortmaier, and G. Hirzinger. Teleoperation concepts in minimal invasive surgery. *Control Engineering Practice*, 10(11):1245 – 1250, 2002.

[148] B. Raj and R. Stern. Missing-feature approaches in speech recognition. *Signal Processing Magazine, IEEE*, 22(5):101–116, 2005.

[149] G. Raju, G. Verghese, and T. Sheridan. Design issues in 2-port network models of bilateral remote manipulation. In *Robotics and Automation, 1989. Proceedings., 1989 IEEE International Conference on*, volume 3, pages 1316 –1321, may 1989.

[150] A. Ramirez, R. Garrido, and S. Mondié. Integrated Retarded Velocity Control of DC Servomotors. *Proceedings of IFAC Joint conference ((5th Symposium on System Structure and Control, 11th Workshop on Time-Delay Systems, 6th Workshop on Fractional Differentiation and Its Applications)), Grenoble*, feb. 2013.

[151] V. Rasvan. Absolute stability of time lag control systems. *Ed. Academiei, Bucharest*, 1975.

[152] B. Richard and C. K. L. *Differential-difference equations / Richard Bellman, Kenneth L. Cooke*. Academic, New York, 1963.

[153] J. Richard. Time-delay systems: an overview of some recent advances and open problems. *automatica*, 39(10):1667–1694, 2003.

[154] E. Rodriguez-Seda, D. Lee, and M. Spong. Experimental comparison study of control architectures for bilateral teleoperators. *IEEE Transactions on Robotics*, 25(6):1304 – 1318, dec 2009.

[155] E. Rodriguez-Seda and M. Spong. A time-varying wave impedance approach for transparency compensation in bilateral teleoperation. In *IEEE/RSJ International Conference on Intelligent Robots and Systems. IROS.*, pages 4609 –4615, Oct. 2009.

[156] E. J. Rodriguez-Seda, D. Lee, and M. Spong. Experimental comparison study of control architectures for bilateral teleoperators. *IEEE Transactions on Robotics*, 25, 2009.

[157] O. Roesch, H. Roth, and S.-I. Niculescu. Remote control of mechatronic systems over communication networks. In *Mechatronics and Automation, 2005 IEEE International Conference*, volume 3, pages 1648 –1653, 2005.

[158] J.-H. Ryu, B. Hannaford, D.-S. Kwon, and J.-H. Kim. A simulation/experimental study of the noisy behavior of the time domain passivity controller. *IEEE Transactions on Robotics*, 21(4):733–741, june 2006.

[159] J.-H. Ryu, D.-S. Kwon, and B. Hannaford. Stability guaranteed control: Time domain passivity approach. *Proceedings of the 2002 IEEE/RSJ Intl. Conference on Intelligent Robots and Systems, Lausanne, Switzerland*, 3:2115 – 2121, 2002.

[160] J.-H. Ryu, D.-S. Kwon, and B. Hannaford. Stable teleoperation with time domain passivity control. *IEEE Intl. Conference on Robotics and Automation, ICRA, Washington, DC*, pages 3260–65, 2002.

[161] M. Saeki. Properties of stabilizing PID gain set in parameter space. *Automatic Control, IEEE Transactions on*, 52(9):1710 –1715, sept. 2007.

[162] S. Salcudean, N. Wong, and R. Hollis. Design and control of a force-reflecting teleoperation system with magnetically levitated master and wrist. *Robotics and Automation, IEEE Transactions on*, 11(6):844 –858, dec 1995.

[163] S. E. Salcudean, M. Zhu, W.-H. Zhu, and K. Hashtrudi-Zaad. Transparent bilateral teleoperation under position and rate control. *The International Journal of Robotics Research*, 19(12):1185–1202, 2000.

[164] G. Sankaranarayanan and B. Hannaford. Experimental comparison of internet haptic collaboration with time-delay compensation techniques. *IEEE International Conference on Robotics and Automation. ICRA 2008.*, pages 206–211, May 2008.

[165] I. SensAble Technologies. Specifications for the PHANTOM®Desktop™and PHANTOM Omni®haptic devices. In *PHANTOM Omni Technical Specification*, jul. 2009.

[166] T. Sheridan. Space teleoperation through time delay: review and prognosis. *Robotics and Automation, IEEE Transactions on*, 9(5):592 –606, oct 1993.

[167] T. Sheridan. Teleoperation, telerobotics and telepresence: A progress report. *Control Engineering Practice*, 3(2):205 – 214, 1995.

[168] G. Silva, A. Datta, and S. Bhattacharyya. *PID Controllers for Time-Delay Systems*. Birkhäuser, 2005.

[169] N. Simaan, R. Taylor, and P. Flint. High dexterity snake-like robotic slaves for minimally invasive telesurgery of the upper airway. *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2004*, pages 17–24, 2004.

[170] R. Sipahi, F. Atay, and S. Niculescu. Stability of traffic flow behavior with distributed delays modeling the memory effects of the drivers. *SIAM Journal on Applied Mathematics*, 68(3):738–759, 2007.

[171] S. Skaar and C. Ruoff. *Teleoperation and robotics in space*. Progress in astronautics and aeronautics. American Institute of Aeronautics and Astronautics, Inc., 1994.

[172] T. Slama, D. Aubry, R. Oboe, and F. Kratz. Nonlinear predictive control for bilateral scaled teleoperation systems using a nflat output: Theory and experiments. In *46th IEEE Conference on Decision and Control*, pages 4791–4796, dec. 2007.

[173] A. Smith and K. van Hashtrudi-Zaad. Neural network-based teleoperation using smith predictors. In *Mechatronics and Automation, 2005 IEEE International Conference*, volume 3, pages 1654 – 1659, 2005.

[174] O. J. M. Smith. Closer control of loops with dead time. *Chem. Eng. Prog.*, 53(5):217–219, 1957.

[175] C. Soh, C. Berger, and K. Dabke. On the stability properties of polynomials with perturbed coefficients. *Automatic Control, IEEE Transactions on*, 30(10):1033–1036, 1985.

[176] J. Sreng, A. Lecuyer, C. Megard, and C. Andriot. Using visual cues of contact to improve interactive manipulation of virtual objects in industrial assembly/maintenance simulations. *IEEE Transactions on Visualization and Computer Graphics*, 12(5):1013 –1020, sept.-oct. 2006.

[177] M. Srinivasan. What is haptics? *Laboratory for Human and Machine Haptics: The Touch Lab, Massachusetts Institute of Technology*, 1995.

[178] G. Stépán. *Retarded dynamical systems: stability and characteristic functions*, volume 200. Longman Scientific & Technical UK, 1989.

[179] S. Y. Sun, X. Chunmei, and Y. Huiqun. Research of adjusted smith predictor based on immune feedback. In *Measuring Technology and Mechatronics Automation (ICMTMA), 2010 International Conference on*, volume 2, pages 1072 –1075, Mar. 2010.

[180] S. Tafazoli, S. Salcudean, K. Hashtrudi-Zaad, and D. Lawrence. Impedance control of a teleoperated excavator. *IEEE Transactions on Control Systems Technology*, 10(3):355–367, 2002.

[181] M. Tavakoli, R. V. Patel, and M. Moallem. Haptic interaction in robot-assisted endoscopic surgery: a sensorized end-effector. *The International Journal of Medical Robotics and Computer Assisted Surgery*, 1(2):53–63, 2005.

[182] M. Tavakoli, R. V. Patel, M. Moallem, and A. Aziminejad. *Haptics for teleoperated surgical robotic systems*, volume 1. World Scientific, 2008.

[183] D. Tian, D. Yashiro, and K. Ohnishi. Haptic transmission by weighting control under time-varying communication delay. *Control Theory Applications, IET*, 6(3):420–429, 16 2012.

[184] J. Vetois. Asymptotic stability, convexity, and lipschitz regularity of domains in the anisotropic regime. In *Communications in Contemporary Mathematics (CCM)*, volume 12, pages 35 – 53, 2010.

[185] A. Villaverde, A. Blas, J. Carrasco, and A. Torrico. Reset control for passive bilateral teleoperation. *IEEE Transactions on Industrial Electronics*, 58(7):3037–3045, 2011.

[186] I. Vittorias and S. Hirche. Transparency of the generalized scattering transformation for haptic telepresence. In *Proceedings of the 2010 international conference on Haptics: generating and perceiving tangible sensations, Part I*, EuroHaptics'10, pages 183–188, 2010.

[187] C. R. Wagner, N. Stylopoulos, and R. D. Howe. *The role of force feedback in surgery: analysis of blunt dissection*, volume 2002, pages 68–74. IEEE Comput. Soc, 2002.

[188] F. Wang and D. Liu. *Networked control systems: theory and applications*. Springer, 2008.

[189] S. Wang, B. Xu, and Q. Wang. Delays analysis for teleoperation over internet and smith predictor with adaptive time-delay control. In *Robotics and Biomimetics (ROBIO). 2005 IEEE International Conference on*, pages 664 –669, 2005.

[190] E. T. Whittaker and G. N. Watson. *A Course in Modern Analysis*. Cambridge University Press, England, fourth edition, 1990.

[191] B. Willaert, B. Corteville, D. Reynaerts, H. Van Brussel, and E. Vander Poorten. Bounded environment passivity of the classical position-force teleoperation controller. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4622 –4628, oct. 2009.

[192] Z. Xu. *Bilateral Teleoperation of a Planetary Rover with Time Delays: A Master thesis from SpaceMaster, a Joint European Master in Space Science and Technology*. Number 110 in Applied Mathematical Sciences. LAP Lambert Academic Publishing, 2009.

[193] S. Yang. *Internet-Based Control Systems: Design and Applications*. Springer, 2011.

[194] Y. Yokokohji and T. Yoshikawa. Bilateral control of master-slave manipulators for ideal kinesthetic coupling-formulation and experiment. In *Robotics and Automation, 1992. Proceedings., 1992 IEEE International Conference on*, volume 1, pages 849–858, may 1992.

[195] B. Zhang, A. Kruszewski, and J.-P. Richard. Tracking improvement based on the Proxy control scheme for bilateral teleoperation system under time-varying delays. In *16th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, 2011.

[196] B. Zhang, A. Kruszewski, and J.-P. Richard. H_∞ Control Design for Novel Teleoperation System Scheme: A Discrete Approach. In *10th IFAC Workshop on Time Delay Systems*, 2012.

[197] B. Zhang, A. Kruszewski, and J.-P. Richard. H_∞ Control of Delayed Teleoperation Systems under Polytopic-Type Uncertainties. In *20th Mediterranean Conference on Control and Automation*, July 2012.

[198] B. Zhang, A. Kruszewski, and J.-P. Richard. H_∞ Robust Control Design for Teleoperation Systems. In *7th IFAC Symposium on Robust Control Design*, 2012.

[199] Q. Zhong. *Robust Control of Time-delay Systems*. SpringerLink: Springer e-Books. Springer, 2006.

[200] J. Ziegler and N. Nichols. Optimal settings for automatic controllers. *Trans. ASME*, 64(11):759–768, 1942.

[201] C. Zilles and J. Salisbury. A constraint-based god-object method for haptic display. In *Intelligent Robots and Systems 95. 'Human Robot Interaction and Cooperative Robots', Proceedings. 1995 IEEE/RSJ International Conference on*, volume 3, pages 146–151, aug 1995.

Résumé

Au cours des dernières décennies, les environnements virtuels se sont de plus en plus répandus et sont largement utilisés dans de nombreux domaines comme, par exemple, le prototypage, la formation à l'utilisation de différents outils/appareils, l'aide à la réalisation de tâches difficiles, etc. L'interaction avec la réalité virtuelle, ainsi que le retour d'effort, sont assurés par des interfaces haptiques. En général, ces systèmes sont affectés par des retards de communication et de traitement, entraînant une détérioration des performances. Dans cette thèse, une étude complète des méthodes existantes, les outils théoriques et de nouvelles solutions sont proposés dans le cadre de l'haptique.

Dans un premier temps, une étude comparative, fondée sur des résultats expérimentaux obtenus sur un système haptique à un degré de liberté, met en évidence les avantages et les inconvénients des algorithmes de commande les plus classiques, transposés du domaine de la téléopération à l'haptique. Sont ensuite examinés les outils théoriques nécessaires à l'analyse de la stabilité des systèmes à retard selon différentes situations, tenant compte des limites physiques des plates-formes expérimentales considérées. En plus du cas classique du retard constant, des incertitudes sont également considérées et modélisées par plusieurs types de distributions (distribution uniforme, normale et gamma avec gap).

Finalement, pour surmonter les inconvénients liés aux retards, deux nouvelles approches sont proposées. Tout d'abord, la commande de type prédicteur de Smith est reprise et une solution spécifique pour les systèmes haptiques est mise en œuvre. L'idée principale consiste à introduire dans le prédicteur de Smith les forces liées à l'environnement en utilisant les informations complémentaires issues de la réalité virtuelle, en ce qui concerne les distances entre l'objet virtuel piloté et d'autres objets présents dans la scène. Pour surmonter la perte de performances induite par l'utilisation d'un gain fixe dans les correcteurs, commun à toutes les situations (mouvements libres ou restreints), la seconde approche propose un correcteur Proportionnel Dérivé incluant une stratégie de séquencement de gain en fonction de la distance jusqu'à une éventuelle collision. Les deux approches sont validées expérimentalement sur une plateforme haptique à trois degrés de liberté, pour différents scénarios de complexité progressive, partant de situations avec des mouvements simples - libre et restreints, des contacts avec des objets en mouvement, pour arriver à des situations plus complexes - boîte virtuelle avec des murs fixes ou mobiles.

Abstract

During the last decades, virtual environments have become very popular and are largely used in many domains as, for example, prototyping, trainings for different devices, assistance in completing difficult tasks, etc. The interaction with the virtual reality, as well as the feedback force, is assured by haptic interfaces. Generally, such systems are affected by communication and processing time-delays, resulting in a deterioration of performances. In this thesis, a complete study of the existing methods, as well as theoretical tools and new solutions, are proposed for the haptic framework.

First, a comparative study, based on the experimental results obtained on a 1-DOF haptic system, highlights the advantages and drawbacks of the most common control algorithms ported from teleoperation to haptics. Next, the theoretical tools needed in analyzing the stability of the delayed systems in different situations, as well as the physical limitations of the experimental platforms considered, are examined. Besides the standard case of constant time-delays, uncertainties are also considered and modeled by different types of distributions (uniform, normal and gamma distribution with gap).

In the sequel, for overcoming the drawback of time-delays, two new approaches are proposed. First, the use of Smith predictor-based control is addressed and a specific solution for haptic systems is developed and discussed. The main idea is to introduce into the Smith predictor the environmental forces by using the additional information from the virtual reality regarding the distances between the controlled virtual object and other objects in the scene. To overcome the loss of performances induced by using a fixed gain in the controllers for all situations (free or restricted motions), the second approach proposes a gain-scheduling Proportional Derivative control strategy depending on the distance until a possible collision. Both approaches are experimentally validated on a 3-DOF haptic platform, under different scenarios elaborated gradually from simple situations - free and restricted motion, contacts with moving objects, to more complex situations - virtual box with fixed or moving sides.