# CONTROLLER DESIGN FOR HAPTIC SYSTEMS UNDER DELAYED POSITION AND VELOCITY FEEDBACK

A THESIS

SUBMITTED TO THE DEPARTMENT OF ELECTRICAL AND

ELECTRONICS ENGINEERING

AND THE GRADUATE SCHOOL OF ENGINEERING AND SCIENCE

OF BILKENT UNIVERSITY

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

MASTER OF SCIENCE

By

Ahmet Taha Koru

August, 2012

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Prof. Dr. Hitay Özbay(Advisor)

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Prof. Dr. Ömer Morgül

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Assistant Prof. Dr. Melih Çakmakcı

Approved for the Graduate School of Engineering and Science:

Prof. Dr. Levent Onural
Director of the Graduate School

# ABSTRACT

## CONTROLLER DESIGN FOR HAPTIC SYSTEMS UNDER DELAYED POSITION AND VELOCITY FEEDBACK

Ahmet Taha Koru
M.S. in Electrical and Electronics Engineering
Supervisor: Prof. Dr. Hitay Özbay
August, 2012

This thesis considers controller design for haptic systems under delayed position and velocity feedback. More precisely, a complete stability analysis of a haptic system, where local dynamics are described by some second-order mechanical dynamics, is presented. Characteristic equation of this system with time delays involves quasi-polynomials. By a change of variables in the characteristic equation, stability conditions are obtained analytically and regions are plotted by using Matlab.

Next, using two optimization techniques ($H_\infty$ and stability margin optimization) optimal choice for the controller gains is proposed. $H_\infty$ optimization minimizes tracking error between devices while avoiding large control action inputs. $H_\infty$ analysis requires high computational cost for accurate results due to its dependency to frequency domain. On the other hand, stability margin optimization defines a cost function that expresses the trade-off between system bandwidth and robustness with low computational cost. The derived results are tested on a three degree of freedom real-time experimental platform to illustrate the theoretical results. Finally robustness analysis is performed for optimal parameters to find allowable delay perturbations.

*Keywords:* haptic systems, time delay, H-infinity optimization, stability limits, PD control.

# ÖZET

# GECİKMELİ POZİSYON VE HIZ GERİBESLEMESİNE SAHİP HAPTİK SİSTEMLER İÇİN KONTROLCÜ TASARIMI

Ahmet Taha Koru
Elektrik-Elektronik Mühendisliği, Yüksek Lisans
Tez Yöneticisi: Prof. Dr. Hitay Özbay
Austos, 2012

Bu tez zaman gecikmeli pozisyon ve hız geribeslemesine sahip haptik sistemler için denetleyici tasarımı ile ilgilidir. Mekanik sistemler ikinci dereceden dinamik denklemler ile tanımlanmıştır. Zaman gecikmeli bu sistemin karakteristik denklemi yarı-polinom (kuasi-polinom) içermektedir. Sistemin karakteristik denklemi, çeşitli değişken değişiklikleriyle basitleştirilerek, kararlılık analizi analitik olarak gerçekleştirilmiş, kararlı parametreli gösteren grafikler Matlab yardımı ile elde edilmiştir.

Daha sonra, iki farklı optimizasyon tekniği uygulanılarak, en uygun parametreler hesaplanmıştır. $H_\infty$ optimizasyonu, iki mekanik sistemin birbirini takip etme hatalarını minimize ederken, yüksek enerji gerektirmeyen parametreleri hesaplamaktadır. $H_\infty$ analizinin bilgisayar ortamında gerçekleştirilmesi, her bir frekans için işlem yapıldığı için yüksek hesaplama bedeli gerektirmektedir. Öte yandan kararlılık marjı optimizasyonu tekniği, sistemin bant genişliği ve gürbüzlüğü arasında denge kurmaktadır. Sadece belirli bir frekansta uygulandığı için düşük hesaplama bedeli vardır. Mekanik sistemdeki ve zaman gecikmesindeki karışıklıklara karşı gürbüzlük analizi yapılmıştır. Son olarak elde edilen sonuçlar üç serbestlik dereceli gerçek zamanlı deney düzeneğinde test edilmiştir.

*Anahtar sözcükler*: haptik sistemler, zaman gecikmeli sistemler, H-infinity optimizasyonu, PD kontrolcü.

# Acknowledgement

I am sincerely grateful to Prof. Hitay Özbay for his supervision, guidance and valuable suggestions throughout the development of this thesis. Working with him on this project was the most instructive phase of my academic life.

Thanks to Bogdan Liacu, Silviu-Iulian Niculescu and Claude Andriot for their valuable comments and collaboration in the PIA Bosphorus Project ”109E127” jointly funded by TÜBİTAK and EGIDE. I am very happy to take part in this international project. My thesis based on a joint paper named ”Low-Order Controller Design for Haptic Systems under Delayed Feedback” which we prepared together in the context of the Bosphorus Project. Bogdan and I won the IFAC-TDS best student contribution award for our paper presented at IFAC-TDS 2012, Boston, USA.

Special thanks to Doğukan Dağ for his help in checking the English of the preliminary version of the thesis.

Many thanks to my family for their loving support and for their encouragement to pursue my academic career.

# Contents

# List of Figures

# List of Tables

# Chapter 1

# Introduction

Teleoperation systems, which origins dates back to mid 1940's, has been first considered to be mechanical real to real communication between the master and slave robots by Goertz. Technologic developments has paved the way for new research areas such as space telerobotics, telesurgery, nuclear telerobotics [3, 7]. Along with real to real coupling, new systems has developed which uses virtual interfaces instead of slave robots. One of the research area of haptic systems with virtual environment is the surgery simulations. Such a system allows student surgeons to safely practice and improve their skills [5, 4]. Also haptic feedback is used in computer-aided design(CAD) in the manufacturing and automotive industry. Ability to touch the objects under design may increase creativity and efficiency of designers [1].

From the control theory point of view, design of haptic system involves two main goals:

- **Stability:** Robust stability of the closed-loop system irrespective of behaviour of the user or the environment despite difficulties such as time delays, dynamic uncertainties.
- **Transparency:** Haptic system renders forces, which slave robot encounters, at haptic side to convey sense of touch [1]. Realistic touch perception in haptic side is desired.

These two constraints are conflicting [7]. In order to satisfy these requirements, control system must be carefully designed to obtain high performance with low position tracking error and realistic sense of touch while maintaining robustness. Communication medium leads to complications, since delays have strong impact on stability and performance of overall system.

Many different techniques are proposed to control haptic systems, including passivity theory [2], remote compliance control [8], wave variable encoding [13] and etc. [16, 7]. Wave-variable controllers does not guarantee position coordination between interfaces [14]. Controllers with closed-loop force-feedback systems achieve realistic force feedback, however they are sensitive to time delays and are highly fragile [16]. We proposed a PD-control (feedback from position and velocity) approach in this thesis, which is robust against delays.

PD-controllers, in [14, 10], control position difference between master and slave robots, and contains a dissipating term to guarantee passivity. Thus, these controllers actually are not PD but pretend as PD controllers. On the other hand, Lee and Spong present a real PD control, where dissipating term is still used to obtain passivity. Dissipating term decreases performance of the system.

In this thesis, we present a PD controller (without the dissipating term) design for haptic systems under processing and communication delays. After defining the system with dynamic equations which includes time delayed position and velocity terms, we provide stability analysis. Since our stability analysis does not require passivity assumptions, we come through with complete region of controller gains, for which system is stable, by using classical tools of feedback control theory(such as phase and gain margins). In section 3, optimal gains are chosen using $H_\infty$ based optimization and stability margin optimization. Stability analysis and optimal gains sections are from our joint work with Liacu et al., a conference paper [11]. This thesis also includes robustness analysis, gain scheduling strategy section and a new 1-DOF experimental set-up. Robustness for such systems is crucial since they are used in surgical operations. Unstable behaviour for a solitary moment can cause serious damage or even death of patient. Robustness of the system against perturbations in time delays and plant parameters were analysed besides effect of unmodeled dynamics. Then, a gain

scheduling strategy is presented to get high performance of system in transparency point of view. Thus, high force feedback while interacting with an virtual object and low viscous effect in free motion are obtained. Ultimately, theoretical results were verified with 1-DOF real-time experimental set-up, similar to the generic system shown in Figure 1.1.

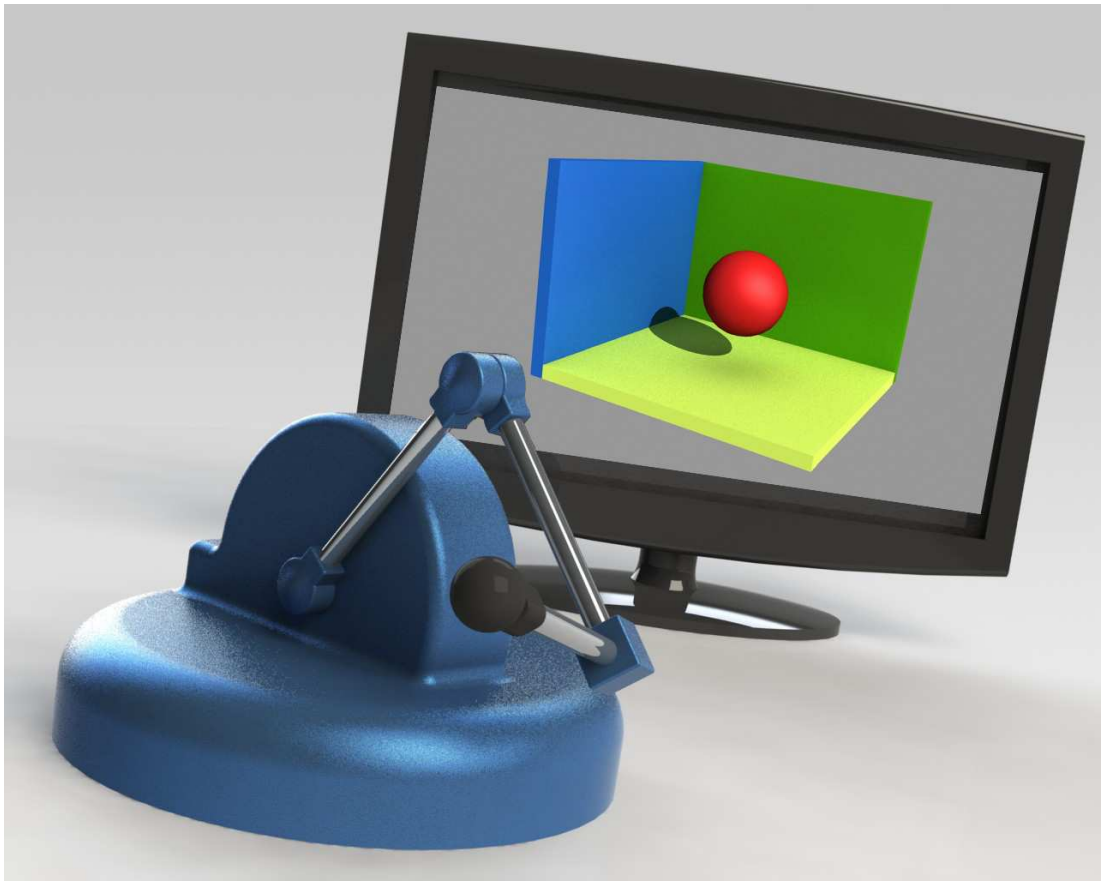

Figure 1.1: Illustration of a haptic system.

# Chapter 2

# Stability Analysis

## 2.1 Mathematical Model

A typical haptic system signal flow graph is shown in Figure 2.1.

An ideal haptic system should realistically mimic slave dynamics in haptic side, so system should satisfy the following conditions:

- Small position tracking error with robust stability.
- Realistic force feedback is desired at haptic side. When slave robot in free motion, force feedback at haptic side should be as small as possible; while stiff response is obtained when slave come into contact with environment.

A linear model describing dynamics of master/slave system can be written as:

$$M_h\ddot{x}_1(t) + B_h\dot{x}_1 = -F_1(t) + F_h(t), \tag{2.1}$$

$$M_v\ddot{x}_2(t) + B_v\dot{x}_2 = -F_2(t) + F_e(t), \tag{2.2}$$

where $x_* \in \mathbb{R}^n$ ($* = h$ or $v$) are the generalized coordinates, $F_* \in \mathbb{R}^n$ are the (generalized) input forces, $M_*$ are the positive inertia matrices, $B_*$ are the damping matrices

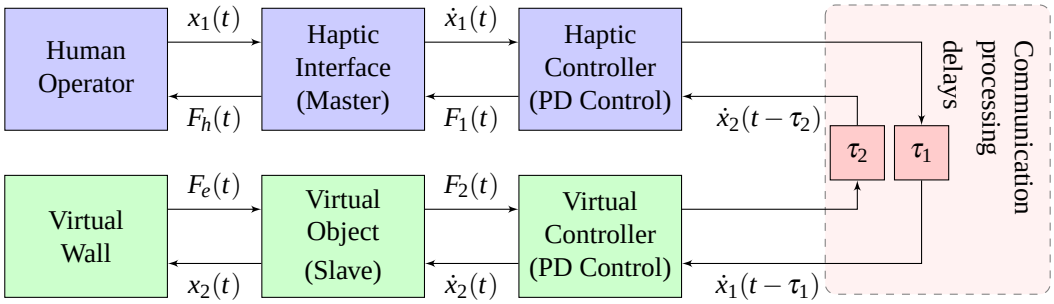


Figure 2.1: General PD control scheme for haptic systems.

and $F_h$, $F_e$ correspond to the external forces exerted by human operator and the environment, respectively [7]. The main idea can be resumed to using *two similar PD controllers*, one for controlling the haptic interface and another for the (corresponding) virtual object. In such a configuration, it follows:

$$F_1(t) = \underbrace{K_{d_1}(\dot{x}_1(t) - \dot{x}_2(t-\tau_2))}_{\text{delayed D-action}} \underbrace{+K_{p_1}(x_1(t) - x_2(t-\tau_2))}_{\text{delayed P-action}}, \tag{2.3}$$

$$F_2(t) = \underbrace{K_{d_2}(\dot{x}_2(t) - \dot{x}_1(t-\tau_1))}_{\text{delayed D-action}} \underbrace{+K_{p_2}(x_2(t) - x_1(t-\tau_1))}_{\text{delayed P-action}}, \tag{2.4}$$

where $\tau_1, \tau_2$ are the constant communication delays and $K_{p_1}, K_{d_1}, K_{p_2}, K_{d_2}$ are the PD control gains corresponding to the haptic and virtual controller respectively.

## 2.2 Stability Analysis

The system represented by Figure 2.1 and equations (2.1)-(2.4) can bu summarized in the block diagram of Figure 2.2 where $P_1$ and $P_2$ represent transfer functions of the plants and $C_1$ and $C_2$ are the controllers.

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

$$X_1(s) = P_1(s)\left(F_h(s) - C_1(s)\left(X_1(s) - e^{-\tau_2 s}X_2(s)\right)\right) \tag{2.5}$$

$$X_2(s) = P_2(s)\left(-F_e(s) + C_2(s)\left(-X_2(s) + e^{-\tau_1 s}X_1(s)\right)\right) \tag{2.6}$$

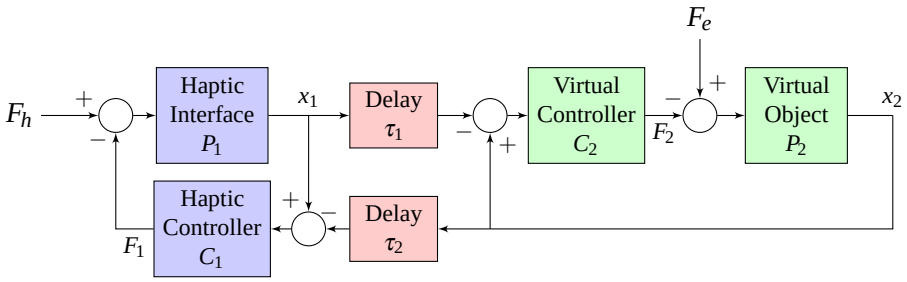


Figure 2.2: Bilateral Haptic System.

where $X_i(s)$ denotes the Laplace transform of the time signal $x_i(t)$, $i = 1,2$; similarly for $F_h(s)$ and $F_e(s)$; $\tau_1 > 0$ and $\tau_2 > 0$ are the time delays. Transfer functions $P_i(s)$ and $C_i(s)$ are taken as

$$P_1(s) = P_2(s) = \frac{1}{s(ms+b)} =: P(s) \tag{2.7}$$

$$C_1(s) = C_2(s) = K_p + K_d s =: C(s) \tag{2.8}$$

$$\text{where } m > 0,\ b > 0,\ K_p > 0,\ K_d > 0.$$

Re-arranging (2.5) and (2.6) we obtain

$$\begin{bmatrix} 1+P_1(s)C_1(s) & -P_1(s)C_1(s)e^{-\tau_2 s} \\ -P_2(s)C_2(s)e^{-\tau_1 s} & 1+P_2(s)C_2(s) \end{bmatrix} \begin{bmatrix} X_1(s) \\ X_2(s) \end{bmatrix} = \begin{bmatrix} P_1(s)F_h(s) \\ -P_2(s)F_e(s) \end{bmatrix} \tag{2.9}$$

Therefore, with the plant and controller definitions (2.7) and (2.8), the characteristic equation of the feedback system is

$$(1+P(s)C(s))^2 - (P(s)C(s))^2 e^{-(\tau_1+\tau_2)s} = 0 \tag{2.10}$$

which is equivalent to

$$\left(1+P(s)C(s)-P(s)C(s)e^{-\tau s}\right)\ \left(1+P(s)C(s)+P(s)C(s)e^{-\tau s}\right) = 0$$

$$\text{where} \quad \tau := \frac{(\tau_1+\tau_2)}{2} \tag{2.11}$$

Note that

$$(1+PC)^{-1} = \frac{s(ms+b)}{ms^2+(b+K_d)s+K_p} \tag{2.12}$$

is a stable second order system with positive coefficients for all $K_p > 0,\ K_d > 0$. Hence, from (2.11) it is clear that the feedback system is stable if and only if the following two equations do not have zeros in $\overline{\mathbb{C}}_+$.

$$1 + G(s)\ \left(\frac{1-e^{-\tau s}}{s}\right) = 0 \quad \text{where} \quad G(s) = \frac{K_p + K_d s}{ms + b} \tag{2.13}$$

$$1 + T(s)\ \ e^{-\tau s} = 0 \quad \text{where} \quad T(s) = \frac{K_p + K_d s}{s(ms+b) + K_p + K_d s} \tag{2.14}$$

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

$$G(s) = K\ \frac{1+\tau_c s}{1+\tau_p s} \tag{2.15}$$

$$T(s) = \frac{K(1+\tau_c s)}{\tau_p s^2 + (1+\tau_c K)s + K}. \tag{2.16}$$

We can further make a frequency normalization

$$\widehat{s} = \tau_p s \tag{2.17}$$

and introduce new definitions

$$L := \frac{1}{K\tau_p} = \frac{b^2}{m\,K_p} \qquad \alpha := \frac{\tau_c}{\tau_p} = \frac{b\,K_d}{m\,K_p} \qquad h := \frac{\tau}{\tau_p} = \frac{(\tau_1+\tau_2)\,b}{2\,m} \tag{2.18}$$

so that the characteristic equations (2.13) and (2.14) can be re-written as

$$1 + \frac{1}{L}\ \frac{(1+\alpha\widehat{s})}{(1+\widehat{s})}\ \ \left(\frac{1-e^{-h\widehat{s}}}{\widehat{s}}\right) = 0 \tag{2.19}$$

$$1 + \frac{(1+\alpha\widehat{s})}{(L\widehat{s}^2 + (L+\alpha)\widehat{s} + 1)}\ e^{-h\widehat{s}} = 0. \tag{2.20}$$

We will try to find the controller parameters $L$ and $\alpha$ (which define $K_p$ and $K_d$), as functions of $h$, that place all the roots of (2.19) and (2.20) in $\mathbb{C}_-$. In what follows we will restrict ourselves the case where $K_p$ and $K_d$ are positive, i.e., $L > 0$ and $\alpha > 0$.

Analysis of stability conditions of transfer functions (2.19) and (2.20) are based on Nyquist Stability Criterion. Let us consider (2.19) 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 $\omega > 0$ and

$$\lim_{\omega \searrow 0^+} \angle(1-e^{-jh\omega}) = +\frac{\pi}{2}.$$

Therefore,

$$0 \leq \angle f(j\omega) \leq -\pi, \qquad \forall\ \omega \in \mathbb{R}, \qquad \text{where} \qquad f(\widehat{s}) := \frac{1-e^{-h\widehat{s}}}{\widehat{s}}. \tag{2.21}$$

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 (2.19) are in $\mathbb{C}_-$ when $\alpha > 1$, independent of $L$ and $h$. Furthermore, when $\alpha = 1$ the equation (2.19) reduces to:

$$1 + \frac{1}{L} f(\widehat{s}) = 0.$$

Note that whenever $\angle f(j\omega) = -\pi$, $|f(j\omega)| = 0$ holds. This fact, together with (2.21), implies that when $\alpha = 1$ all the roots of (2.19) are in $\mathbb{C}_-$, independent of $L$ and $h$. In conclusion, the analysis of (2.19) becomes interesting only if $\alpha < 1$. In this case, all the roots of (2.19) are in $\mathbb{C}_-$ if and only if the following condition is met:

$$L > \left| \frac{1+j\alpha\omega_p}{1+j\omega_p} \right| \ |f(j\omega_p)|, \tag{2.22}$$

where $\omega_p$ is the smallest $\omega > 0$ satisfying:

$$\tan^{-1}(\alpha\omega) - \tan^{-1}(\omega) - \frac{h\omega}{2} = -\pi. \tag{2.23}$$

The condition (2.22) gives an allowable region in the $(\alpha, L)$-plane for all the roots of (2.19) to be in $\mathbb{C}_-$ when $\alpha < 1$.

Note that the following identity used in (2.23):

$$\angle f(j\omega) = -\frac{h\omega}{2}, \qquad \forall\ \omega \in [0\ , \frac{2\pi}{h}],$$

We can re-arrange the equation (2.23) as:

$$\pi - \left(\tan^{-1}(\omega_p) - \tan^{-1}(\alpha\omega_p)\right) = \frac{h\omega_p}{2}. \tag{2.24}$$

It is a simple exercise 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 $\alpha < 1$, (2.22) is now equivalent to:

$$L > \frac{2(1-\alpha)}{\omega_p^2 + 1}, \tag{2.25}$$

where $\omega_p$ is determined from (2.24).

Now consider (2.20). 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 the following condition holds:

$$2(1-\alpha) > L, \tag{2.26}$$

in which case:

$$\omega_c = \sqrt{\frac{2(1-\alpha)}{L} - 1}. \tag{2.27}$$

This means that if (2.26) 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 (2.20) are in $\mathbb{C}_-$ independent of $h$. Since $\omega_p$ is a positive real number, the condition (2.25) also holds irregardless of delay value $h$ when (2.26) is not satisfied. To complete the analysis, now assume (2.26) is satisfied and $\omega_c$ is as defined by (2.27). In this case, by the Nyquist criterion, all the roots of (2.20) are in $\mathbb{C}_-$ if and only if:

$$\tan^{-1}(\alpha\omega_c) - \tan^{-1}\left( \frac{(L+\alpha)\omega_c}{1 - L\omega_c^2} \right) - h\omega_c > -\pi. \tag{2.28}$$

To recapitulate, with the parameter definitions of (2.18), the feedback system described by (2.9) is stable independent of $h$ if $\alpha \geq 1$. When $\alpha < 1$, system is stable independent of $h$ if $L > 2(1-\alpha) > 0$ and 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 (2.22) and (2.28).

Since (2.27) implies:

$$L = \frac{2(1-\alpha)}{1+\omega_c^2},$$

the condition (2.25) can be re-written as:

$$\omega_p > \omega_c. \tag{2.29}$$

Let us now re-consider (2.28). Using the notation $L = 2(1-\alpha)/(1+\omega_c^2)$, then, after simple algebraic manipulations, it is easy to see that:

$$\begin{aligned}\tan^{-1}(\alpha\omega_c) - \tan^{-1}\left(\frac{(L+\alpha)\omega_c}{1-L\omega_c^2}\right) &= -\tan^{-1}\left(\frac{2(1-\alpha)\omega_c(1+\alpha\omega_c^2)}{(1+\alpha\omega_c^2)^2-(2(1-\alpha)\omega_c)^2}\right)\\ &= -2\tan^{-1}\left(\frac{(1-\alpha)\omega_c}{(1+\alpha\omega_c^2)}\right) = -2\left(\tan^{-1}(\omega_c) - \tan^{-1}(\alpha\omega_c)\right).\end{aligned}$$

Thus the condition (2.28) is equivalent to:

$$\frac{\pi - 2\left(\tan^{-1}(\omega_c) - \tan^{-1}(\alpha\omega_c)\right)}{\omega_c} > h. \tag{2.30}$$

Recall from (2.29) and (2.24) that $\omega_c$ is restricted to satisfy $\omega_p > \omega_c$, where $\omega_p$ is defined from:

$$\frac{2\left(\pi - \left(\tan^{-1}(\omega_p) - \tan^{-1}(\alpha\omega_p)\right)\right)}{\omega_p} = h. \tag{2.31}$$

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 $2(1-\alpha) > L > 0$ 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 $\omega_p$ is defined as the solution of the equation $g_p(x) = h$ and $\omega_o$ as the solution of the equation $g_c(x) = h$, then $\omega_o < \omega_p$ and hence, for $\alpha < 1$, the feedback system shown in Figure 2.1 is stable if and only if $\omega_c < \omega_o$, which is equivalent to:

$$L > \frac{2(1-\alpha)}{1+\omega_o^2}, \qquad \text{where} \quad \omega_o > 0 \ \text{ is the solution of } \quad g_c(x) = h\,. \tag{2.32}$$

The stability condition (2.32) is expressed in terms of $L = b^2/(m\,K_p)$ and $\alpha = (b\,K_d)/(m\,K_p)$. From this condition we can determine the allowable range of $m\,K_p/b^2$ and $K_d/b$ for all $h > 0$. Note that the system is stable independent of $h$ when $\alpha > 1$. The stability region is shown for two different time delay values in Figure 2.3.

**Theorem 1.** *The bilateral haptic system is asymptotically stable independent of the delay values ($\tau_1$, $\tau_2$) if the controller gains satisfy the condition:*

$$K_d \geq \frac{m}{b} K_p. \tag{2.33}$$

*Furthermore, when $K_d/K_p < m/b$, we have two cases:*

1. *If $0 < mK_p - bK_d < b^2/2$, then the feedback system is stable independent of the delay values ($\tau_1$, $\tau_2$).*

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

$$mK_p - bK_d < \frac{b^2}{2}(1+\omega_0^2), \tag{2.34}$$

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

$$\frac{\pi - 2\left(\tan^{-1}(x) - \tan^{-1}\left(\frac{b}{m}\frac{K_d}{K_p}x\right)\right)}{x} = \frac{(\tau_1+\tau_2)b}{2m}. \tag{2.35}$$

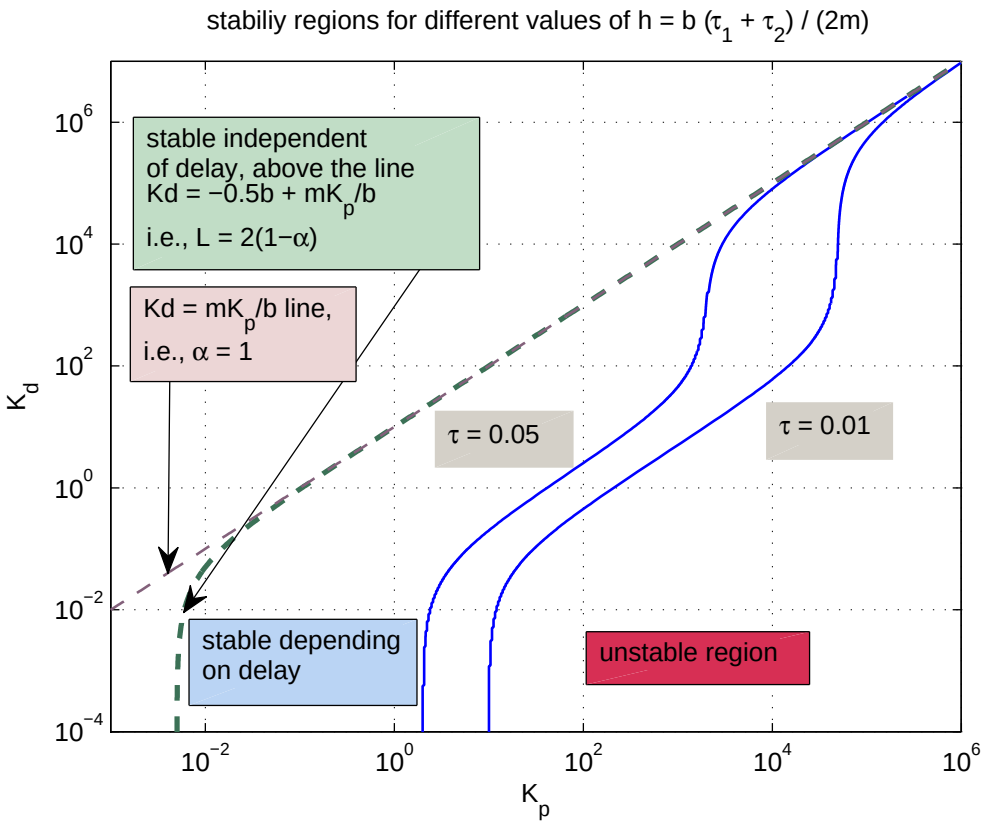


Figure 2.3: Allowable region of controller parameters for stability of the bilateral haptic system.

# Chapter 3

# Optimal Gains

Once the parameter space is identified for stability of the feedback system, the next question is to find the best controller parameters in this set. Clearly, parameters close to the boundary of the stability region are not acceptable, since these will result in fragility. On the other hand, parameters too deep in the stability region are not desirable from the performance point of view.

Two different optimization techniques, which are $H_\infty$ and stability margin optimization, are proposed in this chapter to find optimal parameters for haptic system.

## 3.1 $H_\infty$ Based Design

Let us define the position tracking error

$$e(t) := x_1(t) - x_2(t). \tag{3.1}$$

From (2.9) we compute

$$E(s) = \frac{P(s)}{1 + P(s)C(s) + P(s)C(s)e^{-\tau s}}(F_h(s) + F_e(s)). \tag{3.2}$$

While trying to make the error small we may be forced to use high command signals which may lead to actuator saturation. Since large control signals are not desirable, we

also want to penalize the control. Again, from (2.9), the output of the controller, $F_2(t)$, on the virtual side can be computed as

$$\begin{aligned} F_2(s) &= C(s)(e^{-\tau s}X_1(s) - X_2(s)) \\ &= \frac{\big(C(s)e^{-\tau s}+(1+P(s)C(s)-P(s)C(s)e^{-2\tau s})\big)P(s)(F_h(s)+F_e(s))}{(1+P(s)C(s)+P(s)C(s)e^{-\tau s})(1+P(s)C(s)-P(s)C(s)e^{-\tau s})} \end{aligned}$$

In particular, when $F_e = 0$ we have

$$\begin{bmatrix} E(s) \\ F_2(s) \end{bmatrix} = \left(\frac{T(s)}{1+T(s)e^{-\tau s}}\right) \begin{bmatrix} 1/C(s) \\ \frac{e^{-\tau s}}{1+P(s)C(s)(1-e^{-\tau s})} \end{bmatrix} F_h(s) \tag{3.3}$$

where $T(s) = P(s)C(s)(1+P(s)C(s))^{-1}$. Therefore, optimal gains from the $H_\infty$ control point of view are the ones which solve the problem

$$\min_{K_p,K_d} \left\| \frac{P(s)}{1+P(s)C(s)(1+e^{-\tau s})} \begin{bmatrix} \rho \\ \\ \frac{C(s)}{(1+P(s)C(s)(1-e^{-\tau s}))} \end{bmatrix} \right\|_\infty \tag{3.4}$$

where $\rho$ is a design parameter which represents the trade-off between small tracking error $e$ and small control action $F_2$. Depending on the values of $\rho$ we obtain the optimal $K_p$ and $K_d$, for each fixed $m = 1$, $b = 0.1$ and $\tau = 0.05$, as shown in Table 3.1.

Table 3.1: $H_\infty$ optimal gains for different $\rho$

| $b^2\rho$ | 0.01 | 0.1 | 1 | 10 | 50 | 100 |
|---|---|---|---|---|---|---|
| $K_p$ | 0.8 | 17.1 | 85.0 | 246 | 305 | 310 |
| $K_d$ | 8.8 | 10.2 | 15.2 | 43 | 55 | 51 |

We see that for large values of $\rho$ (emphasizing tracking performance, i.e., trying to make $\|e\|_2$ small compared to $\|F_2\|_2$) $H_\infty$ optimal gains are in the order of $K_p \in [240\ ,\ 310]$ and $K_d \in [40\ ,\ 55]$. In the next section we will compare these values with another set of gains obtained from a different optimality criterion.

## 3.2 Stability Margin Optimization

Recall from (2.25) that one of the stability conditions is

$$\left(\frac{b^2}{m\,K_p}\right)\ \left(\frac{1+\omega_p^2}{2(1-\alpha)}\right) > 1. \tag{3.5}$$

Note that $\omega_o < \omega_p$ so, if we define

$$GM_1 := \left(\frac{b^2}{m\,K_p}\right)\ \left(\frac{1+\omega_o^2}{2(1-\alpha)}\right)$$

then $GM_1 > 1$ implies (3.5). So, we will try to make $GM_1$ as large as possible. On the other hand, for large bandwidth in the system (fast response) we require that $\omega_c$ is as large as possible, i.e.

$$\omega_c^2 + 1 = \frac{m\,K_p}{b^2}\ 2(1-\alpha)$$

should be as large as possible. But this conflicts with $GM_1$ should be large condition. So, we will blend these two conflicting objectives and try to

$$\text{maximize}\ \ \min\{\rho_1(\omega_c^2+1)\ ,\ \frac{1}{\rho_1}GM_1\}$$

where $\rho_1$ assigns a relative weight for each component of the problem. The solution of this problem gives

$$\frac{m\,K_p}{b^2} = \frac{1}{\rho_1}\ \frac{\sqrt{1+\omega_o^2}}{2(1-\alpha)}. \tag{3.6}$$

Under this choice, we have

$$GM_1 = \rho_1\sqrt{1+\omega_o^2}. \tag{3.7}$$

Note that the right hand sides of (3.6) and (3.7) are functions of $\alpha$ once $\rho_1$ and $h = \tau b/m$ are fixed.

Now, $(m\,K_p/b^2)$ is the controller gain, and to avoid actuator saturations this gain should not be too high. So, we can define a new cost function which tries to make $GM_1$ large and $K_p$ small:

$$\text{minimize}\ \ \left(\frac{\rho_2}{\rho_1\sqrt{1+\omega_o^2}} + \frac{b^2}{m\,\rho_2}\ \frac{1}{\rho_1}\ \frac{\sqrt{1+\omega_o^2}}{2(1-\alpha)}\right) \tag{3.8}$$

where $\rho_2$ assigns relative weights for $GM_1$ and $K_p$. Note that $\rho_1$ does not play a role in the solution of (3.8). Once $\rho_2$ and $h = \tau\,b/m$ are fixed, the cost function defined in (3.8)

depends on $\alpha$ only. Minimizing the cost function gives optimal $\alpha$, then this gives $\omega_o$ and $K_p$ via (3.6); and once $K_p$ is known, we can find $K_d = \alpha\, m\, K_p/b$. Table 3.2 shows the optimal gains for varying $\rho_2$ when $\rho_1 = b^2 = 0.01$, $m = 1$ and $h = \tau b/m = 0.005$ are fixed.

Table 3.2: Optimal gains and $GM_1$ for different $\rho_2$, when $\tau = 0.05$, $m = 1$, and $\rho_1 = b^2 = 0.01$.

| $\rho_2$ | 10 | 20 | 30 | 40 | 50 | 60 | 80 | 100 |
|---|---|---|---|---|---|---|---|---|
| $K_p$ | 94 | 207 | 301 | 389 | 425 | 436 | 446 | 453 |
| $K_d$ | 2.4 | 6.3 | 12.7 | 34.3 | 82 | 127 | 207 | 291 |
| $GM_1$ | 1.33 | 2.9 | 4.2 | 5.5 | 6.0 | 6.1 | 6.16 | 6.2 |

Table 2 shows that $GM_1$ increases with increasing $\rho_2$, but for $\rho_2 \geq 50$ additional gain in $GM_1$ is very small. Therefore, a meaningful choice would be $K_p \in [390\ ,\ 410]$ and $K_d \in [35\ ,\ 45]$. Compared to the $H_\infty$ optimal gains corresponding to relatively large $\rho$ values, the above $K_p$ values are about 1.3 to 1.5 times higher, whereas $K_d$ values are 1.14 to 1.25 times lower. For the experimental tests, the values $K_p = 400$ and $K_d = 40$ are used and results are reported in the next section. These correspond to $\rho_2 \approx 42$ in the above table. For the $H_\infty$ optimal gains we may select $K_p = 275$ and $K_d = 45$; we expect the stability margins to be larger in this case, but the response will be slower. For relatively small $\rho$ values in the $H_\infty$ optimal design, we have $K_p = 85$ and $K_d = 15$ (e.g. for $b^2\rho = 1$) in which case the emphasis on tracking performance is diminished compared to larger $\rho$ values. In the next section experimental results for $K_p = 400$, $K_d = 40$ case and $K_p = 85$, $K_d = 15$ case will be illustrated.

# Chapter 4

# Robustness Analysis

## 4.1 Delay Perturbations

Smallest time delay which de-stabilizes the feedback system for a given set of controller and plant parameters can be calculated from Theorem 1, as follows:

$$\tau_{max} = \frac{\pi - 2\left(\tan^{-1}(\omega_0) - \tan^{-1}\left(\frac{b\,K_d}{m\,K_p}\omega_0\right)\right)}{\omega_0}\frac{m}{b}, \tag{4.1}$$

where

$$w_0 = \sqrt{\frac{2}{b^2}(mK_p - bK_d) - 1}. \tag{4.2}$$

This can be seen as the largest tolerable delay. Time domain simulations in Figure 4.1, 4.2 and 4.3 illustrate the results found in Table 4.1.

Table 4.1: Allowable perturbations of delay for $H_\infty$ optimal gain parameters when $m = 1$ and $b = 0.1$

| $K_p$ | 17.1 | 85.0 | 246 | 305 | 310 | 400 |
|---|---|---|---|---|---|---|
| $K_d$ | 10.2 | 15.2 | 43 | 55 | 51 | 40 |
| $\tau_{max}$ | 0.4589 | 0.1811 | 0.1202 | 0.1105 | 0.1080 | 0.0876 |

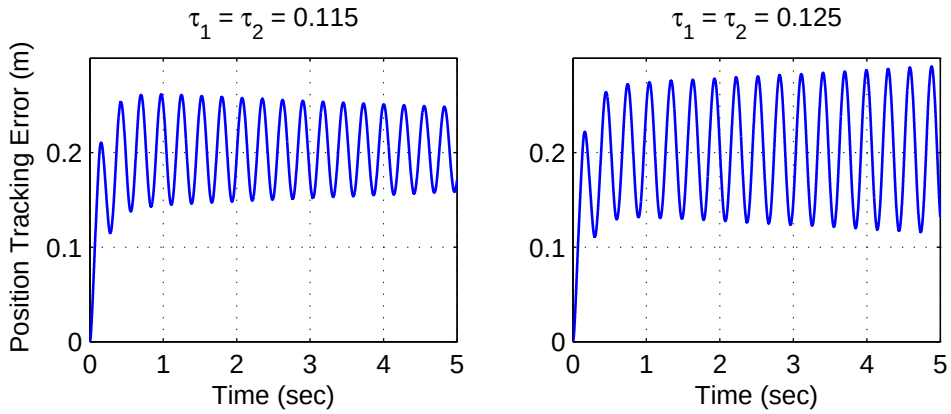


Figure 4.1: System is stable for $\tau < 0.1202$, marginally stable for $\tau = 0.1202$ and unstable for $\tau > 0.1202$ when $K_p = 246$ and $K_d = 43$.

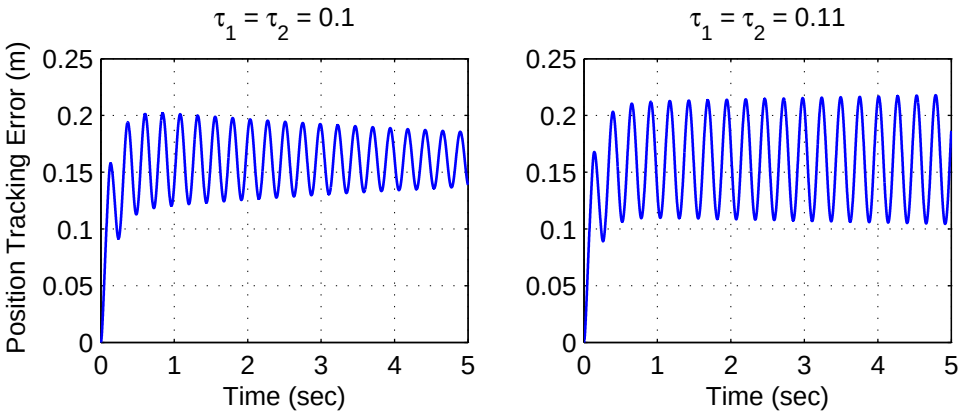


Figure 4.2: $\tau_{max} = 0.1080$ for $K_p = 310$ and $K_d = 51$.

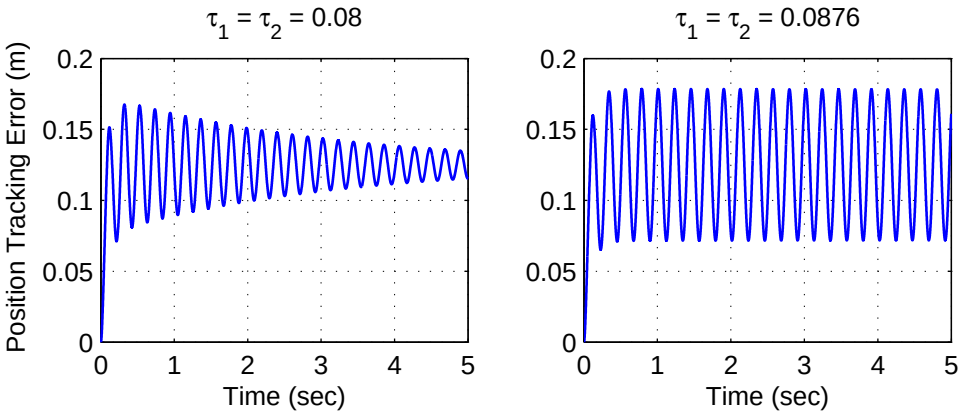


Figure 4.3: $\tau_{max} = 0.0876$ for $K_p = 400$ and $K_d = 40$.

## 4.2 Parametric Plant Perturbations

In this section, we discuss stability robustness due to possible uncertainties in the plant parameters $m$ and $b$.

Introducing

$$C(s) := C_1(s) = C_2(s), \qquad L_1(s) := P_1(s)C(s), \qquad L_2(s) := P_2(s)C(s), \tag{4.3}$$

leads to a characteristic equation

$$1 + L_1(s) + L_2(s) + L_1(s)L_2(s) - L_1(s)L_2(s)e^{-2\tau s} = 0. \tag{4.4}$$

Using the identity

$$P_1(s) = \frac{1}{s(m_1 s + b_1)} \tag{4.5}$$

and some algebraic manipulations, the characteristic equation can be written as:

$$m_1 s + b_1 = \frac{(1 + L_2(s) - L_2(s)e^{-2\tau s})C}{-(1 + L_2(s))} =: H(s). \tag{4.6}$$

We can find $m_1^*$ and $b_1^*$ pairs for marginally stable characteristic equation (4.6) as in [12].

Allowable plant parameters and corresponding simulation results are shown below.

$$m_1^* = \frac{Im(H(j\omega))}{\omega}, \;\; b_1^* = Re(H(j\omega)) \;\; \forall \omega. \tag{4.7}$$

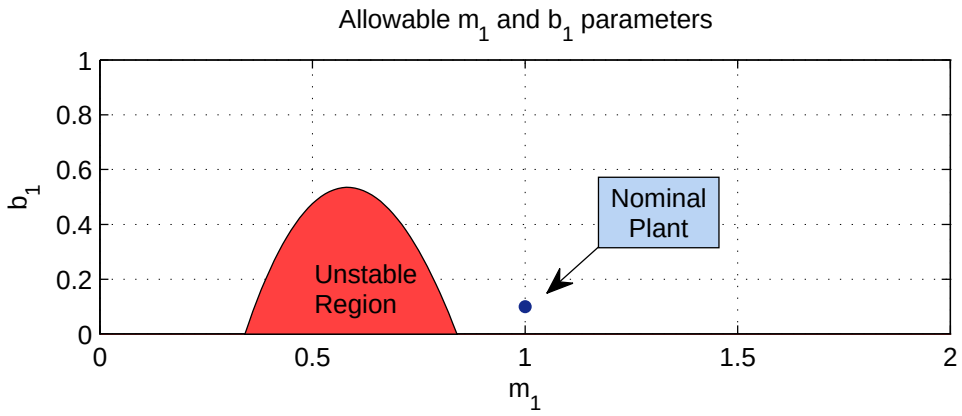


Figure 4.4: Allowable plant parameters for $m_2 = 1$, $b_2 = 0.1$, $K_p = 400$, $K_d = 40$, $\tau = 0.085$.

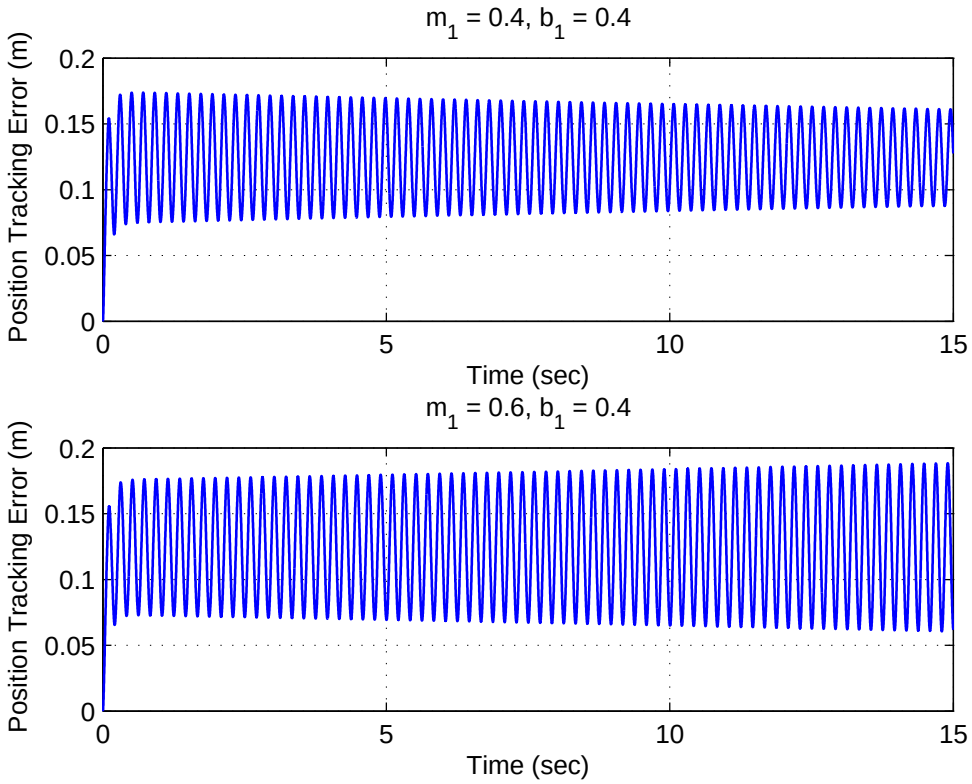


Figure 4.5: System response for $m_2 = 1$, $b_2 = 0.1$, $K_p = 400$, $K_d = 40$, $\tau = 0.085$.

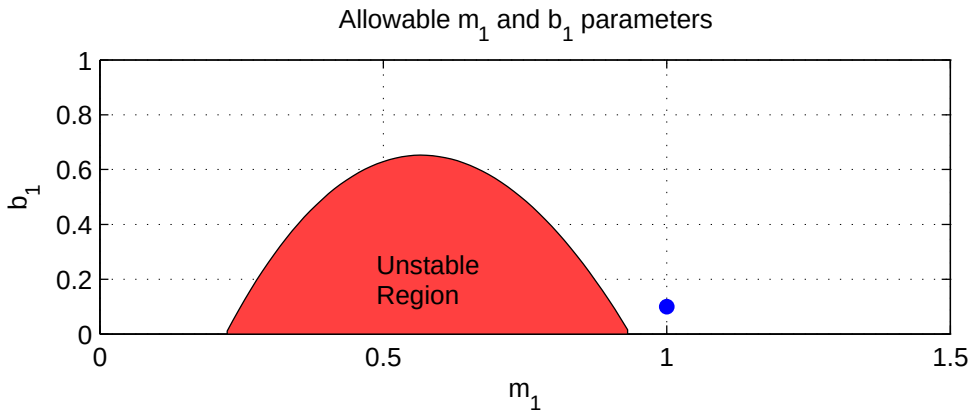


Figure 4.6: Allowable plant parameters for $m_2 = 1$, $b_2 = 0.1$, $K_p = 246$, $K_d = 43$, $\tau = 0.118$.

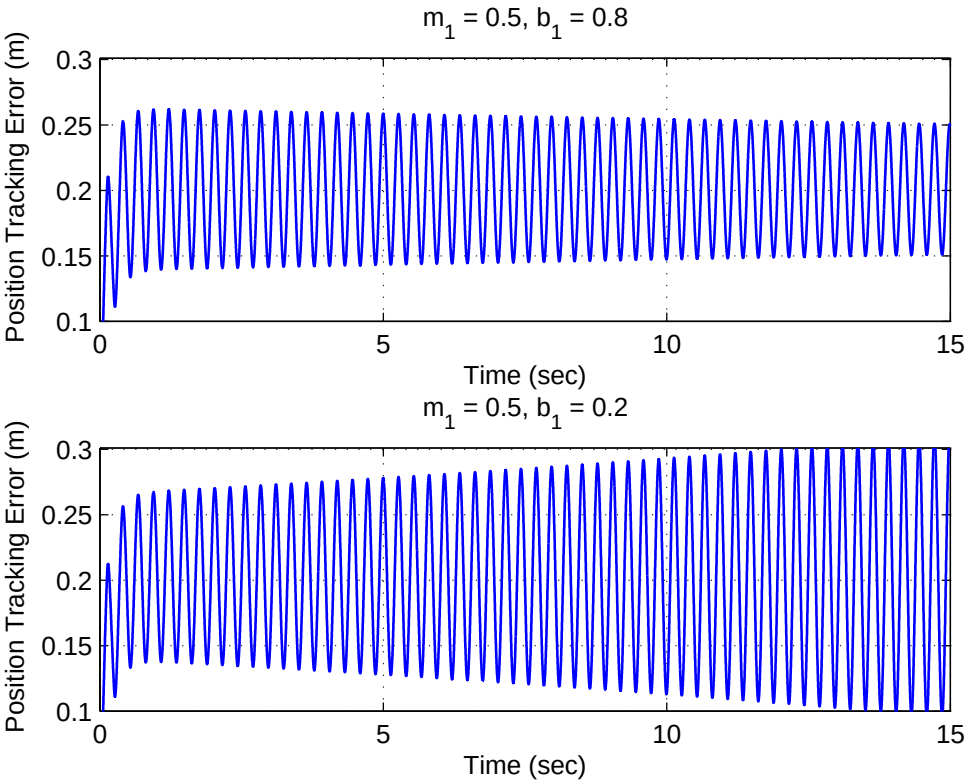


Figure 4.7: System response for $m_2 = 1$, $b_2 = 0.1$, $K_p = 246$, $K_d = 43$, $\tau = 0.118$.

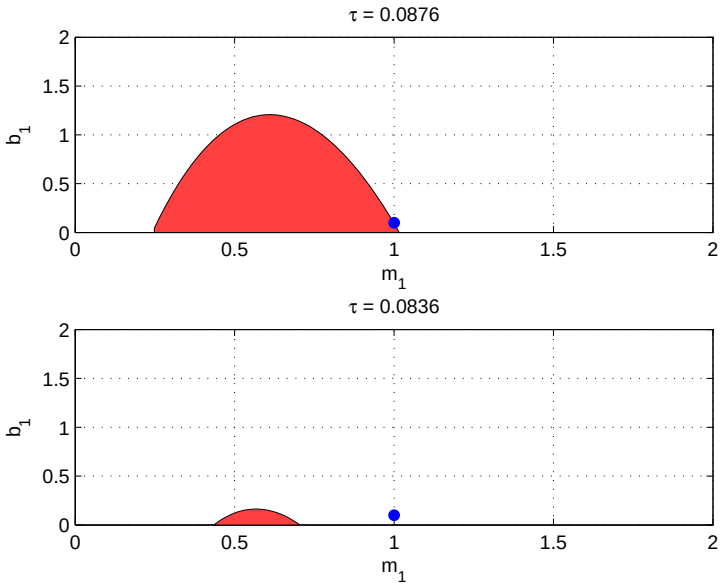


Figure 4.8: Unstable region shrinks as delay decreases, unstable region expands as delay increases ($K_p = 400$, $K_d = 40$).

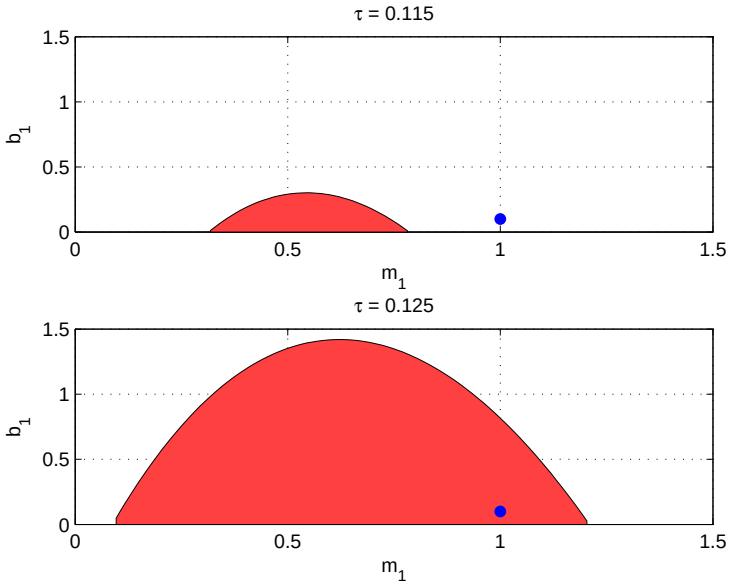


Figure 4.9: More allowable plant parameters region for $m_2 = 1$, $b_2 = 0.1$, $K_p = 246$, $K_d = 43$.

## 4.3 Robustness Against Unmodeled Dynamics

Our plant model can be slightly different than the real model due to uncertainties such as unmodeled dynamics and approximation of the parameters. To avoid undesirable effects of these uncertainties, our controller gains should stabilize all possible plants. If we define one of the plants as:

$$P_1(s) = P(s) + \Delta(s) \tag{4.8}$$

we can apply robust stability test. Characteristic equation of the perturbed system is:

$$(1+P(s)C(s))(1+(P(s)+\Delta)C(s)) - (P(s)+\Delta)P(s)C(s)^2e^{-2\tau s} = 0 \tag{4.9}$$

After some algebraic manipulations, characteristic equation becomes the characteristic equation of nominal plant multiplied by a function with perturbed terms.

$$(1+P(s)C(s))(1+T(s)e^{-\tau s})(1+G(s)f_\tau(s))\left[1+\Delta_m(\frac{T(s)}{1+T(s)e^{-\tau s}})(\frac{1+G(s)f_{2\tau}(s)}{1+G(s)f_\tau(s)})\right] \tag{4.10}$$

where

$$\Delta_m(s) := \frac{P_1(s)-P(s)}{P(s)} \qquad f_\tau(s) = \frac{1-e^{-\tau s}}{s}. \tag{4.11}$$

In 4.10, transfer functions $T(s)$ and $G(s)$ are defined the same way as in equations (2.13) and (2.14) and $\Delta_m$ is called multiplicative perturbation. In Chapter 3, we provided controller parameters for which the *nominal* feedback system is stable and performance criteria is satisfied. For robust stability, these parameters should also satisfy following inequality:

$$\left|\left|\Delta_m(s)(\frac{T(s)}{1+T(s)e^{-\tau s}})(\frac{1+G(s)f_{2\tau}(s)}{1+G(s)f_\tau(s)})\right|\right|_\infty := ||\Delta_m(s)R(s)||_\infty < 1 \tag{4.12}$$

By using equation (4.12), we can derive allowable magnitude of perturbation:

$$|\Delta_m(j\omega)| < \frac{1}{|R(j\omega)|} \tag{4.13}$$

Figure 4.10 shows that the only frequency range where tolerable uncertainty bound is less than 100% is between 20rad/sec and 50 rad/sec (where tolerable uncertainty

bound is between 50% and 100%); any unmodeled lightly damped flexible modes in this frequency range may destabilize the feedback system, otherwise the system is quite robust to unmodeled dynamics.

To illustrate this result, the system is perturbed with:

$$W(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \tag{4.14}$$

which represents an unmodeled flexible mode of the system. Perturbed plant is defined as follows:

$$P_1(s) = P(s)(1 + W(s)). \tag{4.15}$$

Corresponding simulation results with different $\zeta$ and $\omega_n$ are shown in Figure 4.11, for controller gains chosen as $K_p = 400$ and $K_d = 40$.

Figure 4.12 shows allowable plant perturbations for lower controller gains. Figure 4.13 verifies system is more robust for lower gains as expected.

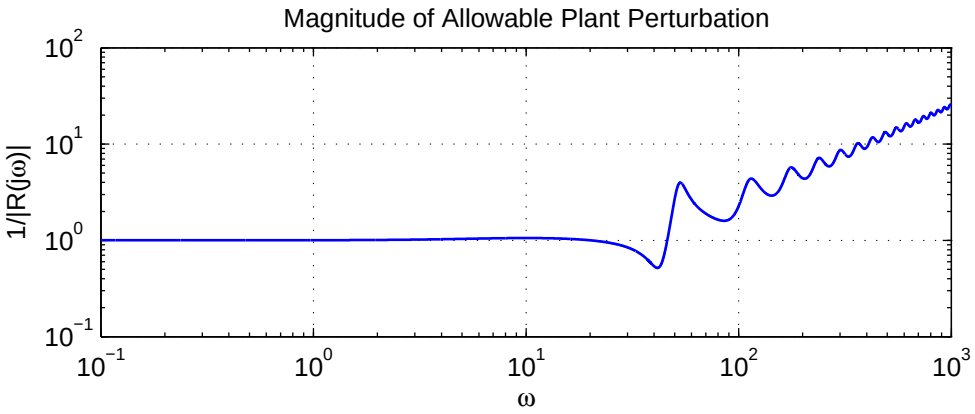


Figure 4.10: $m = 1$, $b = 0.1$, $\tau = 0.05$, $K_p = 400$, $K_d = 40$

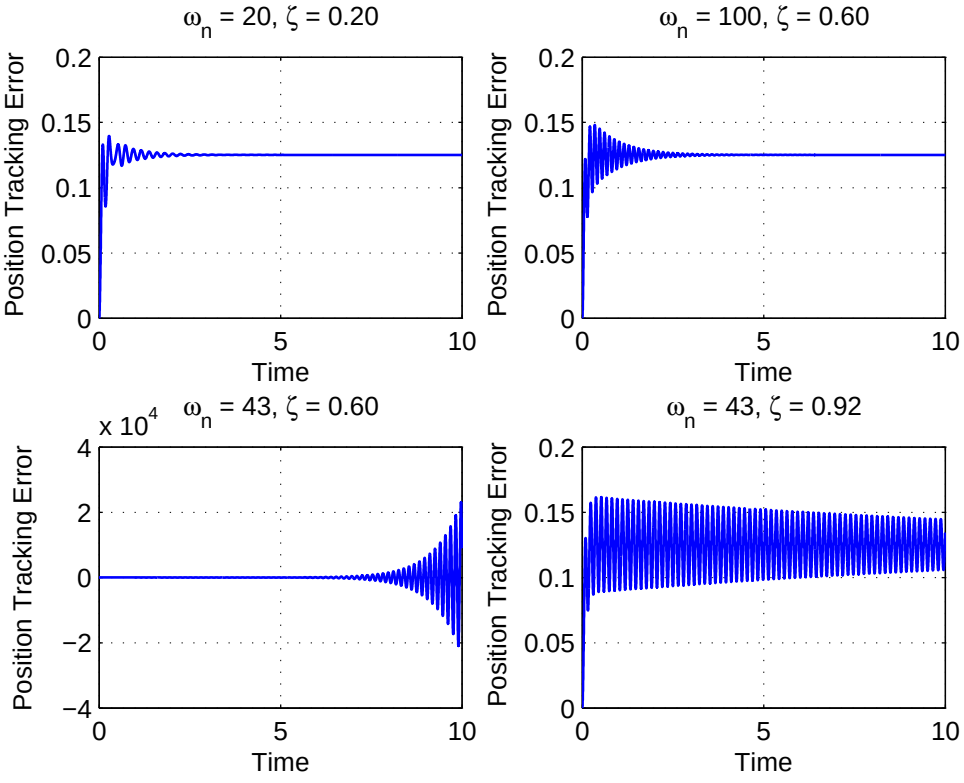


Figure 4.11: $m = 1$, $b = 0.1$, $\tau = 0.05$, $K_p = 400$, $K_d = 40$

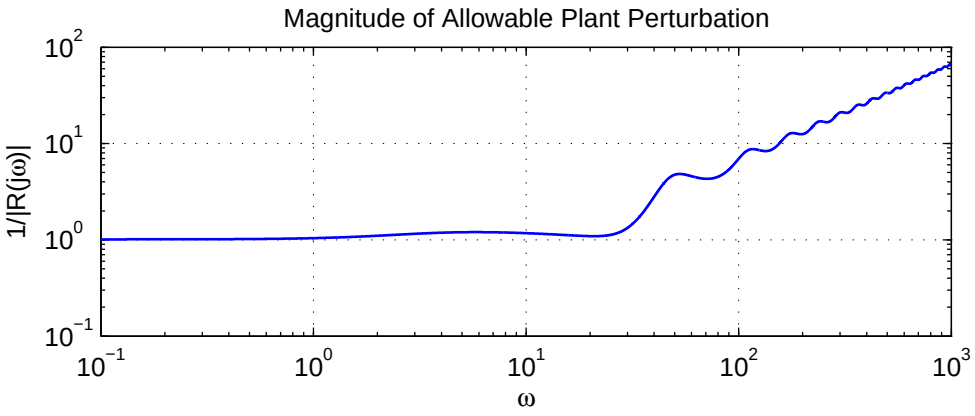


Figure 4.12: $m = 1$, $b = 0.1$, $\tau = 0.05$, $K_p = 85$, $K_d = 15$

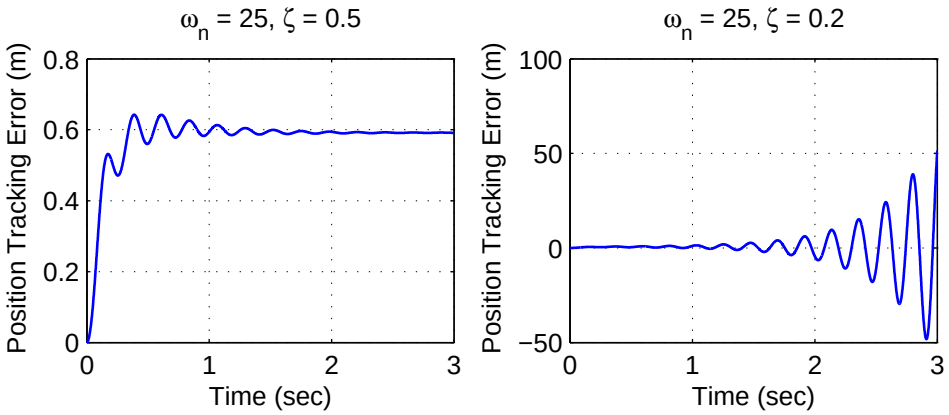


Figure 4.13: $m = 1$, $b = 0.1$, $\tau = 0.05$, $K_p = 85$, $K_d = 15$

# Chapter 5

# Controller Switching Between Free and Restricted Motion

According to ideal haptic system definition, position tracking error of the system should be as small as possible while realistic force feedback is felt at haptic side. In order to design such a controller, dynamics of the system must carefully be analysed to understand how haptic and virtual objects are following each other and what users perceive at haptic side.

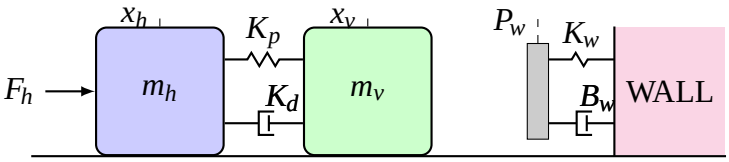


Figure 5.1: Mass-Spring-Damper System

Recall that when time delays are ignored, dynamic equations of linearized system can be written as follows:

$$m_h\ddot{x}_h(t) + b_h\dot{x}_h = -F_1(t) + F_h(t) \tag{5.1}$$

$$m_v\ddot{x}_v(t) + b_v\dot{x}_v = -F_2(t) + F_e(t) \tag{5.2}$$

where

$$F_1(t) = K_{p_1}(x_h(t) - x_v(t)) + K_{d_1}(\dot{x}_h(t) - \dot{x}_v(t)) \tag{5.3}$$

$$F_2(t) = K_{p_2}(x_h(t) - x_v(t)) + K_{d_2}(\dot{x}_h(t) - \dot{x}_v(t)) \tag{5.4}$$

For the moment, let us ignore stability conditions, and concentrate on the perception of force feedback in haptic side (for this analysis time delays can be ignored without loss of generality). Let's consider $K_{p_1} = K_{p_2} := K_p$ and $K_{d_1} = K_{d_2} := K_d$ as in Chapter 2.2. Then, according to (5.3) and (5.4), equations (5.1) and (5.2) are analogues to mass-spring-damper mechanical system in Figure 5.1. A similar approach to haptic system, considering it as a mass-spring-damper, is also mentioned in [9][6].

The imaginary spring in Figure 5.1 is zero-length. Environmental force $F_e(t)$ is defined as follows:

$$F_e(t) = K_w(x_2(t) - x_{wall}) + B_w\dot{x}_2(t). \tag{5.5}$$

If stiffness of wall $K_w$ is larger than $K_p$, it leads to a steady state position tracking error as in Figure 5.2, since $F_h(t)$ elongates imaginary spring between haptic and virtual object as much as it can (see Figure 5.3).

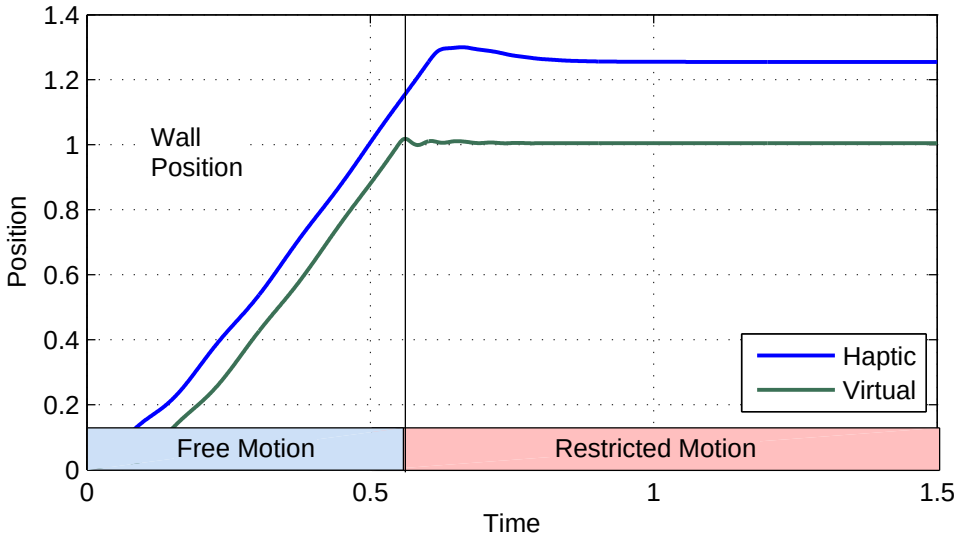


Figure 5.2: PD Controller Simulation

User perceives controller output $F_1(t)$ as force feedback so that $K_{p_1}$ and $K_{d_1}$ should be updated to obtain realistic force feedback. In free motion case, user should perceive no counter force, so controller parameters must be as small as possible. In restricted motion, high gains needed to get realistic render of environmental forces in haptic side. These parameters can be chosen from $H_\infty$ optimal parameters table. Therefore, we propose the switched controller scheme defined in equation (5.6).

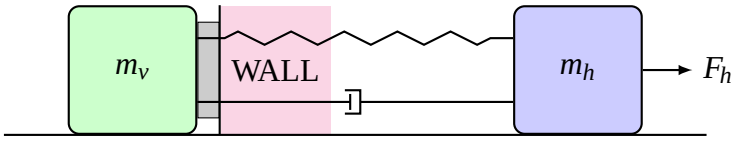


Figure 5.3: Elongation of Imaginary Spring

$$
\begin{aligned}
K_{low} &= \{40,10\},\ K_{high} = \{400,40\} \\
\phi(t) &= \begin{cases} \phi(t_k) & \text{if} \quad t < t_k + T \\ 0 & \text{if} \quad x_v \geq P_w \text{ and } t \geq t_k + T \\ 1 & \text{if} \quad x_v < P_w \text{ and } t \geq t_k + T \end{cases} \qquad t \in [t_k, t_{k+1}) \\
\{K_p, K_d\} &= K_{low}\phi(t) + K_{high}(1-\phi(t))
\end{aligned} \tag{5.6}
$$

where $\phi(t)$ is relay function that makes the controller switch between low and high gains, $t_k$ is the time values where switching occurs, $T$ is the time period, where switching is not allowed. It is called dwell time.

Corresponding simulation results are shown in Figures 5.4-5.8. When there is no dwell time, system can face problem, namely fast switching (chattering), which can damage the system. Oscillating behaviour of chattering can be seen in Figures 5.4 and 5.8. Time-domain simulations in Figures 5.5, 5.6,and 5.7 show that as delay increases required dwell time increases as well.

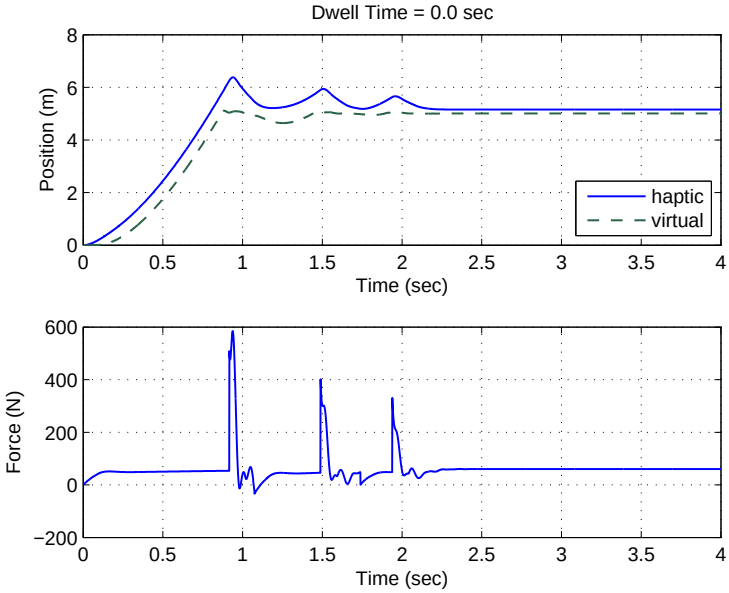


Figure 5.4: Switching Control Simulation for $m = 1, b = 0.1, \tau = 0.05$

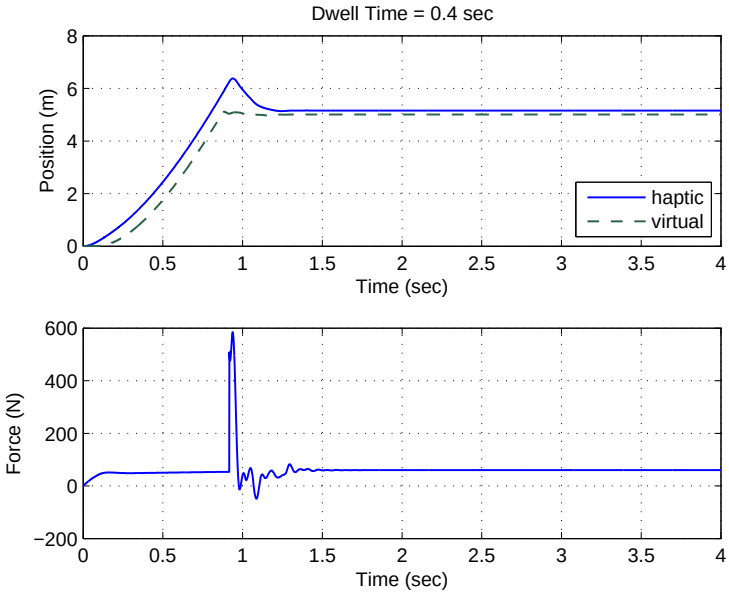


Figure 5.5: Switching Control Simulation for $m = 1, b = 0.1, \tau = 0.05$

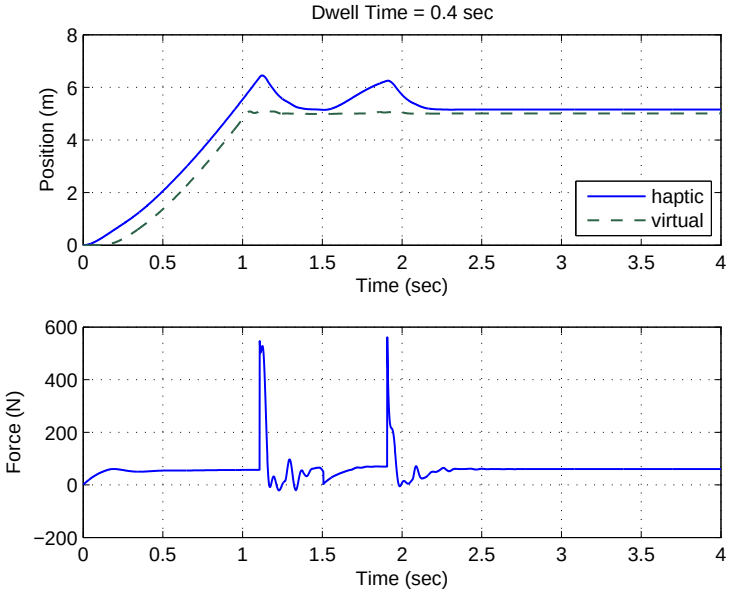


Figure 5.6: Switching Control Simulation for $m = 1, b = 0.1, \tau = 0.08$

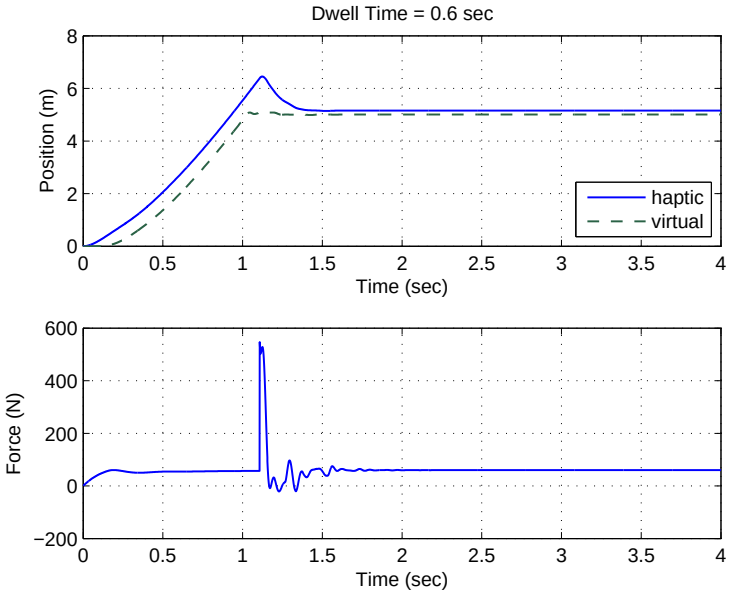


Figure 5.7: Switching Control Simulation for $m = 1, b = 0.1, \tau = 0.08$

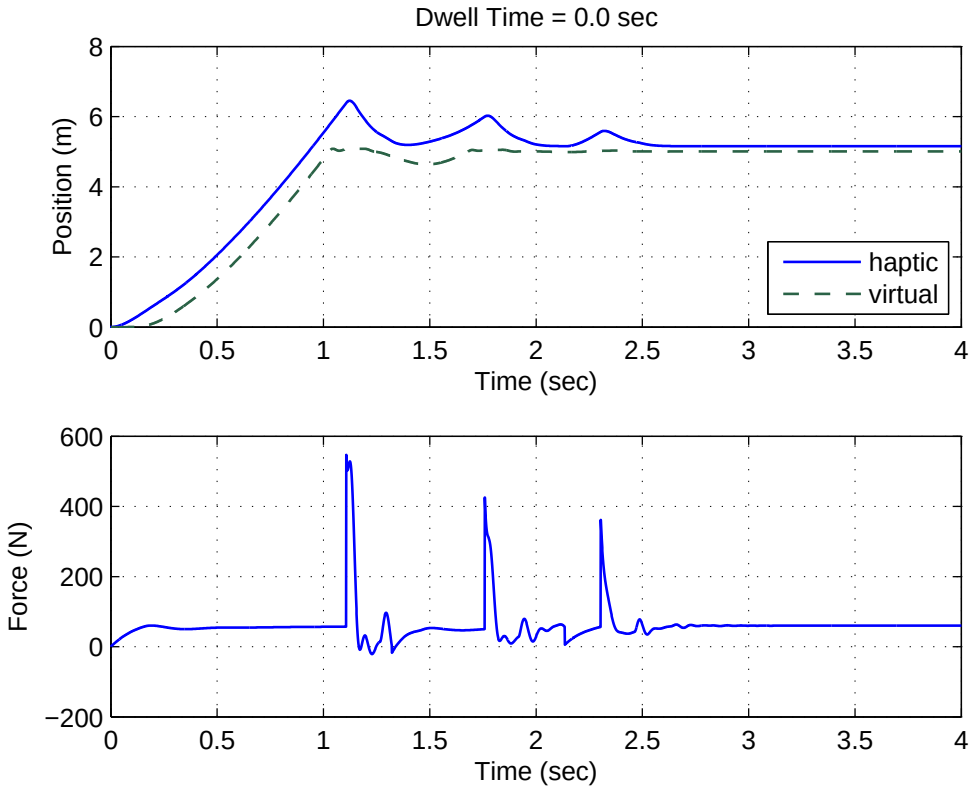


Figure 5.8: Switching Control Simulation for $m = 1, b = 0.1, \tau = 0.05$

# Chapter 6

# Experimental Validation

## 6.1 Experimental Setup

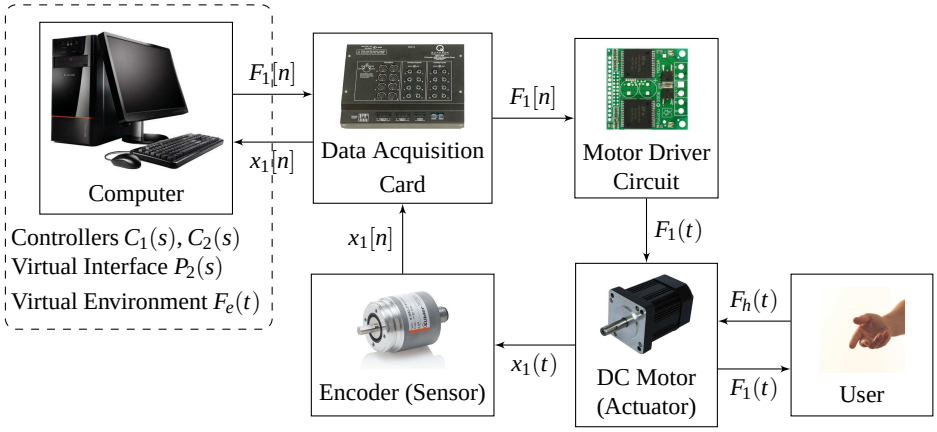


Figure 6.1: Haptic System Experimental Setup Scheme

A 1-DOF experimental setup, as in Figure 6.1, is prepared to verify the results obtained in Chapters 2-5. A computer is used to simulate virtual environment, virtual object and delays. Also haptic and virtual controllers are implemented on the same computer. Haptic interface includes a DC motor, a motor driver circuit, an encoder and a link coupled to DC motor. Computer and haptic interface communicates through data acquisition card. Realization of system can be seen in Figure 6.4.

All the controllers, virtual environment runs on a Mathworks Simulink Program. The Simulink model designed for this purpose is shown below.

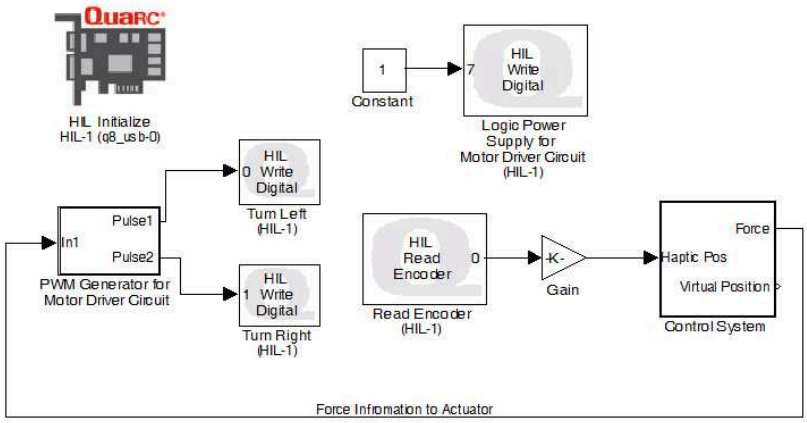


Figure 6.2: Experimental Setup Simulink Main Block

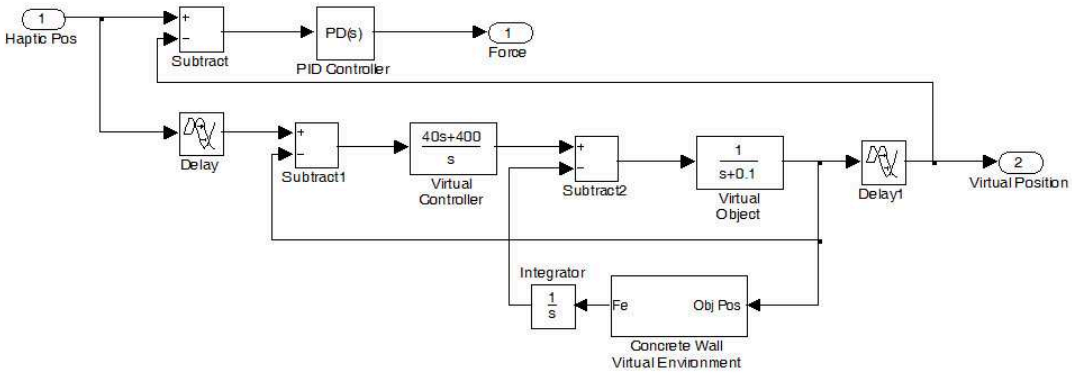


Figure 6.3: Controller System Simulink Block

## 6.2 Experimental Results

First, experiments are conducted for PD controller without switching. Then, switching control is performed.

In Figure 6.5, free motion case is tested for $K_p = 85$ and $K_d = 15$.

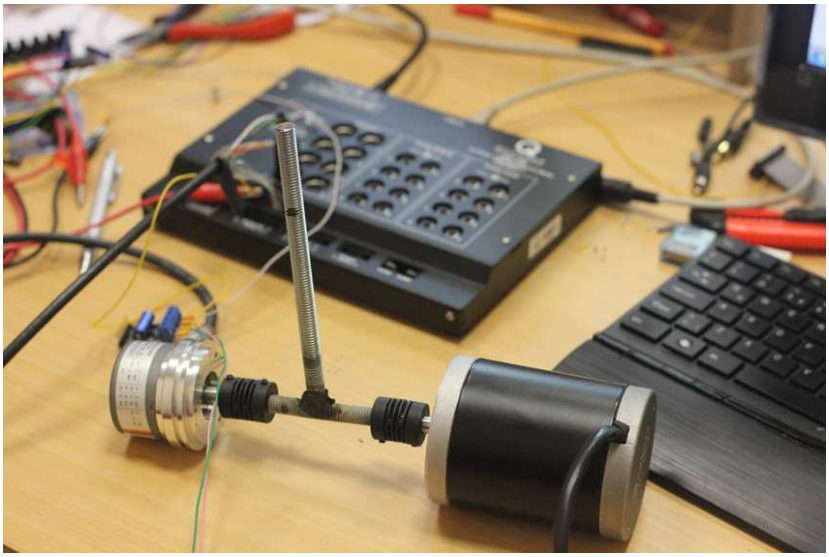

Figure 6.4: Experimental Setup

Position tracking between haptic and virtual interface is good, system is robustly stable. Low force-feedback is felt in free motion as desired. In Figure 6.6, restricted motion is tested with same gains. In all the experiments, wall position $P_w$ is at $-11$ deg.

In restricted motion, system has to have high gains in order to render realistic force reflections. For $K_p = 85$, system can not render stiff objects such as walls etc. as expected. As we increase gains, we obtain better performance in restricted motion. Experimental results for $K_p = 400$ and $K_d = 40$ are presented in Figure 6.7 and 6.8. We obtained stiffer response. However, we desire to feel low force feedback in free motion. Our performance in free motion is getting worse as we increase gains.

In order to increase performance when system is running in both free and restricted motion, switching control is used. Control system automatically switches from low to high gains as virtual object hits the wall as in Equation (5.6). As it seen in Figure 6.9, force feedback is low in free motion and high in restricted motion. Simulations show system response of switching control is closer to ideal haptic system response compared to non-switching PD controller.

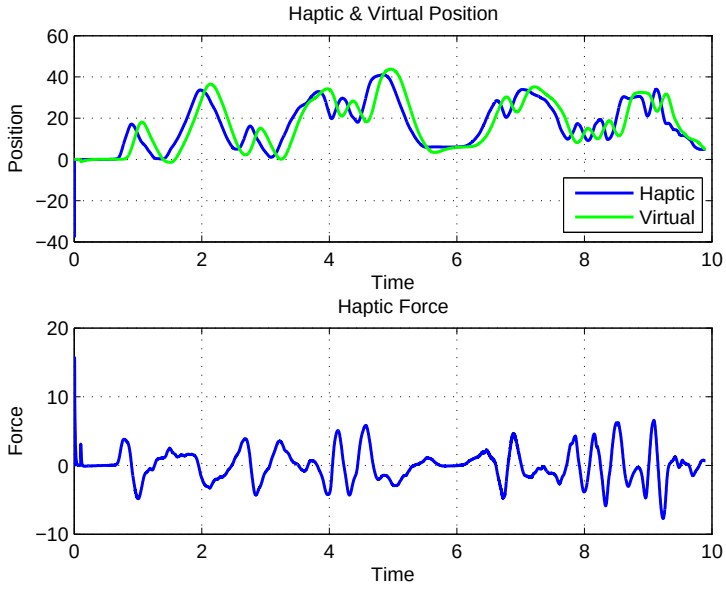


Figure 6.5: Free Motion Case for $K_p = 85$, $K_d = 15$

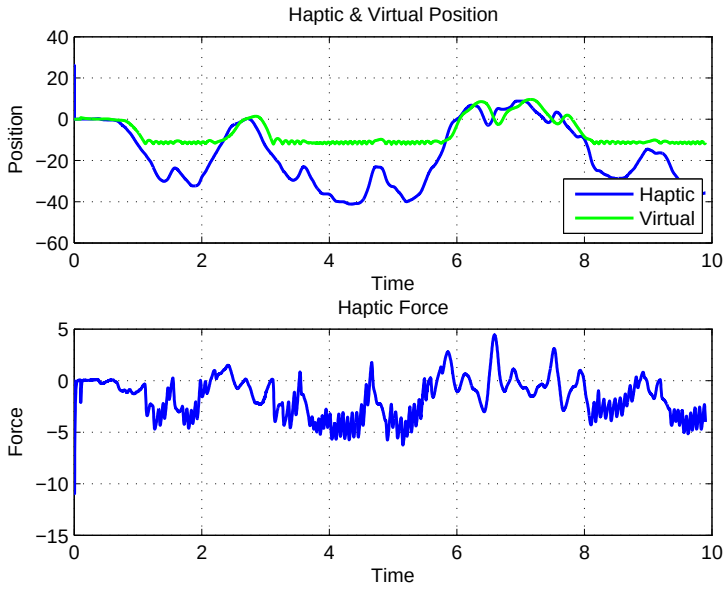


Figure 6.6: Restricted Motion Case for $K_p = 85$, $K_d = 15$

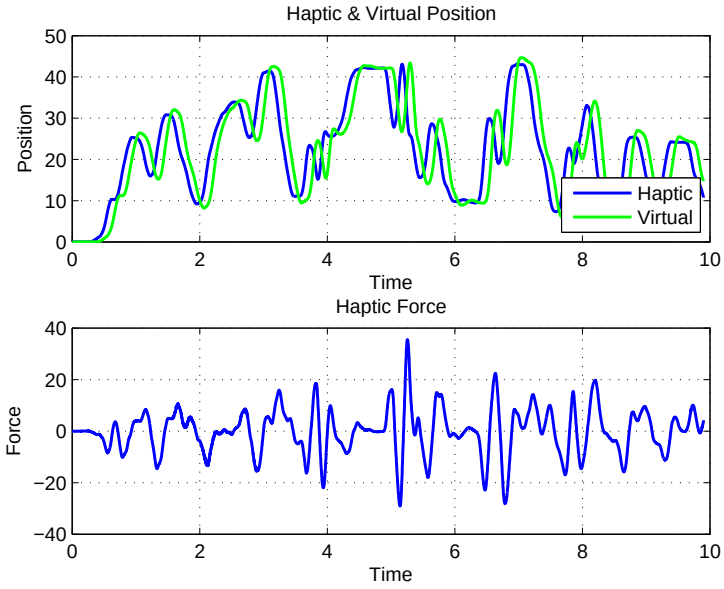


Figure 6.7: Free Motion Case for $K_p = 400$, $K_d = 40$

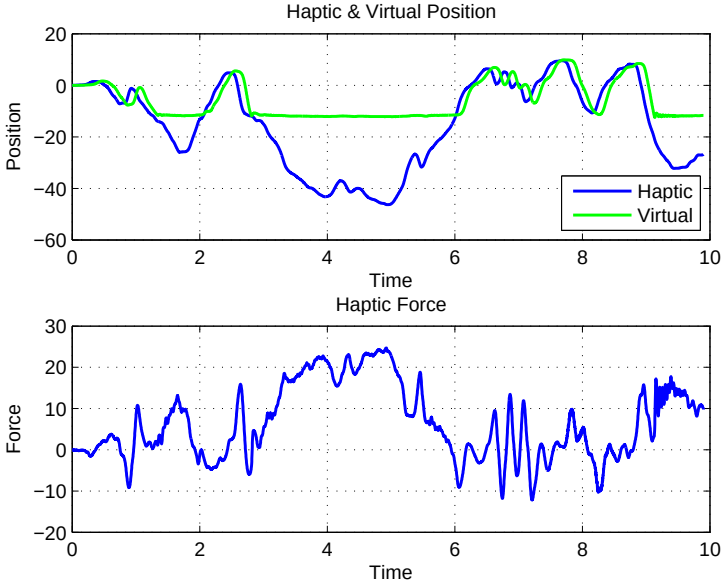


Figure 6.8: Restricted Motion Case for $K_p = 400$, $K_d = 40$

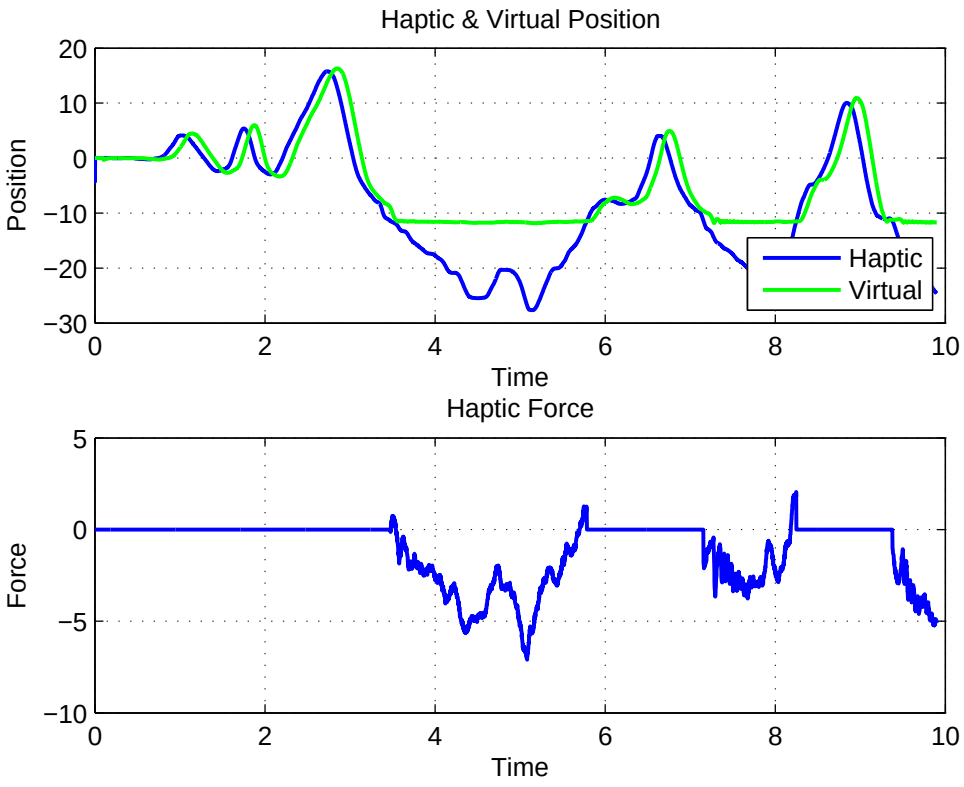


Figure 6.9: Switching Control Experiment Results

# Chapter 7

# Conclusions

The main contribution of this thesis has been a complete stability analysis for a bilateral haptic system, which is coupled to a virtual environment and affected by time-delays. This is summarized in Theorem 1 in Chapter 2, and illustrated in Figure 2.3 for two different time-delays and specific choices of plant parameters.

Once the stability region is identified, by using two different optimization techniques, optimal controller parameters are calculated. One of these optimization techniques uses $H_\infty$ based cost function; and the other one uses a stability margin optimization. The results are illustrated with typical choices of plant parameters in Tables 3.1 and 3.2.

Robustness of the designed control systems are analysed from three different perspectives:

- Robustness to uncertainties in the values of time-delays (maximal allowable time-delay has been computed, see (4.1)).
- Robustness to uncertainties in the plant parameters (region of allowable plant parameters $m_1$ and $b_1$ are calculated, see Figures4.6, 4.4, 4.8, and 4.9).
- Robustness to unmodeled dynamics in the plant transfer function (critical frequency regions are identified, see Figures 4.10 and 4.12).

According to the transparency point of view, using the same PD gains is not sufficient to satisfy performance criterion due to stiffness problems. System can not render stiff objects when low gains are used. On the other hand, high gains make operator to encounter undesired viscous friction during free motion. In order to guarantee performance in both cases (free motion and restricted motion), a simple switching strategy is tested, and it is observed that the corresponding results are closer to desired ones with switching strategy. A special attention is needed for this approach because both controllers must be updated, and since the system is affected by time delays, there is a moment when the gains will be different at each side, this may lead to unwanted fast switching (chattering). In order to avoid this, we proposed a dwell time based switching; simulation results show the effectiveness of this approach. On the other hand, theoretical stability analysis for this approach would fall into the framework of switched time delays systems. Finding the minimal dwell time guaranteeing stability is currently an open problem, see for example [15, 17, 18] and their references.

For verification of all of the results mentioned above, a 1-DOF experimental set-up has been built. Experiments conducted supported our theoretical results.

Possible future work in the lines of this thesis include the following studies:

- **Dwell Time Analysis:** Finding the optimal dwell time in considerations of robustness and performance issues is an interesting open problem. This study would involve simultaneous design of the controller parameters and the dwell time used in the switching scheme.
- **Collisions with Various Objects:** In this thesis, we only considered collisions to hard walls. Simulations and experiments can be conducted for objects with various stiffness (for example, different kind of human tissues for medical robotic applications). Such a work requires a new switching algorithm (perhaps, a continuous mapping from one set of controller parameters to the other set).
- **Improving Mathematical Model:** In this thesis, we have ignored high order dynamics, which may be present due to motor characteristics and possible flexible robot links. A possible future study would consider such high-order plant models for stability analysis.

- **Extension to MIMO case:** In this study, we considered SISO (single input, single output) plant and controller models. Extensions to MIMO (multiple input, multiple output) case where $m$, $b$, $K_p$, $K_d$ are matrices and time-delay is replaced by delay matrices would be an interesting/challenging problem. Three dimensional experimental set-up would provide results for more realistic applications.

# Bibliography


[1] R. J. Adams and B. Hannaford. Control Law Design for Haptic Interfaces to Virtual Reality. *IEEE Transactions on Control Systems Technology*, vol. 10, pp. 3-13, 2002.

[2] R. J. Anderson and M. W. Spong. Asymptotic Stability For Force Reflecting Teleoperators with Time Delays. *IEEE International Conference on Robotics and Automation*, vol. 3, pp. 1618-1625, 1989.

[3] A. Aziminejad, M. Tavakoli, R. V. Patel, and M. Moallem. Stability and Performance in Delayed Bilateral Teleoperation: Theory and Experiments. *Control Engineering Practice*, vol. 16, pp. 1329-1343, 2008.

[4] R. Baumann and R. Clavel. Haptic interface for virtual reality based minimally invasive surgery simulation. *in Proceedings, IEEE International Conference on Robotics and Automation*, Leuven Belgium, pp. 381–386, 1998.

[5] C.V. Edmond et al. ENT endoscopic surgical training simulator. *Global Healthcare Grid*, pages 518–528, 1997.

[6] N. Diolaiti, G. Niemeyer, and N. A. Tanner. Wave Haptics: Building Stiff Controllers from the Natural Motor Dynamics. *The International Journal of Robotics Research*, vol. 26, pp. 5-21, 2007.

[7] P. Hokayem and M. Spong. Bilateral teleoperation: An historical survey. *Automatica*, vol. 42, pp. 2035-2057, 2006.

[8] W. S. Kim, B. Hannaford, and A. K. Bejczy. Force reflection and shared compliant control in operating telemanipulators with time delay. *IEEE Transactions on Robotics and Automation*, vol. 8, pp. 176-185, 1992.

[9] D. Lee and M. Spong. Passive Bilateral Teleoperation With Constant Time Delay. *IEEE Transactions on Robotics*, vol. 22, pp. 269-281, 2006.

[10] G. M. H. Leung, B. A. Francis, and J. Apkarian. Bilateral Controller for Teleoperators with Time Delay via $\mu$-Synthesis. *IEEE Transactions on Robotics and Automation*, vol. 11, pp. 105-116, 1995.

[11] B. Liacu, A. T. Koru, H. Özbay, I. S. Niculescu, and C. Andriot. Low-Order Controller Design for Haptic Systems under Delayed Feedback. *Proceedings of the 10-th IFAC Workshop on Time Delay Systems*, Boston USA, pp. 43-48, June 2012.

[12] I. C. Morarescu, I. S. Niculescu, and K. Gu. The geometry of stability crossing curves of PI controllers for SISO systems with I/O delays. *Revue Roumaine De Mathmatiques Pures et Appliques*, vol. 55, pp. 297-313, 2010.

[13] G. Niemeyer and J. J. E. Slotine. Telemanipulation with time delays. *International Journal of Robotics Research*, vol. 23, pp. 873-890, 2004.

[14] E. Nuno, R. Ortega, N. Barabanov, and L. Basanez. A Globally Stable PD Controller for Bilateral Teleoperators. *IEEE Transactions on Robotics*, vol. 24, pp. 753-758, 2008.

[15] S.Y. Çalışkan, H. Özbay, and S. I. Niculescu. Stability analysis of switched systems using Lyapunov-Krasovskii functionals. *Preprints of the 18th IFAC World Congress*, Milano(Italy), pp. 7492–7496, September 2011.

[16] N. A. Tanner and G. Niemeyer. Improving Perception in Time-delayed Telerobotics. *International Journal of Robotics Research*, vol. 24, pp. 631-644, 2005.

[17] P. Yan and H. Özbay. Stability analysis of switched time-delay systems. *SIAM J. Control and Optimization*, vol. 47, pp. 936–949, 2008.

[18] P. Yan and H. Özbay. Dwell Time Optimization in Switching Control of Parameter Varying Time Delay Systems. *Proc. of the 50th IEEE Conference on Decision and Control*, Shangai PRC, pp. 4909-4914, December 2011.

# Appendix A

# Mathematical Derivations

Let's present derivation of phase and magnitude identities of the following delay term:

$$\begin{aligned} f(j\omega) &= \frac{1-e^{-j\omega h}}{j\omega} \\ &= \frac{1-cos(\omega h)+jsin(\omega h)}{j\omega} \\ &= \frac{sin(\omega h)}{\omega} - j\frac{(1-cos(\omega h))}{\omega} \end{aligned} \tag{A.1}$$

used in Equation (2.23). Phase of equation (A.1) is defined as follows:

$$\angle f(j\omega) = tan^{-1}\left(\frac{cos(\omega h)-1}{sin(\omega h)}\right) \tag{A.2}$$

$$\tag{A.3}$$

By using half-angle formulas, equation (A.2) becomes,

$$\angle f(j\omega) = tan^{-1}\left(\frac{cos^2(\omega h/2)-sin^2(\omega h/2)-1}{2sin(\omega h/2)cos(\omega h/2)}\right) \tag{A.4}$$

Simplification continues with trigonometric identity $cos^2(\omega h/2) = 1 - sin^2(\omega h/2)$,

$$\begin{aligned} \angle f(j\omega) &= tan^{-1}\left(\frac{-2sin^2(\omega h/2)}{2sin(\omega h/2)cos(\omega h/2)}\right) \\ &= tan^{-1}\left(\frac{-sin(\omega h/2)}{cos(\omega h/2)}\right) \\ &= tan^{-1}(tan(-\omega h/2)) \\ &= -\frac{\omega h}{2}. \end{aligned} \tag{A.5}$$

□

Magnitude of Equation (A.1) is defined as follows:

$$\begin{aligned} |f(j\omega)| &= \sqrt{\frac{sin^2(h\omega) + (1 - cos^2(h\omega))^2}{\omega^2}} \\ &= \sqrt{\frac{sin^2(h\omega) + cos^2(h\omega) - 2cos(h\omega) + 1}{\omega^2}} \\ &= \sqrt{\frac{2(1 - cos(h\omega))}{\omega^2}}. \end{aligned} \tag{A.6}$$

(A.7)

By using half-angle formula $cos(h\omega) = cos^2(h\omega/2) - sin^2(h\omega/2)$, Equation (A.6) becomes:

$$\begin{aligned} |f(j\omega)| &= \sqrt{\frac{2(1 - (cos^2(h\omega/2) - sin^2(h\omega/2)))}{\omega^2}} \\ &= \sqrt{\frac{sin^2(h\omega)/2}{h\omega/2}} \\ &= \frac{sin(h\omega)/2}{h\omega/2} \end{aligned} \tag{A.8}$$

□

# Appendix B

# Simulink Models

## B.1 PD Control

Figures 4.1-4.3,4.5,4.7, and 5.2 are generated by using the Simulink model below.

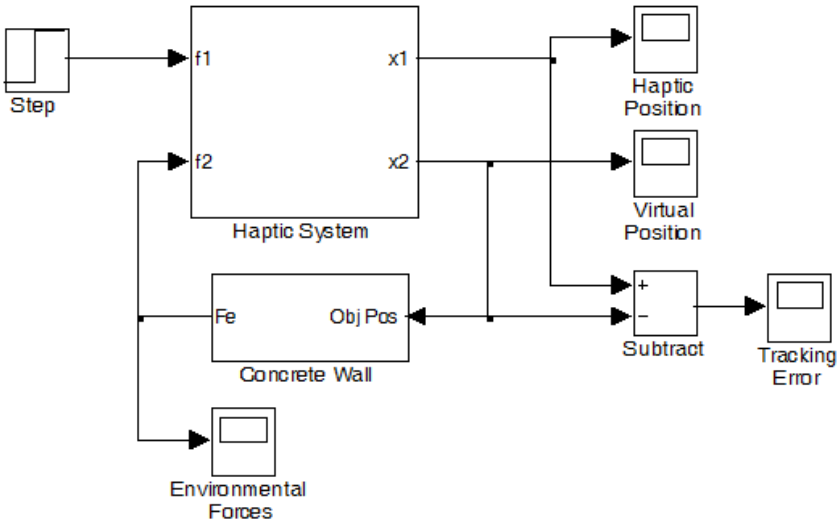


Figure B.1: PD control Simulink main block

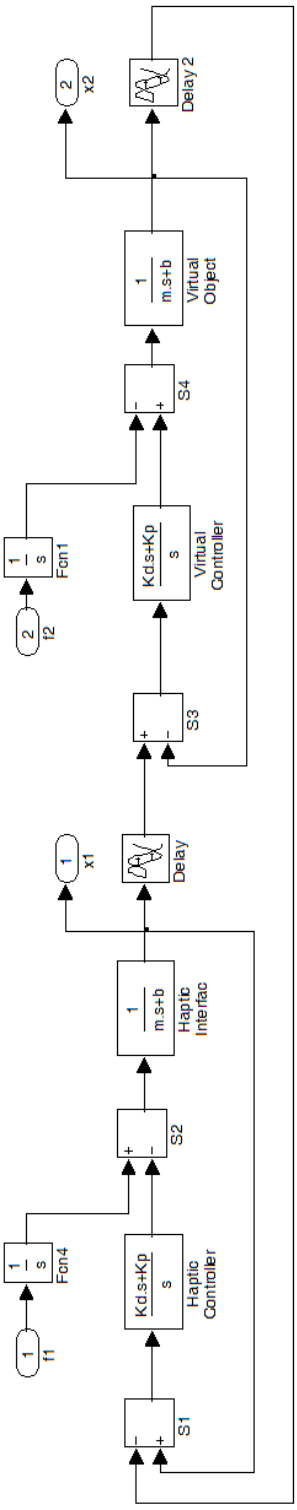


Figure B.2: Look under mask to Haptic System

## B.2 Unmodeled Dynamics Simulations

Haptic Interface plant in Simulink Program in B.2 is replaced with Atomic-subsystem below to realize simulations to verify our results in Chapter 4.3. Figures 4.11 and 4.13 are generated by using this Simulink model.

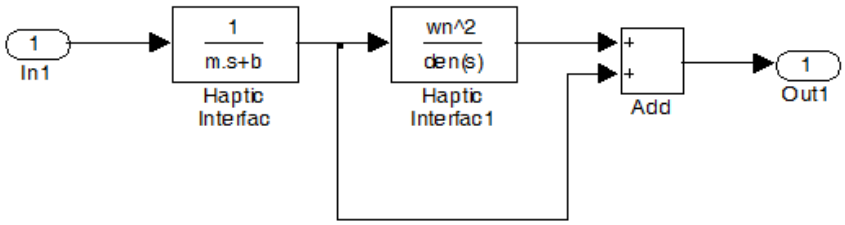


Figure B.3: Perturbed Plant Atomic-Subsystem

## B.3 Switching Control

Figures 5.4-5.8 are generated by using the Simulink model below.

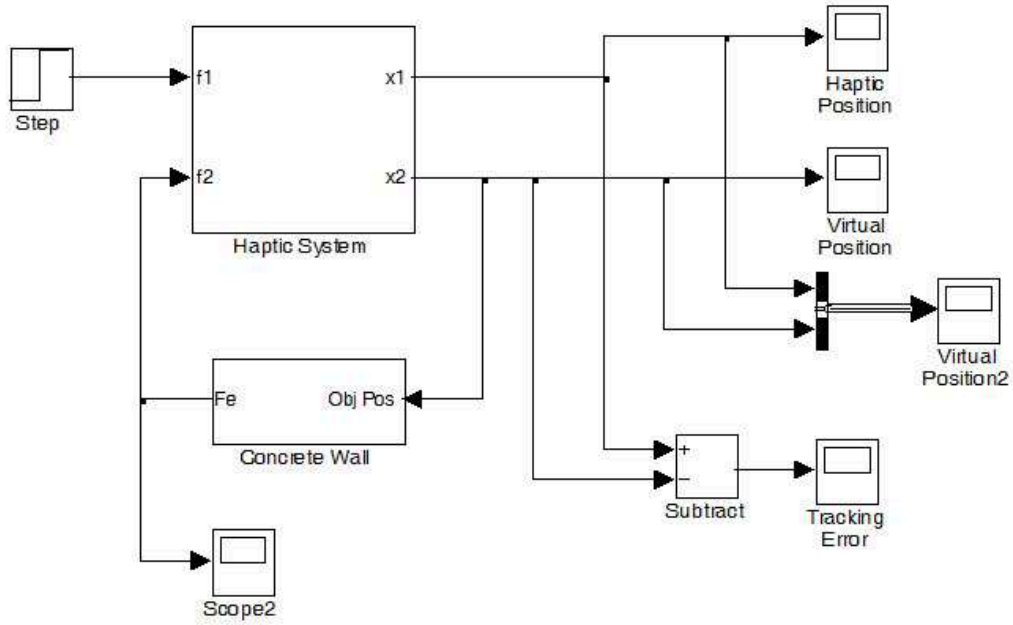


Figure B.4: Switching control Simulink main block

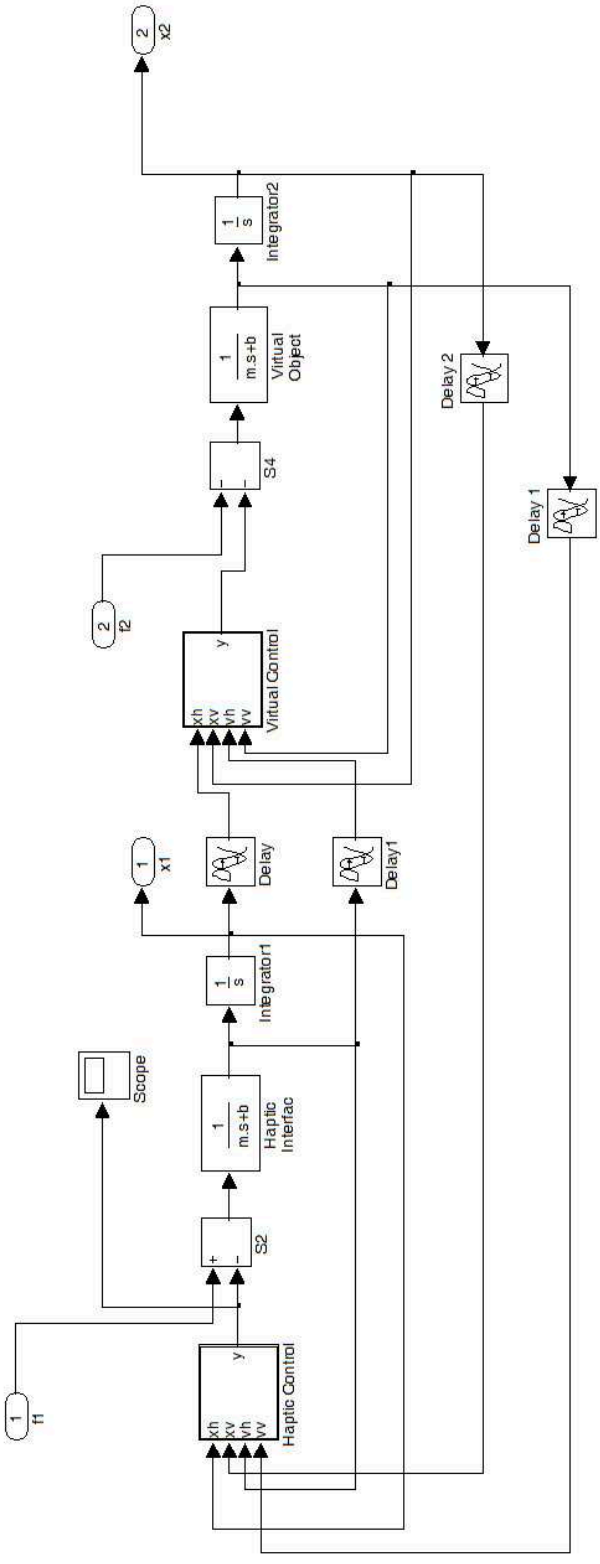


Figure B.5: Look under mask to Haptic System

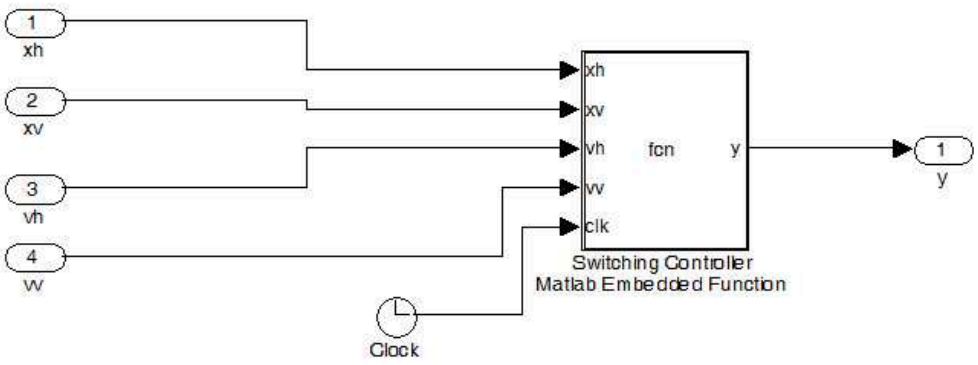


Figure B.6: Look under mask to Haptic Controller

### B.3.1 Matlab Embedded Code

```
%%Embedded Matlab Function code for 'Switching Controller'
function y = fcn(xh,xv,vh,vv,clk)
persistent mode cs; %0 free, 1 restricted sol
if(isempty(mode))
    mode = 0;
    cs = 0;
end

dwell=0.4;
if(mode == 0 && (clk-cs)>dwell)
    if(xv >= 5)
        mode = 1;
        cs = clk;
        [mode,clk]
    end
end
if(mode == 1 && (clk-cs)>dwell)
    if(xv < 5)
        mode = 0;
```

```
        cs = clk;
        [mode,clk]
    end
end

m = 1; b = 0.1;
if(mode == 0)
   Kp = 85; Kd = 15;
else
    Kp = 400;
    Kd = 40;
end
   y = -Kp*(xh - xv) - Kd*(vh - vv);
```

# Appendix C

# Matlab Codes

## C.1 Stability Regions

Figure 2.3 is generated by using following Matlab code.

```
clear all;
h = 0.005;
m = 1; b= 0.1;
w = logspace(-2,6,1000);
alpha = logspace(-8, 0,2500);
for (k = 1:length(alpha))
   for (l = 1:length(w))
      gc(l) = pi - 2*(atan(w(l)) - atan(alpha(k)*w(l))) - h*w(l);
   end
   [mingc,ind] = min(abs(gc));
   wo(k) = w(ind);
   L(k) = (2*(1-alpha(k)))/(wo(k)^2+1);
end
Kp = b^2./(m*L);
Kd = alpha.*Kp*m/b;
loglog(Kp,Kd);
```

## C.2 $H_\infty$ Based Optimization

Optimal gain parameters in Table 3.1 is obtained by using following Matlab code.

```
function [T,pointKd,pointKp] = infnorm(Kdbmin, K0max)
m = 1; b = 0.1; rho = 5000;
tao = 0.05; h = tao*b/m;
length(Kdbmin)
for(p = 1:length(Kdbmin))

    Kd0 = Kdbmin(p);
    Kprange = logspace(log10(Kd0+0.5),log10(0.95*K0max(p)),100);
    p
    for(n = 1:length(Kprange))
        Kp0 = Kprange(n);
        w = logspace(-5,5,2000);
        for(k = 1:length(w))
            s = i*w(k);
            G = (Kp0 + Kd0*s)/(1 + s);
            F = inv(s + G*(1 + exp(-h*s)));
            Tc(k) = max(svd(F * [rho*(m/b^2)*inv(1+s); ...
            s*G / (s+G*(1 - exp(-h*s)))]));

        end
        [ind,ind] = max(Tc);
        wf = linspace(w(ind)*.95,w(ind)*1.05,100);
        for(l = 1:length(wf))
            sf = i*wf(l);
            Gf = (Kp0 + Kd0*sf)/(1 + sf);
            Ff = inv(sf + Gf*(1 + exp(-h*sf)));
            Tf(l) = max(svd(Ff * [rho*(m/b^2)*inv(1+sf); ...
            sf*Gf / (sf+Gf*(1 - exp(-h*sf)))]));
        end
        T(p,n) = max(Tf); pointKd(p,n) = Kd0; pointKp(p,n) = Kp0;
    end
end
end
```

## C.3 Stability Margin Optimization

Optimal gain parameters in Table 3.2 is obtained by using following Matlab code.

```
clear all
h=0.005; b=0.1;
rho1=0.01; rho2= [10:10:100];
w = logspace(-3,4,10000);
alpha = logspace(-4,-0.01,5000);

for qq=1:length(rho2)
    for kk=1:length(alpha)
        for k=1:length(w)
            e(k)=abs(pi-2*(atan(w(k))-atan(alpha(kk)*w(k)))-h*w(k));
        end
        [mine,mm]=min(e);
        wo(kk)=w(mm);
        x=sqrt(1+wo(kk)^2);
        K0(kk)=(1+wo(kk)^2)/(2*(1-alpha(kk)));
        cost(kk)=(1/rho1)*(rho2(qq)/x + (b^2/rho2(qq))*x/(2*(1-alpha(kk))));
    end

    [mincost,nn]=min(cost);
    KPopt(qq)=(b^2/rho1)*sqrt((1+wo(nn)^2)/(2*(1-alpha(nn))));
    KDopt(qq)=(1/b)*KPopt(qq)*alpha(nn);
    GM1(qq)=(rho1*sqrt((1+wo(nn)^2)));
end

[KPopt', KDopt']
```

## C.4 Allowable Perturbations of Delay

Maximum tolerable delay values in Table 4.1 are obtained by using following Matlab code.

```
m = 1; b = 0.1;
Kpl = [17.1,85.0,246,305,310,400];
Kdl = [10.2,15.2,43,55,51,40];

for(k = 1:length(Kpl))
    Kp = Kpl(k); Kd = Kdl(k);
    w0 = sqrt(2/b^2*(m*Kp-b*Kd)-1);
    tao = (pi-2*(atan(w0)-atan(b*Kd/(m*Kp)*w0)))/w0*2*m/b/2;
    ['for Kp = ', num2str(Kp), ', Kd = ', num2str(Kd), ...
    '; critic tao = ', num2str(tao),'percentage = ', ...
    num2str((tao-0.05)*100/0.05)]
end
```

## C.5 Allowable $m_1$ and $b_1$ Parameters

Figures 4.6, 4.4, 4.8, and 4.9 are generated by using following Matlab code.

```
clear all
Kp = 246; Kd = 43; tao = 0.115;
m = 1; b = 0.1; cnt = 1;
w = logspace(-3,3,15000);
for(k = 1:length(w))
    wk = w(k); jw = j*w(k);
    G2 = (Kp+Kd*jw)/(m*jw^2+b*jw);
    C1 = (Kp+Kd*jw);
    A = ((1+G2-G2*exp(-2*tao*jw))*C1)/(jw*(-1-G2));
    b1 = real(A); m1 = imag(A)/wk;
    USS(k,:) = [m1,b1];
    if(m1 >= 0 & b1 >= 0)
       US(cnt,:) = [m1,b1];
       cnt = cnt + 1;
    end
end
m = US(:,1); b = US(:,2);
mass = [0:0.01:4];
friction = spline(m,b,mass);
```

```
area(m,b);
axis([0 1.5 0 1.5]); grid on;
title(['\tau = ',num2str(tao)])
hold on;
plot(1,0.1,'.','MarkerSize',18);
```

## C.6 Robustness Against Unmodeled Dynamics

Figures 4.10, and 4.12 are generated by using following Matlab code.

```
clear all;
m = 1; b = 0.1; h = 0.05;
Kp = 400; Kd = 40;

w = logspace(-1,3,10000);
for(k = 1:length(w))
    wk = w(k); jw = j*wk;
    P = 1/(jw*(jw*m+b)); C = Kp + Kd*jw;
    T = P*C/(1+P*C); G = jw*P*C;
    ft = (1-exp(-h*jw))/jw; f2t = (1-exp(-h*2*jw))/jw;
    W(k) = inv(T/(1+T*exp(-h*jw))*(1+G*f2t)/(1+G*ft));
end
loglog(w,abs(W),'LineWidth',1.0)
title(['Magnitude of Allowable Plant Perturbation']);
xlabel('\omega')
ylabel('1/|R(j\omega)|')
grid on;
```