μ-Controller Design Framework for Haptic Telemanipulation Systems with Multiple Uncertain Sources and Time Delay

Li Li

A Thesis

in

The Department

of

Electrical and Computer Engineering

Presented in Partial Fulfillment of the Requirements

for the Degree of Master of Applied Science at

Concordia University

Montreal, Quebec, Canada

November 2004

© Li Li, 2004

Library and Archives Canada

Published Heritage Branch

395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque et Archives Canada

Direction du Patrimoine de l'édition

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence
ISBN: 0-494-04381-4
Our file Notre référence
ISBN: 0-494-04381-4

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

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

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

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

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

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

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

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

Canada

ABSTRACT

μ-Controller Design Framework for Haptic Telemanipulation Systems with Multiple Uncertain Sources and Time Delay

Li Li

The control of bilateral haptic telemanipulation systems in the presence of multiple uncertain sources including uncertain communication time-delay is indeed a non-trivial problem. Such control is typically evaluated by three criteria: stability, performance and transparency. This thesis discusses these criteria in the context of how linear fractional transformation (LFT) technique and μ-theory can be applied into the design of robust bilateral haptic telemanipulation systems. As a result, a framework of controller design for such a system based on PHANToM haptic devices is presented under the assumptions that all of the components in the system, including the communication channel with time-delay, possess uncertainty.

The proposed design framework possesses following new characteristics compared to the previous work in the literature: 1) all components in the system are potentially uncertain and the environment possesses parametric uncertainty; 2) the master controller is structured as of two-degree-of-freedom (2-DOF) to avoid poor transient performance; 3) force reflection from slave to master is realized as a function of environment reaction forces and differences between master and slave displacements to meet the system transparency requirement, and 4) the communication channel is treated as a second-order system with parametric and multiplicative uncertainties, which represents a more accurate and less conservative realization.

The analysis of the resulting controllers made in both frequency and time domain shows that the resulting system realizes well the robust stability and performance requirements that are envisaged in this problem.

Acknowledgements

I could not have completed the research work presented in this thesis without assistance of many people.

First and the most, I would like to express my deepest gratitude to Professor Khashayar Khorasani, my advisor, for his constructive advice and helpful guidance throughout my graduate studies at Concordia University. His insight, suggestion and unending patience contributed immensely to the fulfillment of this thesis.

I would also like to thank my friends, Xiang Liu, Liang Gao and Wei Wan, for their many kindly helps and suggestions during the writing of my thesis. I appreciate the time that they devoted to my thesis.

Lastly, I would like to address my special thanks to my family for their abundant love, consistent encouragement and emotional support, which made me overcome a number of difficulties and finally finish the studies.

Contents

List of Figures

Chapter 1

Introduction to Haptic Telemanipulation

Haptic telemanipulation systems are designed to enable human operators to handle slave manipulators, which are usually located in remote sites, via master manipulators, which are made up of haptic devices (also called haptic displays), to complete remote, usually dangerous or delicate tasks safely and accurately. It is well-known that the control of bilateral haptic telemanipulation systems in the presence of uncertain communication time delay and multiple sources of uncertainty which stem from system modeling or unknown system dynamics and unknown environment, is not a trivial problem.

In general, the control quality or performance of a haptic telemanipulation system is typically evaluated by three criteria: stability, performance and transparency [1-7]. Like any other control system, the stability of a haptic telemanipulation system becomes the prime requirement to be met. Indeed, in the face of its modeling and measuring errors (i.e., model uncertainties and noise uncertainty), the system should remain stable while performing various tasks in remote sites (i.e., dealing with various environments). Meanwhile, the system should possess the ability to realize the pre-specified performance requirement, such as translating well the movement at the local site (master-manipulator side) into identical or similar movement at the remote site (slave-manipulator side). In addition, how well the impedance felt at the slave is reflected to the master, or in terms of the transparency of the system, should be under the consideration.

This thesis discusses how robust control theory, which has been utilized in a number of control engineering areas and others where optimization is needed in recent years, can be applied into the design of a series of robust controllers for a haptic telemanipulation system, especially the linear fractional transformation (LFT) technique and the structured singular value (SSV) theory, which is the so-called μ-analysis and synthesis method [8-11, 29]. A framework for the controller design for such a system based on PHANToM haptic devices is put forward with the assumption that all of the components in the system possess uncertainty.

In this chapter, some of the specific terminologies and concepts that are used in the thesis are first explained and the researches available in the literature are reviewed and finally, problems considered in the thesis as well as the contribution of the thesis are presented.

1.1 Terminologies and Concepts

Bilateral haptic telemanipulation is actually a human-machine interaction. Haptic devices or haptic displays, robotics and computer aided telemanipulation in a real or virtual environment (VR) have been incorporated into an emerging area of research in recent years.

Haptic Devices / Haptic Displays

The word *haptic* implies something related to the kinesthetic and tactile human perception with hand. A device which provides haptic information to a human is called a

haptic device or a *haptic display*. It has to be in contact with human operators and gives them sensations as similar as possible to those they could perceive from the interaction with an object. Therefore, by using haptic devices, a bilateral telemanipulation system can provide a good interface for the human operator to carry out a task in a remote site or a virtual environment.

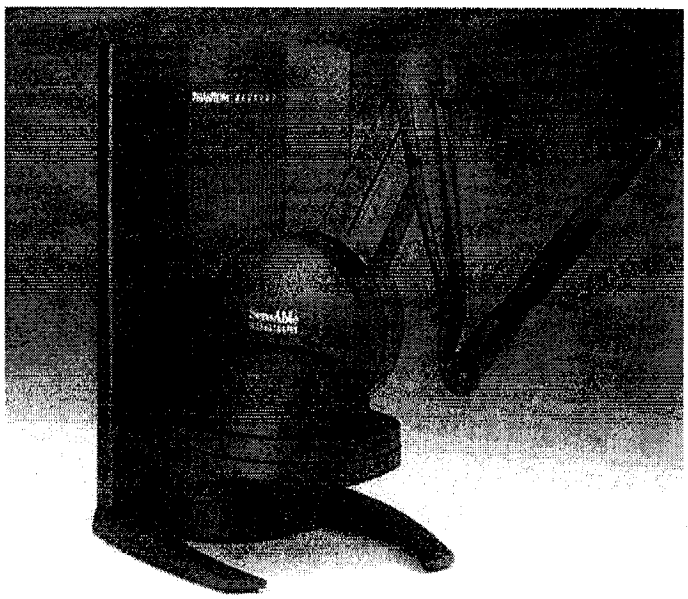


Fig. 1.1 The PHANToM Haptic Display

There exist many kinds of haptic devices which can usually produce motions and forces sensed through both touch and kinesthesia. Among them the PHANToM family of haptic devices from SensAble Technologies, shown in Figure 1, is a unique representative with commercial purposes.

The haptic devices resemble robotic manipulators [12]. They are developed to give the user a certain kind of perception of what is going on in a remote environment or a simulated virtual scene as realistic as possible. For example, they have been used in

situations where direct training in the real operation is difficult to be made such as in virtual surgery, flight simulators and etc. They are also applied in military and entertainment. In practice, they are often used as master manipulators in telemanipulation control systems to provide human operators a good means of handling the course of the execution of tasks happening at remote sites or virtual environments.

Telemanipulation System

Roughly speaking, a *telemanipulation system* consists of two devices, one local manipulator (the master) operated by a human operator and the other in a remote location (the slave), which often interacts with hazardous or complex environment, and a communication channel between the master and the slave, as shown in Figure 1.2. A force exerted by the operator on the master manipulator is translated via the communication channel with transmission delay as a displacement or motion or force at a remote site to the slave. The slave is usually a manipulator and the master is kinematically similar to it or a haptic device. It is well known that the time delay caused by the communication channel may deteriorate system performance and thus could result in tight constraints on the control performances.

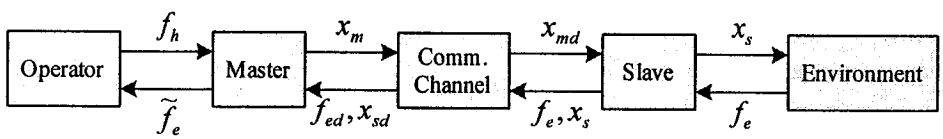


Fig. 1.2 A General Telemanipulation System

Virtual Reality (VR) System

In a *virtual reality system* the remote side does not physically exist but it is simulated with software, and a haptic device may be the only physical component, whereas in a telemanipulation system the haptic device constitutes only one part of the physical system, i.e., the master.

Time delays in the above-mentioned two systems have different characteristics: in the VR system the time delays are due to the simulation computing time whereas in the telemanipulation system the delays are due to the communication channel transmission delay.

Transparency

Transparency, also called *telepresence* sometimes, is an important objective in the controller design for bilateral telemanipulation systems. It implies the accuracy of rendering the environment to the human operator. It is understood that the transparency of a bilateral telemanipulation system is achieved if the slave displacement and force follow the master's faithfully [3, 7, 13].

1.2 Review of Control Techniques for Telemanipulation

The main objective of the controller design for a bilateral telemanipulation system is to realize a stable system with satisfying pre-specified performance requirements in the presence of time delays, disturbances, uncertainties and/or measurement noise, all of

which possess the effects of making the system unstable and must be addressed in the robust controller design. Ideally, the slave manipulator is controlled so that in the steady state its displacement or velocity is scaled or equal to the master's and the force feedback to the master is equal to the contact force generated at the remote site.

The topic of controller design for telemanipulations has been discussed for many years. There are many control methods and system architectures, which address this problem from different angles, having been proposed by various researchers. Techniques such as passivity, compliance, adaptive control, wave variable insertion, robust control theory, among others, have become the foundation of control methods in the literature.

Aimed at the transmission delay, Anderson and Spong [1] introduced a control law for telemanipulations which overcame the instability caused by the time delay. They developed a criterion by using passivity and scattering theory, and implemented the control law on a one-degree-of-freedom (1-DOF) linear time-invariant (LTI) telemanipulation system with the assumption that the master and slave manipulators were strictly passive, which constrained the design of master and slave controllers. According to Leung and Francis [14], however, the discretization process of a passive system via sample and hold will in general result in a non-passive system.

Hannaford [15] applied a hybrid two-port model into telemanipulations with force and velocity sensing at the master and slave. In his design framework, the interfaces between human operator and master, and between environment and slave, were ports through which the telemanipulators were designed to exchange energy between the operator and

the environment. However, the possibility of existing transmission time delays was not considered in this work.

Force reflection concept was widely used in the control of robot manipulators (for example, [16]), which in some control methods led to position information being sent from the master to the slave and force information being transmitted in the opposite direction if the remote interaction occurred. Kim and Hannaford [17] proposed a shared compliance control method which inserted a compliance term in the controller at the remote site to modify the behavior of the slave manipulator according to the interaction with the environment.

Niemeyer and Slotine [18, 19] extended the force reflection control concept by a damping injection term in order to guarantee the system passivity. They showed that the stability could be preserved through the systematic use of specially designed wave-variable filters. The motivation for introducing wave variables was their effect on passivity. Their scheme may also be viewed as a position error control integrated with one or more dissipative elements in order to guarantee the passivity. In a pure position error control, forces applied on both sides are proportional to the position difference between the two manipulators; position information flows in both directions of the communication channel [20].

Park and Cho [21] worked out a sliding-mode control method which offered a variable structure control defined at the slave side in order to achieve a perfect tracking of the delayed master manipulator displacement. It provided robustness against various uncertainties and can be used to deal with problems caused by transmission time delay.

In order to achieve transparency, four-channel control architecture was adopted in [2, 3, 7, 13, 22, 23], among others. In these schemes, both master and slave displacement or velocity and force information were sent to the other side in both directions, resulting in four channels which can offer perfect transparency theoretically. However in practice, the dynamics of environment and human operators compromise both performance and stability, so the controllers cannot preserve transparency and stability in the presence of system uncertainties and time delays.

Zhu and Salcudean [24, 25] put adaptive motion/force control concept into their telemanipulation control scheme. As in the four-channel control architecture, the master and slave position and force information were exchanged via communication channels. Both master and slave manipulators have their own local adaptive position/force controllers so that the parameters of the manipulators are locally estimated.

An extensive effort on robust control theory is also being conducted in telemanipulations by some researchers. One advantage of robust control theory is that it allows designers design a controller for both performance and stability margins in one step. Due to the possibility of making system unstable, the presence of system model uncertainties, disturbances, measurement noises, and time delays in the communication channel should be considered in the controller design; therefore, we believe that robust control approach is the most effective way to cope with these issues.

Colgate [5] introduced an approach to robust impedance shaping based on kinematic similarity between tasks of different scale. He treated the bilateral manipulator as a means of constructively altering the impedance of a task. Based on passivity and scattering

matrix concepts, certain robustness criteria were developed with the help of μ-synthesis and analysis theory.

Kazerooni et. al. [26, 27] utilized H_∞-control theory into the controller design for telemanipulations and shaped the relationships between positions and forces of both master and slave. With the assumption of existence of no time-delay in the communication channel, the performance requirements were described in terms of desired dynamic relationships between master and slave manipulators. Also with H_∞-control theory, Yan and Salcudean [22] took a four-channel architecture in which the control signals were based on displacements and forces. Weighting functions and noise disturbances were used to trade off between system performance and stability robustness. However, their methods did not consider uncertainties and transmission time delays.

Leung et. al. [28] focused on the time delay problem in telemanipulations through μ-synthesis framework. The system designed was stable for a pre-specified time delay while its performance was optimized. Two cases, free movement and constrained movement were considered. The design approach consists of two steps: first, controllers for the master and slave are designed in the case of free movement; and second, an outer loop controller is designed for the constrained movement with communication time-delay. However, in their controller design process, the time delay in the communication channel was viewed as a plant with a nominal model $G_{T0}(s) = 1$ and an uncertain factor $\Delta_T = e^{-Ts} - 1$, where T is the fixed time delay of the communication channel. This definitely enlarged the uncertainty of the communication channel and thus made their

method conservative. In addition, transmission delay between master and slave usually varies with direction.

Manipulator displacement information is generally of highest significance in practice and velocity stability may not guarantee displacement stability [22]. However, in Leung et. al. method [28], velocity information is transmitted from the master to the slave and force information flows in the opposite direction.

Since most of the previous research in the literature used the passivity theory, the constraints on the parameters of these systems can become more than other techniques. This has made the controller designs conservative although the stabilities of these telemanipulation systems were guaranteed. Furthermore, not many researchers have paid attention to haptic telemanipulation systems with multiple sources of uncertainties. Meanwhile, the time-delay of the communication channel is mostly treated as a constant in most of the literature. This unavoidably degrades the quality of the controller design. Given that the nominal models for the perturbed plants in the telemanipulation systems are usually derived from identification processes, in which the identification errors and unmodeled plant dynamics exist, it is unavoidable that these models cannot precisely and accurately describe the actual physical system dynamics.

Motivated by above-mentioned factors and based on LFT machinery and H_∞/μ-analysis and synthesis techniques, this thesis presents a synthesis framework for a bilateral haptic telemanipulation system with multiple sources of uncertainties including uncertain time-delays.

1.3 Contributions of the Thesis

A control scheme has to provide the system or plant to be controlled with good handling qualities over a designated specification envelope. It also has to ensure adequate robustness against unmodeled dynamics, modeling uncertainties and variations of operation point within the envelope. Acceptable noise and disturbance rejection must also be demonstrated.

The theory used in this thesis is H_∞/μ-analysis and synthesis, one of the robust control methodologies recently developed, for which an extensive set of software tools has been developed [11]. As mentioned earlier, this theory provides the designer with an inherent multivariable design approach and a direct way of incorporating uncertainty and robustness aspects into the design process. It has been shown that H_∞/μ-analysis and synthesis can accommodate the designer in dealing with control problems ranging from fairly simple to quite elaborate.

With an example of a concrete controller design process for a bilateral haptic telemanipulation system in which all components are uncertain, this thesis illustrates how the linear fractional transformation (LFT) machinery and μ-analysis and synthesis techniques are used to design three robust controllers for the system; that is, a local controller for the master manipulator, a remote controller for the slave manipulator, and a global controller for the whole system which interacts with an environment. With the LFT and μ-analysis techniques, the controller problem can be changed into μ-analysis and synthesis problems through some suitable frequency dependent weights, which define

uncertainties in the system and pre-specified control objectives. In this way, performance and stability requirements can be considered in one step.

Similar to Leung et. al. method [28], the design process presented in this thesis also includes two stages: one for the case of no interaction between slave manipulator and environment, i.e., the free movement of the slave manipulator, and the other for the case of existing interaction between the slave and the environment, i.e., the constrained movement of the slave. However, the design framework proposed in this thesis possesses the following new characteristics as compared with the Leung et. al. [28] and other methods which apply robust control theory:

- All components in the system are assumed to be potentially uncertain, which is common in practice, and the interacted environment is represented by parametric uncertainties;
- To prevent large overshoots, especially when a model reduction technique is necessarily applied to the controller design, the master controller is designed as a two-degree-of -freedom (2-DOF) controller;
- Force reflection from the slave to the master is realized through environment force reaction and difference between the master and the slave displacements;
- Time delays stemmed from the communication channel is treated as a second-order system with parametric and multiplicative uncertainties, which is much more accurate and of much less conservative nature than other methods in the literature;
- The global controller located at the master side for the constrained movement case is designed as a 2-DOF controller with both the delayed environment reaction

force and the difference between the displacement of the master and the delayed displacement of the slave as its inputs in order to meet the transparency requirement of the system.

In order to frame the design concept clearly and succinctly, both the master and the slave are considered to have the same dynamics; i.e., PHANToM haptic device's dynamics in the X-direction (1-DOF), without loss of generality.

With LFT technique and μ-theory as well as available model reduction techniques, the resulting controllers have well met the robust stability and performance requirements that are envisaged in this problem.

1.4 Contents of the Thesis

In this chapter, certain specific terminologies and concepts are explained and a brief introduction to various controller design methods for bilateral telemanipulation systems has been made. Also, some of most critical efforts and contributions of this thesis have been highlighted.

In Chapter 2, the theoretical foundation of the proposed controller design framework based on H_∞, μ-synthesis and analysis methods as well as LFT technique are briefly introduced.

Chapter 3 presents the formulation of the telemanipulation problem. The design strategies and detailed steps for the proposed schemes are then provided.

In Chapter 4, the controllers obtained and designed in Chapter 3 are analyzed in both frequency domain and time domain using μ-analysis method and Matlab Simulink software tools respectively. The effects of the time delay in the communication channel on the system performance are also discussed in details.

Chapter 5 draws some conclusions associated with the research conducted in this thesis. Considerations for future work and directions of research are stated.

Finally, some of Matlab codes compiled for the research and some simulation results presented in this thesis are provided in Appendix A and B, respectively.

Chapter 2

LFT Machinery, μ-Analysis and Synthesis

Usually, classical design approaches take advantage of techniques such as root locus, Bode and Nyquist plots, which enable designers to visualize how the system dynamics are being modified. Unfortunately, these approaches do not generally guarantee the system considered achieve stability and performance in the presence of model uncertainties. In addition, a nominal model with low order is often used to represent the system, which only describes the system behaviour within low-middle frequency ranges. This makes the system behaviour uncertain when it is expected to run at high frequencies.

Robust control theory provides a means for overcoming this problem. H_∞-control theory, μ-analysis and synthesis [9, 10, 30] are widely used in recent years to yield a controller or compensator achieving nominal performance and robust stability simultaneously. Since μ-analysis and synthesis methods take the structured uncertainties into account, the linear fractional transformation (LFT) machinery [9, 10, 29, 31] is brought into the method. In other words, the use of μ-theory depends on its intimate relationship with LFT to a large extent.

Briefly speaking, H_∞-synthesis is a process of designing control laws to minimize the H_∞-norm, a function of the closed-loop frequency response, for multivariable systems; μ-analysis is an analysis test that determines the worst-case effects on system stability

and performance in the presence of uncertainty perturbations while μ-synthesis is an iterative technique, which combines H_∞-synthesis and μ-analysis, and minimizes the structured singular values (SSV) μ, which is equivalent to optimizing the design objective known as robust performance. Achieving robust performance implies that the closed-loop system realizes the performance requirements for the complete set of perturbations to the nominal model. In this way, μ-synthesis directly incorporates both performance and robustness objectives in the design process. In addition, the shape of the closed-loop frequency response is specified by the designer using frequency-dependent weighting functions during the design process [36].

In this thesis, H_∞/μ-theory are used to design a series of controllers for a bilateral haptic telemanipulation system, using the requirements stated in the design specifications. This chapter reviews some of the most important points of H_∞/μ-theory as well as LFT machinery, which are related to the research investigated in this thesis, based on [10, 11, 29, 30, 32, 33, 35, 36].

2.1 Basic Concepts

A linear time-invariant finite-dimensional (LTIFD) system can be described in the state-space as

$$\begin{cases} \dot{x} = Ax + Bu, \quad x(0) = x_0 \\ y = Cx + Du \end{cases} \tag{2.1}$$

with input u, output y, state x, and the coefficient matrices A, B, C, and D. Here u, y, x are signals which are functions of time $t \in [0, \infty)$ that are piece-wise continuous. Conventionally, $u(\cdot)$ is used to denote the signal u as a whole while $u(t)$ denotes the value of u at time instant t.

The transfer matrix $G(s)$ of the system (2.1) is defined as $G(s) = C(sI - A)^{-1}B + D$, and is often represented using symbol

$$\left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right].$$

Suppose that the system is at rest at time 0, i.e., $x_0 = 0$, and so $y(s) = G(s)u(s)$. Therefore, the system can be viewed as a device which processes signals and maps the input signals into the output signals.

A matrix whose elements are real rational functions is stable if it has no poles at infinity (i.e., proper) and in the closed right half plane. The set of real rational proper and stable matrices of dimension $m \times n$ is denoted by $RH_\infty^{m\times n}$ or simply RH_∞ if the dimension is implied from the context. The results of the following operations are still in this set - namely, a scalar multiple of one stable transfer matrix, the sum and the product of stable transfer matrices of compatible dimensions.

System (2.1) is stable if all the eigenvalues of A are in the open left-half plane. The set of eigenvalues of A, $\lambda(A)$, is called the spectrum of A.

The following relationships exist between the stability of system (2.1) and its transfer matrix $G(s)$: If the system (or A) is stable, then $G(s)$ is stable. Conversely, if $G(s)$ is stable, (A, B) is stabilizable, and (A, C) is detectable, then the system (or A) is stable.

For a system represented by (2.1), the vector-valued signal $x(\cdot)$ is bounded if the maximal amplitude or peak

$$\|u\|_\infty = \sup_{t\geq 0}\|u(t)\| \tag{2.2}$$

is finite, where

$$\|u(t)\| = \sqrt{u(t)^T u(t)} \tag{2.3}$$

is called the Euclidean norm of the vector u(t) while $\|u\|_\infty$ is called the L_∞-norm, a norm on the vector space of all bounded signals.

Definition 2.1 (BIBO stability) [32] *A system is said to be bounded-input bounded-output (BIBO) stable if it maps an arbitrary bounded input $u(\cdot)$ into a bounded output $y(\cdot)$.*

This implies that for a BIBO stable system, if $\|u\|_\infty < \infty$, then $\|y\|_\infty < \infty$. This property is equivalent to the stability of the system transfer matrix $G(s) = C(sI - A)^{-1}B + D$.

Theorem 2.1 [32] *System (2.1) maps bounded inputs $u(\cdot)$ into bounded outputs $y(\cdot)$ if and only if the corresponding transfer matrix $G(s) = C(sI - A)^{-1}B + D$ is stable.*

Take $\|u\|_\infty$ and $\|y\|_\infty$ as measures of the sizes of the input and the output of a system, then the system amplifies the input signal by $\|y\|_\infty / \|u\|_\infty$, and

$$\gamma_p = \sup_{0<\|u\|_\infty<\infty} \frac{\|y\|_\infty}{\|u\|_\infty} \tag{2.4}$$

is called its peak-to-peak gain. From Definition 2.1, $\|y\|_\infty \le \gamma_p \|u\|_\infty$ holds for all bounded input signals; i.e., γ_p quantifies how the amplitudes of bounded input signals are amplified or attenuated by the system and is the smallest value such that the inequality is satisfied.

The energy of a signal $x(\cdot)$

$$\|x\|_2 = \sqrt{\int_0^\infty \|x\|^2 dt} \tag{2.5}$$

can also be used to measure its size. A signal with a large energy can have a small peak and vice versa.

A BIBO system maps any signal of finite energy into a signal of finite energy; i.e., if $\|u\|_2 < \infty$, then $\|y\|_2 < \infty$. This property is also equivalent to the stability of the system transfer matrix.

The energy gain of system (2.1) is defined by

$$\gamma_e = \sup_{0<\|u\|_2<\infty} \frac{\|y\|_2}{\|u\|_2} \tag{2.6}$$

which is equal to the maximal singular value of the system transfer matrix over frequency $\omega \in \boldsymbol{R}$, that is

$$\gamma_e = \sup_{\omega \in \mathbf{R}} \bar{\sigma}[G(j\omega)]. \tag{2.7}$$

***Definition 2.2* (H_∞-norm)** [10] *The infinity-norm of the transfer function matrix $G(s)$ is defined as the supremum over the frequency ω, of its maximum singular value evaluated on the $j\omega$-axis:*

$$\|G\|_\infty := \sup_{\omega \in R} \bar{\sigma}(G(j\omega)). \tag{2.8}$$

Equation (2.8) defines a norm on the vector space of all real-rational proper and stable matrices $RH_\infty^{m\times n}$ and is called the H_∞-norm. It can be proved that the energy gain of the stable LTI system (2.1) is just equal to the H_∞ norm of the corresponding transfer matrix: $\gamma_e = \|G\|_\infty$.

The supremum is the same as maximum for continuous smooth functions. Recall that the maximum singular value is the same as maximum multivariable gain and the transfer function matrix is the representation of a multivariable system. Therefore, the above definition can be rewritten in control terms as follow.

Definition 2.3 [36] *The infinity-norm of a (multivariable) system is the maximum over the frequency of the maximum multivariable gain from the input vector to the output vector.*

Note that the maximum singular value $\bar{\sigma}$ of a system is a function of frequency and H_∞-norm of a system is a scalar. It is simply the maximum possible gain of the system over all frequencies.

According to the definition, the H_∞-norm for transfer function matrices is closely related to the L_2-norm for matrices and vectors instead of the infinity-norm for matrices and vectors; that is,

$$\|G\|_\infty = \sup_{\omega \neq 0} \frac{\|y\|_2}{\|u\|_2}. \tag{2.9}$$

2.2 Linear Fractional Transformation (LFT) Machinery

LFT is a very useful and efficient machinery to represent uncertainty in matrices and systems. It provides a general framework for analysis and synthesis of system stability and performance in the presence of uncertainty. The fundamental points of LFT for modeling an uncertain system are to separate what is known from what is unknown in a connection, which is similar to feedback, and to bound the unknown elements with possible values.

LFT can be viewed as a direct generalization of the notion of a state-space realization, where a linear dynamical system is written as a feedback interconnection of a constant matrix and a very simple dynamic element made up of a diagonal matrix of deltas, which represent normalized uncertainties in the system, and/or integrators. This realization greatly facilitates manipulation and computation of linear systems.

Two properties of LFT make it very important in the analysis of robust stability and performance using structured singular values. One is that the interconnections of LFTs are still LFTs, which makes it possible to take several uncertainties, represented by Δ-

blocks, at various places in an interconnection of several systems and bring these back in the basic LFT structure. The other is that using LFT, unstructured uncertainties at component level can become structured uncertainties at system level.

The advantage of the LFT formulation is that it gives a common base for uncertainty modeling, stability and performance analysis of perturbed system (called the analysis problem) and for controller synthesis (the synthesis problem).

2.2.1 Definition of LFT

Definition 2.4 [10] *Assume that both M and* $\widetilde{M}$ *are complex matrices partitioned as*

$$M = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix} \in \mathbf{C}^{(p_1+p_2)\times(q_1+q_2)} \tag{2.10}$$

and

$$\widetilde{M} = \begin{bmatrix} \widetilde{M}_{11} & \widetilde{M}_{12} \\ \widetilde{M}_{21} & \widetilde{M}_{22} \end{bmatrix} \in \mathbf{C}^{(p_2+p_1)\times(q_2+q_1)} \tag{2.11}$$

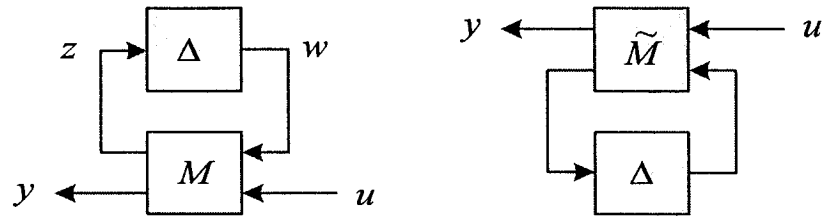
and $\Delta \in \mathbf{C}^{q_1\times p_1}$ *is a complex matrix. Then an upper LFT with respect to* Δ*, as shown in Figure 2.1 (a), is defined as*

$$F_u(M,\Delta) := M_{22} + M_{21}\Delta(I - M_{11}\Delta)^{-1}M_{12} \tag{2.12}$$

provided that the inverse $(I - M_{11}\Delta)^{-1}$ *exists. Similarly, a lower LFT with respect to* Δ*, as shown in Figure 2.1 (b), is defined as*

$$F_l(\widetilde{M},\Delta) := \widetilde{M}_{11} + \widetilde{M}_{12}\Delta(I - \widetilde{M}_{22}\Delta)^{-1}\widetilde{M}_{21} \tag{2.13}$$

provided that the inverse $(I - \widetilde{M}_{22}\Delta)^{-1}$ *exists.*



(a) (b)

Fig. 2.1 LFT Description: (a) Upper LFT and (b) Lower LFT

According to Definition 2.4, if the uncertainties can be separated from a plant with input u and output y, as shown in Figure 2.1, the transformation from u to y will lead to

$$F_u(M,\Delta) = F_l(\tilde{M},\Delta) \tag{2.14}$$

with

$$\tilde{M} = \begin{bmatrix} M_{22} & M_{21} \\ M_{12} & M_{11} \end{bmatrix}. \tag{2.15}$$

The transfer function matrix M_{22} in the upper LFT represents the nominal plant and Δ the perturbations of this plant by uncertainties. The transfer function matrices M_{12}, M_{21} and M_{11} represent the knowledge on how the perturbation Δ affects the nominal plant M_{22}. In this way, a LFT can be used to model uncertainties and variations of nominal conditions of a system and to analyze to what degree of variations (Δ) the system is robust against. On the other hand, the lower LFT may be used as a model of the closed-loop system. With M as the system to be controlled while Δ is replaced by the controller K, the closed-loop transfer function can be synthesized using Equation (2.13).

Usually, mathematical models of uncertain systems are not obtained in linear fractional form. Thus, for robustness analysis and design, it is essential to develop a systematic

procedure to determine linear fractional representations of systems with uncertainty. In addition, it is desirable to obtain the linear fractional representation of minimal order among all possible linear fractional representations of the system in order to get a controller with lower order.

2.2.2 LFT Representation of Uncertain Linear Systems

Suppose that an uncertain linear time invariant (LTI) system of the form

$$\begin{cases} \dot{x} = A\,x + B_1\,w + B_2\,u, \\ z = C_1 x + D_{11} w + D_{12} u, \\ y = C_2 x + D_{21} w + D_{22} u, \\ w = \Delta z \end{cases} \tag{2.16}$$

which is shown in Figure 2.2 with the definition

$$x = \frac{I_n}{s}\dot{x}, \tag{2.17}$$

where I_n is an identity matrix of dimension n, as an artificial feedback. Closing the I_n/s loop, Figure 2.2 becomes the standard Figure 2.1(a) with $M = \begin{bmatrix} M_{11}(s) & M_{12}(s) \\ M_{21}(s) & M_{22}(s) \end{bmatrix}$ where

$$\begin{cases} M_{11}(s) = D_{11} + C_1 (sI - A)^{-1} B_1, \\ M_{12}(s) = D_{12} + C_1 (sI - A)^{-1} B_2, \\ M_{21}(s) = D_{21} + C_2 (sI - A)^{-1} B_1, \\ M_{22}(s) = D_{22} + C_2 (sI - A)^{-1} B_2. \end{cases} \tag{2.18}$$

Thus, the transfer matrix of this uncertain system from u to y can be specified by equations (2.12) and (2.18).

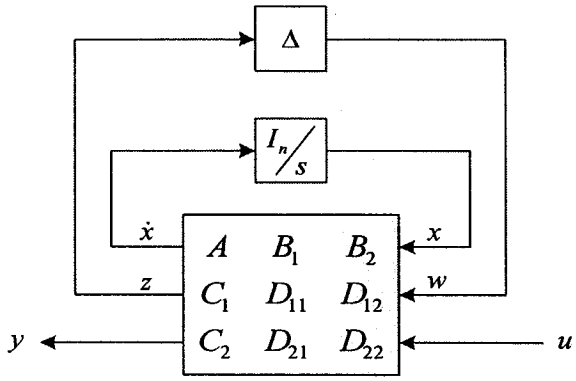


Fig. 2.2 LFT Representation of Standard Uncertain Linear System

On the other hand, by closing the Δ loop in Figure 2.2, a parameter-dependent state-space representation can be achieved as shown in Figure 2.3 with

$$\begin{cases} A(\Delta) = A + B_1\Delta(I - D_{11}\Delta)^{-1}C_1, \\ B(\Delta) = B_2 + B_1\Delta(I - D_{11}\Delta)^{-1}D_{12}, \\ C(\Delta) = C_2 + D_{21}\Delta(I - D_{11}\Delta)^{-1}C_1, \\ D(\Delta) = D_{22} + D_{21}\Delta(I - D_{11}\Delta)^{-1}D_{12}. \end{cases} \tag{2.19}$$

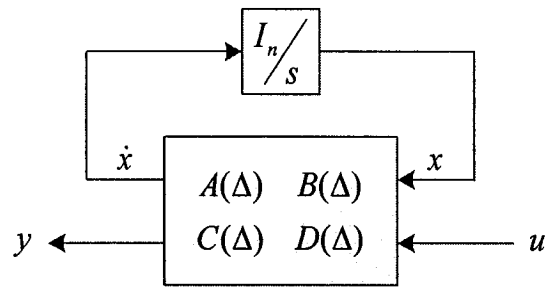


Fig. 2.3 Parameter-dependent state-space representation

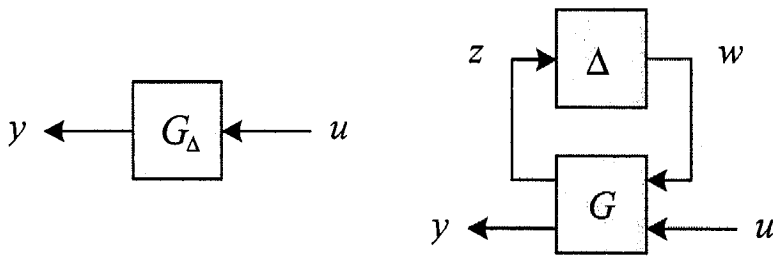
2.2.3 Separating Uncertainties from Plant/System

A critical point in robust stability analysis is to calculate the system transfer matrix seen from Δ in an open-loop interconnection. Usually an interconnection is built from some plants or subsystems, which might be perturbed by some uncertain factors. Assume that

they are parameterized as G_Δ with $\Delta \in \mathbf{\Delta}$ as shown in Figure 2.4(a). To separate those uncertainties from the system in the form shown in Figure 2.4(b), let $G = \begin{bmatrix} G_{11} & G_{12} \\ G_{21} & G_{22} \end{bmatrix}$.

Since $\begin{bmatrix} z \\ y \end{bmatrix} = G \begin{bmatrix} w \\ u \end{bmatrix} = \begin{bmatrix} G_{11} & G_{12} \\ G_{21} & G_{22} \end{bmatrix} \begin{bmatrix} w \\ u \end{bmatrix}$ and $w = \Delta z$ in Figure 2.4(b), according to Definition 2.4, we have

$$y = [G_{22} + G_{21}\Delta(I - G_{11}\Delta)^{-1}G_{12}]u . \tag{2.20}$$



(a) An uncertain plant **(b) Pulling out the uncertainty**

Fig. 2.4 Separating the uncertainty from an uncertain Plant

A procedure for modeling all the uncertainties in the system is as follow:

- Create a new input vector w and a new output vector z, both with as many elements as there are uncertainties in the system. To create a new system that can be perturbed through an upper LFT, add these new input and output vectors to the system before the original input and output;
- Using the extra inputs to perform variation in the state equations and choosing the perturbation matrix Δ equal to the identity matrix, the upper loop equation gives the relation between the new inputs and outputs;
- Find out the relationship between the new input vector w and the system states x in order to satisfy the rewritten, perturbed state equations, and

- Define the extra outputs as such and construct the new system.

For additive uncertainty,

$$y = (G_0 + \Delta)u = [G_0 + I \cdot \Delta(I - \mathbf{0} \cdot \Delta)^{-1} \cdot I]\ u, \tag{2.21}$$

where G_0 is the nominal plant and I is an identity matrix. Comparing Equation (2.21) with Equation (2.20), we obtain that

$$\begin{bmatrix} z \\ y \end{bmatrix} = \begin{bmatrix} \mathbf{0} & I \\ I & G_0 \end{bmatrix} \begin{bmatrix} w \\ u \end{bmatrix}, \quad w = \Delta z. \tag{2.22}$$

Similarly, for input multiplicative uncertainty,

$$y = G_0(I + \Delta)u = [G_0 + G_0 \cdot \Delta(I - \mathbf{0} \cdot \Delta)^{-1} \cdot I]u, \tag{2.23}$$

$$\begin{bmatrix} z \\ y \end{bmatrix} = \begin{bmatrix} \mathbf{0} & I \\ G_0 & G_0 \end{bmatrix} \begin{bmatrix} w \\ u \end{bmatrix}, \quad w = \Delta z, \tag{2.24}$$

and for output multiplicative uncertainty,

$$y = (I + \Delta)G_0 u = [G_0 + I \cdot \Delta(I - \mathbf{0} \cdot \Delta)^{-1} \cdot G_0]u, \tag{2.25}$$

$$\begin{bmatrix} z \\ y \end{bmatrix} = \begin{bmatrix} \mathbf{0} & G_0 \\ I & G_0 \end{bmatrix} \begin{bmatrix} w \\ u \end{bmatrix}, \quad w = \Delta z. \tag{2.26}$$

All three can be used to model uncertainty of the plant G. The additive uncertainty model provides a more structured, direct way of modeling plant uncertainty while the two multiplicative uncertainty models are often used to model general, unstructured plant uncertainty. Since the latter can be made frequency-dependant, they are also used for unmodeled plant dynamics that occur in a specified frequency range. In addition,

depending on their position in the loop, another application of them is to model actuator and sensor dynamics uncertainty respectively.

Uncertainties might be structured or not at the component level. By separating them from an interconnection, the resulting uncertainty for the interconnection is automatically block-diagonal, even if the component uncertainties are not originally structured. And the uncertainties of the components appear on the diagonal.

2.2.4 LFT Formulation for Parametric Uncertainties

Here, the basic principles for LFT formulation for parametric uncertainties are explained with a general plant. Consider an actual plant

$$G(s) = \frac{y(s)}{u(s)} = \frac{c_4 s^4 + c_3 s^3 + c_2 s^2 + c_1 s + c_0}{s^4 + d_3 s^3 + d_2 s^2 + d_1 s + d_0}$$

where $c_i = \bar{c}_i(1 + \delta_{c_i}\Delta_{c_i})$, $d_j = \bar{d}_j(1 + \delta_{d_j}\Delta_{d_j})$, $i = 0,\ 1,\ \cdots,\ 4,\ j = 0,\ 1,\ 2,\ 3$; Δ_{c_i} and Δ_{d_j} are normalized uncertainties; i.e., they are unknown but lie in the interval [-1, 1]; δ_{c_i} and δ_{d_j} denote the relative deviations of actual coefficients c_i and d_j from their nominal values $\bar{c}_i$ and $\bar{d}_j$, respectively.

To represent the plant as an LFT in terms of uncertainty parameters Δ_{c_i} and Δ_{d_j}, all the uncertain sources must be separated from the generalized plant. The first step is to draw a block diagram for the input/output relation with each uncertain source separated as shown in Figure 2.5.

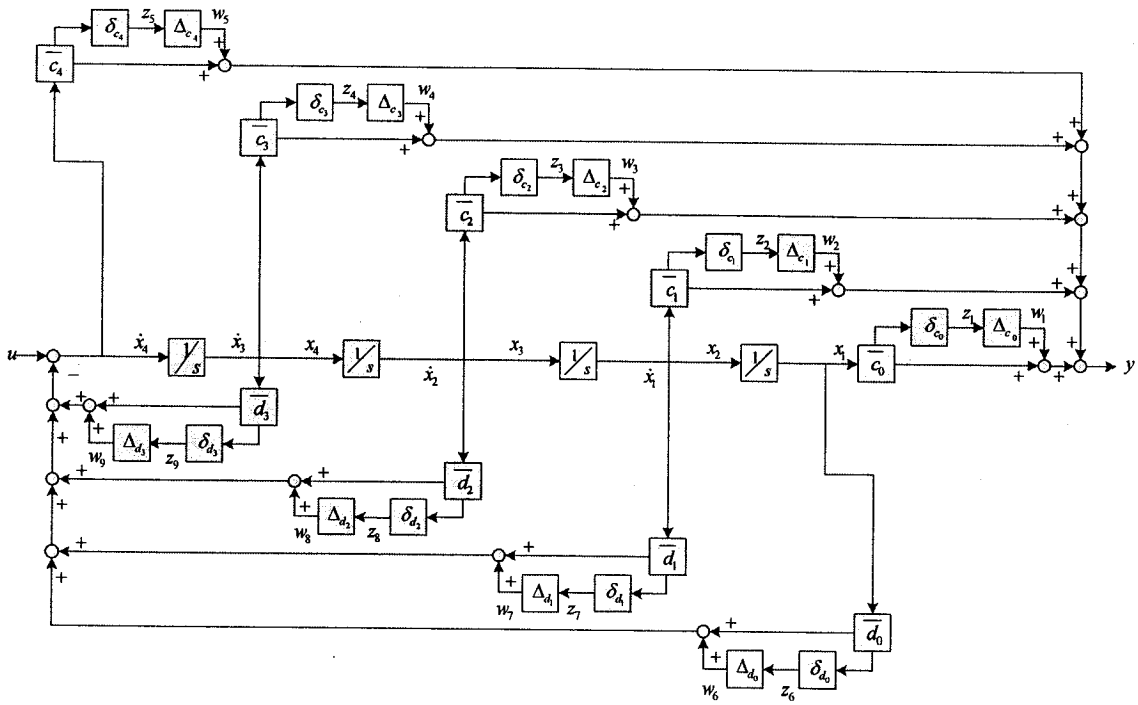


Fig. 2.5 Block diagram of a system with uncertain parameters

Then denote the inputs and outputs of those isolated uncertainties with z's and w's respectively.

Finally write out the relations between z's and w's in the case that the connections to all uncertain sources are broken; i.e., pulling them out from Figure 2.5. In this way, we obtain that

$$\begin{aligned}&[z_1 \quad z_2 \quad z_3 \quad z_4 \quad z_5 \quad z_6 \quad z_7 \quad z_8 \quad z_9 \quad \dot{x}_1 \quad \dot{x}_2 \quad \dot{x}_3 \quad \dot{x}_4 \quad y]^T \\ &= M[w_1 \quad w_2 \quad w_3 \quad w_4 \quad w_5 \quad w_6 \quad w_7 \quad w_8 \quad w_9 \quad x_1 \quad x_2 \quad x_3 \quad x_4 \quad u]^T,\end{aligned}$$

where

$$M=\left[\begin{array}{ccccc|cccc}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \bar{c}_0\delta_{c_0} & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \bar{c}_1\delta_{c_1} & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \bar{c}_2\delta_{c_2} & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \bar{c}_3\delta_{c_3} & 0 \\
0 & 0 & 0 & 0 & 0 & -\bar{c}_4\delta_{c_4} & -\bar{c}_4\delta_{c_4} & -\bar{c}_4\delta_{c_4} & -\bar{c}_4\delta_{c_4} & 0 & 0 & 0 & 0 & \bar{c}\delta_{c_4} \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \bar{d}_0\delta_{d_0} & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \bar{d}_1\delta_{d_1} & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \bar{d}_2\delta_{d_2} & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \bar{d}_3\delta_{d_3} & 0 \\
\hline
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & -1 & -1 & -1 & -1 & -\bar{d}_0 & -\bar{d}_1 & -\bar{d}_2 & -\bar{d}_3 & 1 \\
1 & 1 & 1 & 1 & 1 & -\bar{c}_4 & -\bar{c}_4 & -\bar{c}_4 & -\bar{c}_4 & \bar{c}_0 & \bar{c}_1 & \bar{c}_2 & \bar{c}_3 & \bar{c}_4
\end{array}\right],$$

and

$$\begin{aligned}
&[w_1 \quad w_2 \quad w_3 \quad w_4 \quad w_5 \quad w_6 \quad w_7 \quad w_8 \quad w_9]^T \\
&= \Delta[z_1 \quad z_2 \quad z_3 \quad z_4 \quad z_5 \quad z_6 \quad z_7 \quad z_8 \quad z_9]^T,
\end{aligned}$$

where $\Delta = diag\{\Delta_{c_0}, \Delta_{c_1}, \Delta_{c_2}, \Delta_{c_3}, \Delta_{c_4}, \Delta_{d_0}, \Delta_{d_1}, \Delta_{d_2}, \Delta_{d_3}\}$. Therefore,

$$[\dot{x}_1 \quad \dot{x}_2 \quad \dot{x}_3 \quad \dot{x}_4 \quad y]^T = F_u(M,\Delta)[x_1 \quad x_2 \quad x_3 \quad x_4 \quad u]^T.$$

2.3 The Small Gain Theorem

The small gain theorem is the foundation for analyzing robust stability of nominally stable systems under unstructured perturbation.

Theorem 2.2 (The small gain theorem) [10] *If* $G \in RH_\infty$ *and* $\gamma > 0$, *then the system shown in Figure 2.6 is well-posed and internally stable for all* $\Delta(s) \in RH_\infty$ *with*

1) $\|\Delta\|_\infty \le 1/\gamma$ *if and only if* $\|M(s)\|_\infty < \gamma$;

2) $\|\Delta\|_\infty < 1/\gamma$ *if and only if* $\|M(s)\|_\infty \le \gamma$.

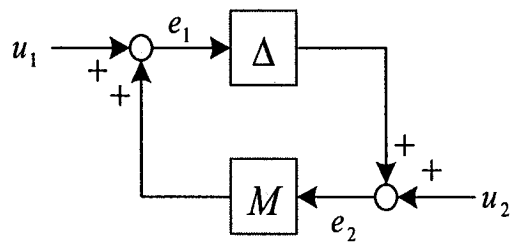


Fig. 2.6 The closed-loop for stability analysis

According to the above small gain theorem, the condition for which the closed-loop system is stable is:

$$\|\Delta\|_\infty \|M\|_\infty < 1$$

$$\Leftrightarrow \quad \|\Delta\|_\infty < \frac{1}{\|M\|_\infty}$$

$$\Rightarrow \quad \bar{\sigma}(\Delta) < \frac{1}{\bar{\sigma}(M)} \quad , \quad \forall s = j\omega.$$

This implies that a plot of inverse of the maximum singular value of M on the frequency axis reflects the robust stability level of the closed-loop system; an arbitrary Δ with maximum singular value below this plot will not endanger the stability of the closed-loop system. The higher are the gains in the $1/\bar{\sigma}(M)$, the greater the robustness level of the closed-loop system. Lower gains mark the frequencies of perturbations to which the closed-loop stability is more sensitive.

2.4 The Structured Singular Value (SSV) μ

H_∞-theory has been widely applied in controller designs. However, there are two main limitations to it [34]: 1) the robustness singular value analysis based on the small gain theorem can only be made on systems with unstructured uncertainties; i.e., only full complex perturbations $\Delta(s) \in \mathbf{C}^{m\times n}$ can be treated in an H_∞ robust stability test; 2) robust performance can only be handled in a conservative way even for full complex perturbations since stability and performance can not be separated in the H_∞ structure.

Robustness analysis on systems with multiple uncertainties incorporated in a structured perturbation matrix Δ requires the use of the structured singular value μ-theory.

μ-analysis and synthesis theories deal with the problem of achieving a desired performance in the face of various uncertainties; i.e., the robust performance problem. The desired performance is automatically achieved for SISO systems if the system has guaranteed robust stability and nominal performance. However, this does not hold in the MIMO case. Therefore, μ-analysis and synthesis theories become a tool to address the multivariable robust performance problem.

μ-analysis and synthesis theories naturally arise from stability of general class of systems of LFT form. The existence of LFTs automatically leads to the formulation of the robust performance problem.

Definition 2. 5 [10, 32] *A control system with uncertainties is said to achieve:*

1) nominal stability (NS) if the closed-loop system with no uncertainty is stable;

2) *robust stability (RS) if the closed-loop system with all defined uncertainties is stable;*
3) *nominal performance (NP) if the closed-loop system with no uncertainty satisfies the performance requirements;*
4) *robust performance (RP) if the closed-loop system with all defined uncertainties satisfies the performance requirements.*

Furthermore, robust stability implies that the system is stable in the worst case and robust performance implies that the system satisfies performance requirements in the worst case.

Since model uncertainty can be easily formulated using H_∞-norm, this norm is selected for performance evaluation. Any model uncertainty Δ can be normalized with a weight such that its H_∞-norm is less than 1, while

$$\|\Delta\|_\infty \le 1 \Leftrightarrow \sigma(\Delta(j\omega)) \le 1, \quad \forall\omega . \tag{2.27}$$

If all uncertainties are lumped into one perturbation matrix, an unstructured uncertainty, the conservativeness cannot be avoided since this method includes a number of plant cases that may not occur in practice [8, 35]. Therefore, the structured uncertainty concept is used to model uncertainty more tightly, that is, we use several perturbation blocks, each of which is related to a specific physical source of uncertainty; for example, a parametrical uncertainty.

Definition 2.6 (The Structured Singular Value) [9-10] *For* $M \in \boldsymbol{C}^{n\times n}$ *and* $\boldsymbol{\Delta} \in \boldsymbol{C}^{n\times n}$ *which is defined as*

$$\Delta = \left\{ diag[\delta_1 I_{r_1}, \cdots, \delta_S I_{r_S}, \Delta_1, \cdots, \Delta_F] : \delta_i(s) \in \mathbf{C}, \Delta_j \in \mathbf{C}^{m_j \times m_j} \right\} \tag{2.28}$$

where $\sum_{i=1}^{S} r_i + \sum_{j=1}^{F} m_j = n$, *S and F represent the number of repeated scalar blocks and the number of full blocks respectively, the structured singular value* $\mu_\Delta(M)$ *is a function of the complex matrix M and the structured uncertainty* $\mathbf{\Delta}$. *It is defined as*

$$\mu_\Delta(M) := \frac{1}{\min\{\bar{\sigma}(\Delta) : \Delta \in \mathbf{\Delta}, \det(I - M\Delta) = 0\}} \tag{2.29}$$

unless no $\Delta \in \mathbf{\Delta}$ *makes* $\det(I - M\Delta) = 0$, *in which case* $\mu_\Delta(M) := 0$. *In other words,* $\mu_\Delta(M)$ *is the reciprocal of the size of the smallest set* Δ *from the perturbation set* $\mathbf{\Delta}$ *that makes the matrix* $(I - M\Delta)$ *singular.*

Note that $\det(I - M\Delta)$ is the multivariable characteristic polynomial; therefore, $\mu_\Delta(M)$ reveals the size of the smallest Δ from $\mathbf{\Delta}$ that makes the closed-loop in Figure 2.6 unstable. That is to say, μ can be used as a tight measurement of robust stability with respect to a structured Δ with multiple perturbation blocks.

According to Definition 2.6, for a multivariable system, μ is a function of frequency and thus can be evaluated and calculated with $s = j\omega$ in the transfer function matrix at any frequency. μ is also a function of the complex matrix M and the structured perturbation collection Δ. The smaller the values of μ are, the better the robustness and performance. Specifically, if Δ has only one repeated scalar block and no full complex block ($S = 1$, $F = 0$),

$$\mu_\Delta(M) = \rho(M) = \max_i |\lambda_i(M)|, \quad \forall\ \Delta = \{\delta I : \delta \in \mathbf{C}\} \tag{2.30}$$

and if Δ has only one full complex block and no repeated scalar block ($S = 0$, $F = 1$),

$$\mu_{\Delta}(M) = \overline{\sigma}(M), \quad \forall\ \Delta \in \mathbf{C}^{n\times n}. \tag{2.31}$$

For a general $\mathbf{\Delta}$ defined in Equation (2.28),

$$\rho(M) \le \mu_{\Delta}(M) \le \overline{\sigma}(M). \tag{2.32}$$

These bounds can be refined with two subsets of $\mathbf{C}^{n\times n}$:

$$\mathbf{D} = \left\{ diag[D_1, \cdots, D_S, d_1 I_1, \cdots, d_F I] : d_i \in \mathbf{C},\ D_j \in \mathbf{C}^{r_j \times r_j} \right\} \tag{2.33}$$

and

$$\mathbf{U} = \left\{ U \in \mathbf{\Delta} : U^* U = I_n \right\}. \tag{2.34}$$

Theorem 2. 3 [29] *For all* $D \in \mathbf{D}$ *and* $U \in \mathbf{U}$

$$\mu_{\Delta}(UM) = \mu_{\Delta}(MU) = \mu_{\Delta}(M) = \mu_{\Delta}(DMD^{-1}). \tag{2.35}$$

With the above theorem, the bound in Formula (2.32) can be tightened to

$$\max_{U\in\mathbf{U}} \rho(UM) \le \max_{\Delta\in\mathbf{B\Delta}} \rho(\Delta M) = \mu_{\Delta}(M) \le \inf_{D\in\mathbf{D}} \overline{\sigma}(DMD^{-1}) \tag{2.36}$$

where

$$\mathbf{B\Delta} = \left\{ \Delta \in \mathbf{\Delta} : \overline{\sigma}(\Delta) \le 1 \right\} \tag{2.37}$$

According to Doyle [8], the lower bound of $\mu_{\Delta}(M)$ is nothing but $\max_{U\in U} \rho(UM)$. With Matlab μ-toolbox it is possible to compute the upper and lower μ-bounds of any complex matrix M in the face of any structured Δ.

2.5 μ-Analysis and Synthesis Theories

Generally speaking, controller design for a system in the face of various perturbations consists of three respects: 1) finding a controller K which makes the closed-loop system internally stable for a family of systems; 2) this stabilizing controller K should make the closed-loop system respond well under various external signals, and 3) the control signals should remain within physical and practical limits.

2.5.1 General System Formulation and μ-Analysis

There exist three basic components in any uncertain closed-loop system: 1) the generalized plant P or an open interconnection of the system, which includes the actual plant to be controlled, weighting functions, reference models, etc.; 2) the controller K to be designed for implementing robust performance, and 3) the uncertainty matrix Δ with unit norm (belonging to a pre-specified set). With LFT machinery, such an uncertain closed-loop system can be transformed into a general form as shown in Figure 2.7.

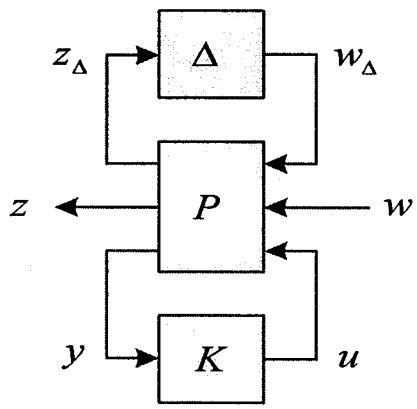


Fig. 2.7 General description of a closed-loop uncertain system

For the purpose of synthesis and analysis, three pairs of input-output variables should be identified for the above system:

- w and z, which are the generalized disturbances and controlled variables and consist of the disturbance, or external input, and error signals respectively;
- u and y, which denote the control inputs and measurement outputs, respectively;
- w_Δ and z_Δ, which represent the signals that are introduced to separate the uncertainties from the nominal system.

With the above-mentioned three pairs of input-output variables, the open interconnection structure of the system, P, as shown in Figure 2.7, can be described as follow:

$$\begin{bmatrix} z_\Delta \\ z \\ y \end{bmatrix} = P \begin{bmatrix} w_\Delta \\ w \\ u \end{bmatrix} = \begin{bmatrix} P_{11} & P_{12} & P_{13} \\ P_{21} & P_{22} & P_{23} \\ P_{31} & P_{32} & P_{33} \end{bmatrix} \begin{bmatrix} w_\Delta \\ w \\ u \end{bmatrix}. \tag{2.38}$$

The command *systic* in Matlab μ-toolbox [11] is quite useful in automating the calculation of a state-space representation of this system. With $u = Ky$, the closed-loop interconnection, shown in Figure 2.8, can be described as

$$\begin{bmatrix} z \\ y \end{bmatrix} = F_u(P,\Delta) \begin{bmatrix} w \\ u \end{bmatrix} = \left(\begin{bmatrix} P_{22} & P_{23} \\ P_{32} & P_{33} \end{bmatrix} + \begin{bmatrix} P_{21} \\ P_{31} \end{bmatrix} \Delta (I - P_{11}\Delta)^{-1} \begin{bmatrix} P_{12} & P_{13} \end{bmatrix} \right) \begin{bmatrix} w \\ u \end{bmatrix} \tag{2.39}$$

and

$$\begin{bmatrix} z_\Delta \\ z \end{bmatrix} = F_l(P,K) \begin{bmatrix} w_\Delta \\ w \end{bmatrix} = \left(\begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix} + \begin{bmatrix} P_{13} \\ P_{23} \end{bmatrix} K (I - P_{33}K)^{-1} \begin{bmatrix} P_{31} & P_{32} \end{bmatrix} \right) \begin{bmatrix} w_\Delta \\ w \end{bmatrix}. \tag{2.40}$$

It is not difficult to find a property of LFT, according to equations (2.39) and (2.40), that

$$F_l[F_u(P,\Delta),K] = F_u[F_l(P,K),\Delta]. \tag{2.41}$$

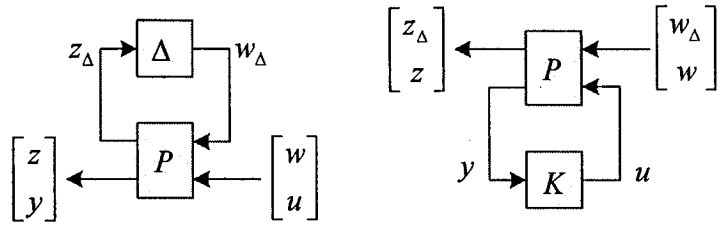


Fig. 2.8 The closed-loop interconnection

Therefore, the following hypothesis is made on the open interconnection P, the uncertain class Δ, and the performance specifications.

Hypothesis 2.1 [32] *Suppose that in the general system representation shown in Figure 2.7*

1) P is a generalized plant;

2) $\Delta(s)$ takes the form of

$$\Delta(s) = \left\{ diag[\delta_1 I_{r_1}, \cdots, \delta_S I_{r_S}, \Delta_1, \cdots, \Delta_F] : \delta_i(s) \in RH_\infty, \Delta_j(s) \in RH_\infty \right\} \quad (2.42)$$

with $\|\delta_i\|_\infty \le 1$ and $\|\Delta_j\|_\infty \le 1$, where $i = 1, \cdots, S$ and $j = 1, \cdots, F$;

3) $I - P_{11}(\infty)\Delta$ is non-singular for all $\Delta \in \mathbf{\Delta}$;

4) The performance of the system is as desired if $F_l[F_u(P,\Delta),K] \le 1$ or $F_u[F_l(P,K),\Delta] \le 1$.

For convenience, let

$$M = F_l(P,K) = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix} \quad (2.43)$$

be the closed-loop system as shown in Figure 2.9, in which M_{11} is the transfer matrix seen from Δ. With the assumption that all the performance weights and uncertainty

weights are incorporated in P, $\|M_{11}\|_\infty < 1$ implies stability, and the robust performance requirement is given by

$$\|F_u(M,\Delta)\|_\infty = \|M_{22} + M_{21}\Delta(I - M_{11}\Delta)^{-1}M_{12}\|_\infty < 1. \tag{2.44}$$

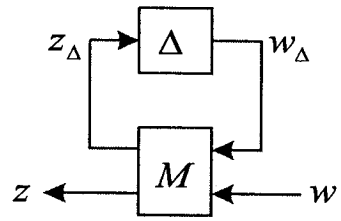


Fig. 2.9 M-Δ interconnection

Condition (2.44) cannot be examined directly since Δ is not known exactly. However, this problem can be solved by treating performance as an artificial uncertainty Δ_P, as shown in Figure 2.10, where Δ_U denotes Δ in Figure 2.9, the existing norm-bounded uncertainty. In this way, $\mu_1(M_{11})$ with respect to Δ_U, $\mu_2(M_{22})$ with respect to Δ_P, and $\mu_{\tilde{\Delta}}(M)$ with respect to $\tilde{\Delta} = diag\{\Delta_U, \Delta_P\}$ all make sense.

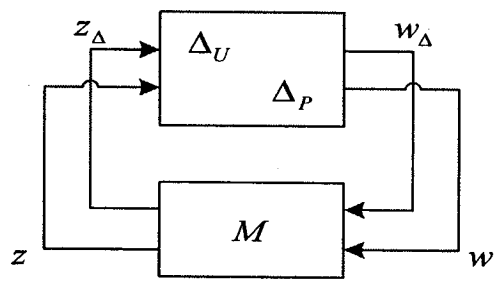


Fig. 2.10 μ-Analysis with augmented uncertainty block $\tilde{\Delta} = diag\{\Delta_U, \Delta_P\}$

Theorem 2.4 (The Main Loop Theorem) [9] *The following are equivalent:*

$$\mu_{\tilde{\Delta}}(M) < 1 \quad \Leftrightarrow \quad \begin{cases} \mu_1(M_{11}) < 1, \ \ and \\ \max\limits_{\Delta_P \in B\Delta_P} \mu_2(F_u(M, \Delta_P)) < 1. \end{cases} \tag{2.45}$$

As an extension of Theorem 2.4, the following theorem provides the necessary and sufficient conditions on system robust stability and performance.

Theorem 2.5 [43] *The generalized system shown in Figure 2.10 achieves*

1) *robust stability if and only if* $\bar{\sigma}(M_{11}(j\omega)) < 1, \ \forall \omega$;
2) *nominal performance if and only if* $\bar{\sigma}(M_{22}(j\omega)) < 1, \ \forall \omega$;
3) *robust performance if and only if* $\mu(M(j\omega)) < 1, \ \forall \omega$.

Above two theorems show robust performance can be examined with μ-analysis by augmenting the uncertainty block and the performance block into one perturbation matrix so that it is exactly equivalent to robust stability in the face of uncertainty $\tilde{\Delta} = diag\{\Delta_U, \Delta_P\}$. This μ-analysis on robust performance is not conservative if the necessary and sufficient conditions are met.

The main loop theorem tells us that if robust performance is achieved both robust stability and nominal performance are also achieved. In addition, robust stability implies nominal stability. Therefore, robust performance test can be used to replace other tests.

In practice, μ-plots about robust performance, robust stability and nominal performance are drawn to analyze the system robustness and performance. The robust performance μ-plot should be above both the robust stability and the nominal performance μ-plots by definition. Furthermore, with unit normalization of real existing uncertainties and gain of

performance channel, the conditions in Theorems 2.4 and 2.5 are satisfied if the corresponding μ-plots lie below unity. If the robust performance μ-plot exceeds unity at some frequency, it implies that the performance is worse than that specified at that frequency and this will happen for plant perturbations smaller than those specified at that frequency.

2.5.2 μ-Synthesis

Assume that the general system description is the same as in Section 2.5.1. The task is to design a controller K which stabilizes the closed-loop $F_u(P,\Delta)$ in Figure 2.8 and leads to $\| F_l[F_u(P,\Delta),K] \| \leq 1$ over all frequencies for all $\Delta \in \mathbf{\Delta}$.

By augmenting Δ to $\tilde{\Delta} = diag\{\Delta_U, \Delta_P\}$, the closed-loop system achieves robust performance if K stabilizes the nominal system P and makes the following equation

$$\mu_{\tilde{\Delta}}(M(j\omega)) = \mu_{\tilde{\Delta}}(F_l(P,K)(j\omega)) \leq 1 \quad \text{for all} \quad \omega \in R \tag{2.46}$$

hold. There is no direct way to find such a K. However, according to Theorem 2.3,

$$\mu_{\tilde{\Delta}}(F_l(P,K)(j\omega)) \leq \inf_{D \in \mathbf{D}} \left\| D^{-1} F_l(P,K)(j\omega) D \right\|.$$

That is to say, any stabilizing controller guarantees equation (2.46) as long as

$$\inf_{D \in \mathbf{D}} \left\| D^{-1} F_l(P,K)(j\omega) D \right\| \leq 1 \quad \text{for all} \quad \omega \in R. \tag{2.47}$$

Hence the μ-synthesis problem reduces to finding a controller which minimizes the upper bounds of the SSV obtained with scaling matrices set **D**.

Since the maximum over frequency of the maximum singular value is the same as the H_∞-norm, the precise description of the controller μ-synthesis problem is to solve

$$\min_{\substack{K \\ \text{stabilizing}}} \min_{D \in \mathbf{D}} \| D^{-1} F_l(P,K)(j\omega) D \|_\infty \tag{2.48}$$

over all controllers K that stabilizes P and over all frequency-dependent scalings $D(\omega) \in \mathbf{D}$. If the value of expression (2.48) is smaller than unity, the problem is solved and equations (2.47) and (2.46) are guaranteed. If the value of expression (2.48) is larger than unity, it might still be possible to let the SSV below unity since only its upper bound is considered here.

The problem (2.48) cannot be solved over K and $D(\omega)$ simultaneously. An indirect but reasonable method is to realize (2.48) by the *D-K* iteration, which iteratively optimizes over the stabilizing controller K while holding $D(\omega)$ fixed and then optimizes over all scaling functions $D(\omega)$ while holding K fixed. Matlab μ-Analysis and Synthesis toolbox provides a very convenient and efficient *D-K* iteration user interface tool, *dkitgui*, to fulfill this task.

The controller K resulting from the μ-synthesis through *D-K* iteration possesses the following properties:

- stabilizes the perturbed closed-loop system $F_u(F_l(P,K),\Delta)$;
- minimizes the perturbed closed-loop gain $\| F_u(F_l(P,K),\Delta) \|_\infty$; and
- has the number of states equal to the sum of the states of generalized plant P and the states of the final *D*-scale.

Fortunately, the states of the resulting controller K can always be reduced by model reduction techniques. Matlab Robust Control Toolbox and μ-Analysis and Synthesis Toolbox provide the designers with some routines, such as *schmr*, *balmr*, *ohklmr*, etc., to have this done. The robustness and performance of the resulting reduced-order controller can then be analyzed with μ-analysis method.

The *D-K* iteration procedure incorporates robust performance objectives directly into the process of controller design and uses the same framework as μ-analysis. One of the benefits of its automation is that it makes designers focus on the construction of the interconnection structure and the selection of weighting functions which reflect the design objectives.

Chapter 3

Controller Synthesis for Bilateral Haptic Telemanipulation Systems

This chapter presents a systematic controller design framework, based on LFT machinery and μ-analysis and synthesis theories, for a haptic telemanipulation system with multiple perturbed components or uncertainties.

3.1 Problem Formulation

In advance of designing or setting up a control system, all of its components and their dynamic behaviours should be well understood. Generally in a bilateral haptic telemanipulation system, the slave manipulator interacts with the environment under the control action of an operator via a haptic device (the master manipulator). If the interaction occurs, the contact force between the slave and the environment is feedback to the master side. Figure 3.1 shows a typical haptic telemanipulation system which consists of a haptic device acting as a local master manipulator, a remote slave manipulator, a communication channel existing between the master and the slave, a human operator, and an environment interacted by the slave, excluding the controllers.

In Figure 3.1, a human operator applies a force f_h on the haptic device, the master manipulator, to drive it to the desired position x_m. The operator is assumed to be able to drive the master to any desired position. The output of the master, i.e., the position x_m is sent to the remote site as a command input of the slave via the communication channel. This command is used to generate a force to drive the slave to a position. If there is no interaction between the slave and the environment, there will be no force reflection to the master site. If the interaction happens, there exists a reaction force f_e whose measured value, together with the displacement of the slave, will be sent back to the master site as a feedback force exerted on the haptic device to provide the operator the information about the remote site.

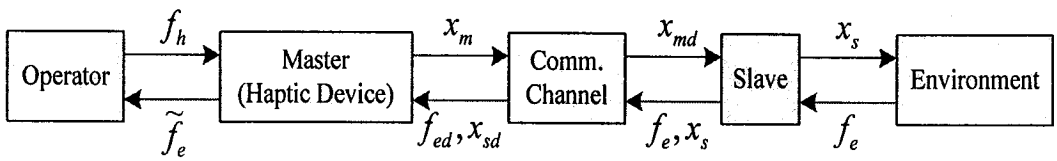


Fig. 3.1 General Structure of a Haptic Telemanipulation System

For the following reasons, the local master haptic manipulator and the remote slave manipulator should have their own independent closed-loop controllers:

- the local master manipulator must be stable in any situation to ensure the safety of the human operator;
- the remote slave manipulator must be stable in any situation to ensure the safety of the object to be manipulated; that is to say, the local controller in the remote site should be able to keep the slave stable even if the communication between the slave and the master is cut off.

In addition, in the situation when the behaviour of the slave manipulator is constrained by an environment which it interacts with, a global closed-loop force/position controller should be designed to compensate the impedance effects in the whole telemanipulation system and the time-delay effects brought by the communication channel.

3.1.1 System Configuration

Figure 3.2 depicts the block diagram of a haptic telemanipulation control system which is adopted in this thesis, where the slave manipulator is position controlled. G_m and G_s are the models of the perturbed or uncertain master and slave manipulators respectively; G_e is the perturbed or uncertain environment plant; $K_m = [K_{m1} \quad K_{m2}]$ and $K_s = [K_{s1} \quad K_{s2}]$ are controllers with two degrees of freedom (2-DOF) to be designed for the master and slave manipulators, respectively, in the case of free motion; G_{T_1} represents the time-delay dynamics of the communication channel from master side to slave side, while G_{T_2} represents the time-delay dynamics of the communication channel from slave side to master side; $K_g = [K_{g1} \quad K_{g2}]$ is the controller to be designed for the global system under the constrained situation; x_m and x_s are the positions of the end-effectors of the master and the slave manipulators, respectively; and f_h and f_e are the forces exerted by the human operator and the environment, respectively.

In the case of the free movement of the telemanipulators, the following control laws are applied onto the master manipulator and the slave:

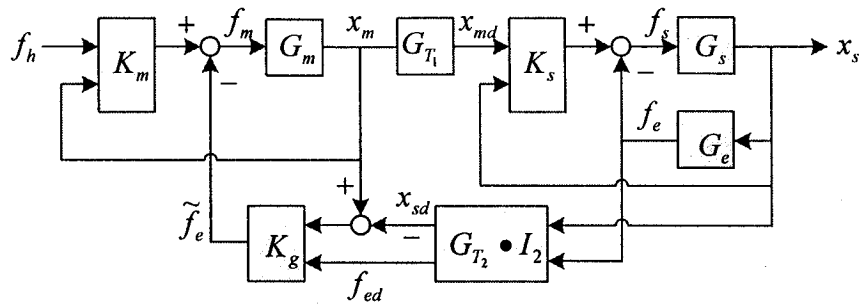


Fig. 3.2 Control Structure of the Haptic Telemanipulation System

$$f_m = K_{m1} f_h + K_{m2} x_m \tag{3.1}$$

and

$$f_s = K_{s1} x_{md} + K_{s2} x_s, \tag{3.2}$$

where $x_{md} = x_m e^{-T_1 s}$ and $T_1 \geq 0$ denotes the transmission delay from the master to the slave. As for the global system in the case of the constrained movement, the feedback control law is selected as

$$\tilde{f}_e = K_{g1}(x_m - x_{sd}) + K_{g2} f_{ed}, \tag{3.3}$$

where $x_{sd} = x_s e^{-T_2 s}$, $f_{ed} = f_e e^{-T_2 s}$, and $T_2 \geq 0$ denotes the transmission delay from the slave to the master.

3.1.2 Design Objectives

The desired task is to design robust controllers, K_m, K_s, and K_g in Figure 3.2, to achieve the following objectives:

- the system should be internally stable to guarantee that all signals in the system are bounded provided that the injected (at any location) signals are bounded;

- the global system should be stable in the presence of all uncertain factors and time-delays of the communication channel;
- a desired master compliance should be realized; that is to say, the displacement of the haptic device (the master manipulator) x_m should be related to the force f_h exerted on the device by the human operator with an ideal or desired transfer function (matrix) $G_{rm}(s)$, i.e.,

$$\frac{X_m(s)}{F_h(s)} = G_{rm}(s); \tag{3.4}$$

- the slave displacement x_s, in the case when the master and the slave are identical, should follow the probably delayed master displacement x_{md} accurately; that is ideally,

$$x_s = x_{md} = x_m e^{-sT_1} \tag{3.5}$$

where T_1 denotes the communication delay from the master to the slave. In this way, the slave displacement x_s follows f_h asymptotically.

- once the slave interacts with the environment, the reaction force imposed by the latter should be reflected to the master.

3.1.3 The Process of Controller Synthesis

Since the system may operate under two scenarios, namely the free movement, i.e., no interaction between the slave and the environment, and the constrained movement, i.e., interaction existing, the controller synthesis includes accordingly the following two stages:

1) Under the situation that the master and the slave manipulators move freely with no force reaction generated by the environment (i.e., $f_e = 0$), it is clear that two objectives should be achieved: one is the desired master compliance, and the other is that the motion of the slave should follow the master's. Thus, a local controller for the master and a remote controller for the slave are designed, and the communication is from the master to the slave only.
2) In the case that the movements of the manipulators are constrained; that is, the slave manipulator interacts with the environment and a contact force is assumed to be sensed at the slave (i.e., $f_e \neq 0$), the transparency requirement is that any force sensed by the slave manipulator must be reflected back to the master manipulator, and furthermore, the controller yielded for this stage cannot affect the results in stage 1). In this stage a global controller is designed and the time-delay of the communication channel can be modeled as a linear plant with parametric and multiplicative uncertainty simultaneously.

The design for the three above-mentioned controllers consists of the following steps, as illustrated in Figure 3. 3 [10, 11, 36, 37]:

- determine the uncertainty models for all plants in the system or subsystems;
- select weighting functions in order to reflect the performance specifications for the system or subsystems;
- translate the given problem into a μ-synthesis problem; i.e., construct the system interconnection (structure P), which includes nominal plant models, controllers, weighting functions, and modeled uncertainties, by using LFT machinery;

- synthesize and analyze μ-controllers (D-K iterations) and weighting functions using μ-plots, controller frequency properties and closed-loop time responses;
- reduce the order of the resulting controller, and
- verify if the resulting controller satisfies the design specifications using simulation studies in the time domain.

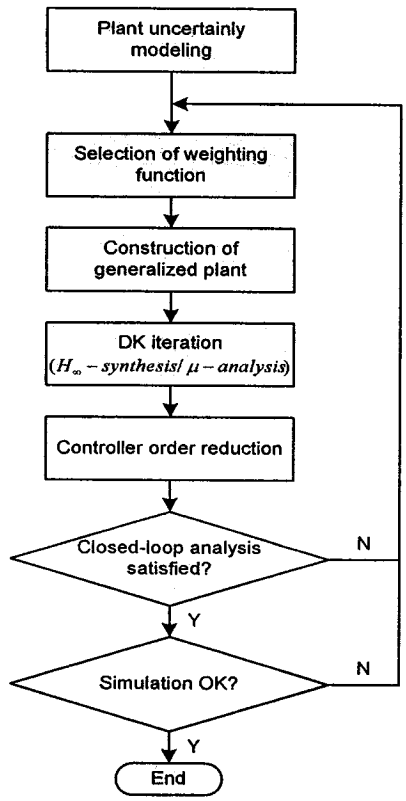


Fig. 3.3 The design cycle of the μ-controllers

The above steps may need to be carried out in an iterative way according to the situation involved. The large part of the design process consists of the selection of the weighting functions, the design of the μ-controllers, the evaluation of the resulting system and the redesign of the weighting functions.

The general guidelines for translating the controller design problem into the μ-synthesis problem are [40]:

- identify signals for the sets or vectors w and z such that all design objectives are reflected in the H_∞-norm of the transfer functions (matrices) between them while keeping the sets w and z as small as possible;
- select and make the order of uncertainties and performance weights as low as possible to avoid resulting in a high-order system interconnection P and thereby a high-order controller, which at least has the same order as that of P;
- associate the parameters of the weighting functions with design specifications since they will be tuned in the design process.

3.2 Modeling of the Uncertain Plants

One of the most important issues in the controller design for any system is the determination of the models for all the plants in the system.

3.2.1 Modeling the Manipulators

In this thesis, the μ-analysis and synthesis theory introduced in Chapter 2 is applied to a bilateral telemanipulation system like the one shown in Figure 3.2, in which two identical 3-DOF PHANToM v1.5 haptic devices, for simplicity of the description, act as the master and the slave manipulators. Taking the advantage of the result of [38], where the dynamics of the device was decoupled in three directions of the Cartesian space and there was no cross coupling between any two Cartesian dynamics, the discussion can be carried

out with one axis without any influence on the system analysis. Thus, the 1-DOF dynamics model along the X-axis direction is taken into account, without loss of generality, and the following 4-th order transfer function model from force to measured position in the X-axis direction, derived by Cavusoglu et. al. in [38], is then viewed as the nominal model of the master and the slave manipulators respectively:

$$G_0(s) = \frac{X(s)}{F(s)} = \frac{s^2 + 5.716\ \ s + 9.201\times 10^4}{s^2\ \ (3.329\times 10^{-6}\ \ s^2 + 0.001226\ \ s + 1.536)} \tag{3.6}$$

This linear model closely approximates the low frequency behaviour (up to about 200 *Hz*) of the manipulators in the X-axis direction; however, it does not include higher order dynamics [38]. Therefore, the actual plants are uncertain.

The assumption that the magnitudes of the actual plants deviate from that of the nominal plant G_0 expressed with equation (3.6) by as much as 5% at low frequencies and 100% at high frequencies is made in this thesis, according to the results in [38]. Therefore, the manipulators can be modeled as nominal plants with an input multiplicative uncertainty as shown in Figure 3.4:

$$G_m,\ G_s\ \in\ \{G_0(1+\Delta_{mul}W_{mul}) : \Delta_{mul} \in RH_\infty,\ \|\Delta_{mul}\|_\infty \le 1\}. \tag{3.7}$$

where the multiplicative uncertainty weighting function

$$W_{mul} = \frac{s+10}{s+200}, \tag{3.8}$$

which is frequency dependent and reflects a priori information about the neglected plant dynamics. At any frequency ω, the value of the robustness weight W_{mul} can be viewed as the percentage of uncertainty in the plant model at that frequency.

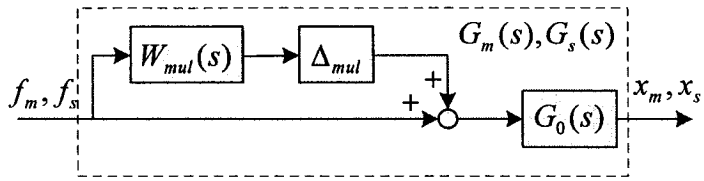


Fig. 3.4 Modeling of the manipulators

W_{mul} is usually chosen to be a filter with low gain at low frequencies and high gain at higher frequencies since the uncertainty in the model commonly increases with frequency.

3.2.2 Modeling of the Communication Channel

Anderson and Spong [1] have pointed out that the time delay existing between the master and the slave would destabilize a bilateral telemanipulation system. Among the researchers who addressed this problem, Leung et. al. [28] proposed a μ-synthesis framework for a pre-specified time delay. They treated the communication channel as a plant with the following transfer function

$$G_T(s) = e^{-Ts} = G_{T_0}(s) + \Delta_T(s), \tag{3.9}$$

where $G_{T_0}(s) = 1$ was the nominal model, $\Delta_T = e^{-Ts} - 1$ is an additive uncertainty, and T is a predefined fixed value representing the time delay of the communication channel. Furthermore, they assumed that the transmission delay present at the forward channel is the same as the one present at the feedback channel. Obviously, in this way the uncertainty of the communication channel was enlarged by choosing the nominal model as in [28]. Therefore, their method possessed some sort of conservativeness. On the other hand, transmission delay between master and slave usually varies with direction.

In this thesis, the transmission delays in both forward and feedback paths are treated as a second-order plant (by using Pade approximation) with parametric uncertainties and an input multiplicative uncertainty:

$$G_T(s) = \frac{Y_T(s)}{U_T(s)} = e^{-Ts} = \tilde{G}_T(s)[1 + \Delta_{Tm}(s)W_T(s)], \tag{3.10}$$

where

$$\tilde{G}_T(s) = \frac{s^{-2} - \frac{T}{2}s^{-1} + \frac{T^2}{8}}{s^{-2} + \frac{T}{2}s^{-1} + \frac{T^2}{8}} = 1 - \frac{Ts^{-1}}{s^{-2} + \frac{T}{2}s^{-1} + \frac{T^2}{8}} \tag{3.11}$$

with parametric uncertainties

$$\begin{cases} T = T_0(1 + \delta_T \Delta_T), \ \ T \in [T_1, T_2]; \\ T_0 = \frac{1}{2}(T_1 + T_2), \ \ \delta_T = \frac{1}{2}(T_2 - T_1), \ \ \|\Delta_T\| \le 1. \end{cases} \tag{3.12}$$

and the multiplicative uncertainty

$$\tilde{\Delta}_{Tm}(s) = \Delta_{Tm} W_T(s) = \frac{G_T(s) - \tilde{G}_T(s)}{\tilde{G}_T}, \ \ \|\Delta_{Tm}\|_\infty \le 1 \tag{3.13}$$

as shown in Figure 3.5, where $W_T(s)$ is a frequency dependent weighting function and can be found such that $|\tilde{\Delta}_{Tm}(j\omega)| \le |W_T(j\omega)|$. The parametric uncertainties are used to reflect the range of the delay-time T and the multiplicative uncertainties are used to reflect the modeling error. In this way, the modeling of the communication channel is more accurate and less conservative than previous methods.

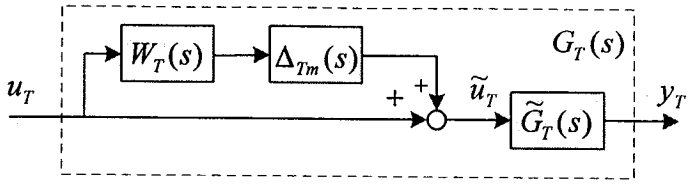


Fig. 3.5 Block Diagram of the Communication Channel Model

The nominal model of the communication channel is

$$G_{T_0}(s) = \frac{s^{-2} - \frac{T_0}{2}s^{-1} + \frac{T_0^2}{8}}{s^{-2} + \frac{T_0}{2}s^{-1} + \frac{T_0^2}{8}}. \tag{3.14}$$

In the design process we have assumed that $T_0 = 4$ seconds and $\delta_T = 0.2$. Figure 3.6 shows the block diagram of the transfer function $\tilde{G}_T(s)$ which is used for separating the parametric uncertainties from the nominal model $G_{T_0}(s)$ in order to build the interconnection structure of the whole communication channel.

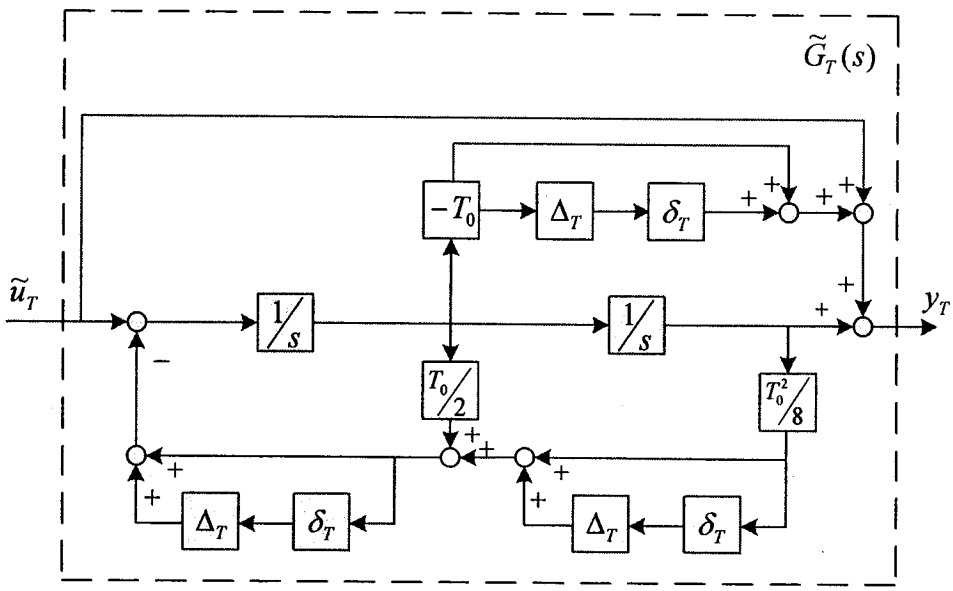


Fig. 3.6 The block diagram of $\tilde{G}_T(s)$

The weighting function $W_T(s)$ which is over the bounds

$$\left\| \frac{G_T(s) - \tilde{G}_T(s)}{\tilde{G}_T} \right\|_\infty \tag{3.15}$$

can be obtained by using Matlab function *fitmag* in μ-Analysis and Synthesis Toolbox [11] . Function *fitmag* fits a stable, minimum phase transfer function to magnitude data with a supplied frequency domain weighting function. The Matlab code *Wd.m* in Appendix A.1.1 is compiled to determine $W_T(s)$. Assuming that the time-delay T has a nominal value $T_0 = 4$ seconds and may deviate from this value by 20%, that is, $\delta_T = 0.2$ and $T \in [3.2,\ 4.8]$, we obtain

$$W_T(s) = \frac{2.3458\ \ s^2 + 0.5671\ \ s + 0.0320}{s^2 + 1.7263\ \ s + 2.3478}. \tag{3.16}$$

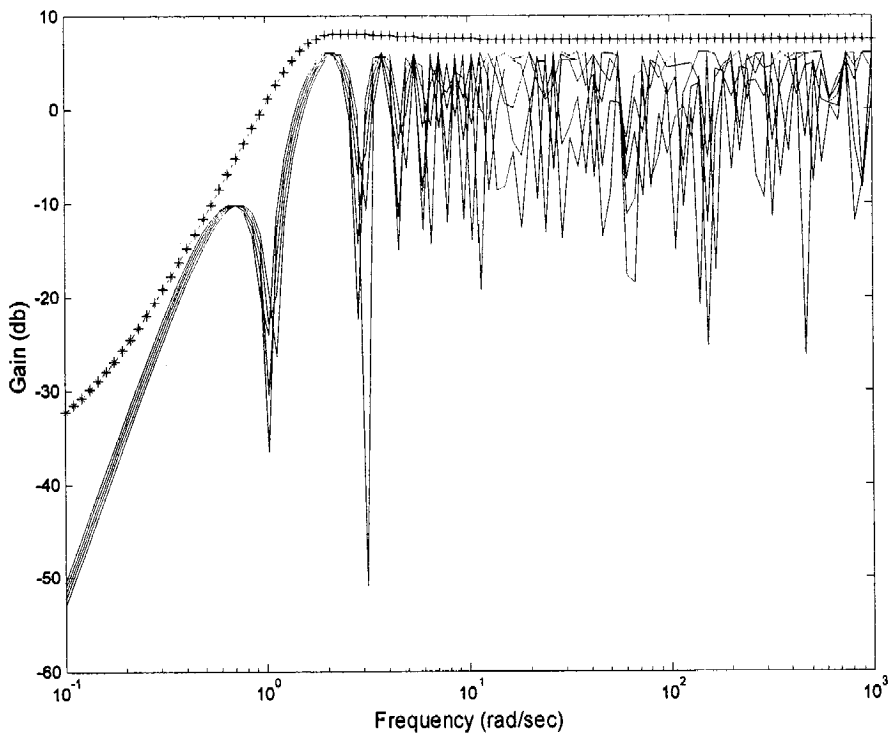


Fig. 3. 7 $\tilde{\Delta}_{Tm}(s)$ **(the dotted lines) and a bound** $W_T(s)$ **(the asterisked line)**

Figure 3.7 shows the fitting result by running the Matlab code *Wd2.m* in Appendix A.1.2. The top asterisked curve stands for the gain-magnitude plot of $W_T(s)$ and the dotted curves for the various input multiplicative uncertainties, namely

$$\tilde{\Delta}_{Tm}(s) = \frac{G_T(s) - \tilde{G}_T(s)}{\tilde{G}_T} \tag{3.17}$$

that could happen for the specified range of the parameter uncertainties.

3.2.3 Modeling the Environment

Suppose that the slave manipulator interacts with an environment modeled as a spring and a damper, as shown in Figure 3.8. Such an environment model is commonly used as a virtual wall in the virtual environment studies. Furthermore, suppose that there exist parametric uncertainties in this environment model, i.e.,

$$G_e(s) = \frac{F_e(s)}{X_s(s)} = B_e s + K_e \ , \tag{3.18}$$

where

$$\begin{cases} B_e = B_0(1 + db \cdot \Delta_b), \ \ B_0 = 0.0175, \ \ db = 0.5, \ \ \|\Delta_b\| \le 1; \\ K_e = K_0(1 + dk \cdot \Delta_k), \ \ K_0 = 0.35, \ \ dk = 0.5, \ \ \|\Delta_k\| \le 1. \end{cases} \tag{3.19}$$

Thus, the nominal model becomes

$$G_{e0}(s) = B_0 s + K_0, \tag{3.20}$$

which is the same as the one in [39], but the variation ranges of B_e and K_e are different from the ones in that paper.

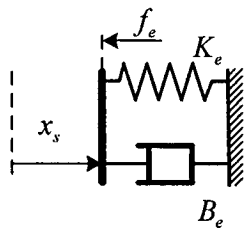


Fig. 3.8 Modeling the Environment

3.3 Synthesis of the Controllers

Two cases are considered in the controller design: the slave manipulator interacts with the environment (constrained movement) or not in contact with the environment (free movement). Thus, the controller design process is divided into two stages respectively. In the first stage, the free movement case is dealt with since the two manipulators themselves should be stable in any case with the requirement that the whole system is internally stable. The second stage involves the constrained movement case.

3.3.1 Controllers Design for the Free Movement Case

In this case, the remote manipulator has no interaction with the environment so that there is no force feedback from the environment. According to the design objectives described in Section 3.1.2, a desired master compliance and a desired slave tracking ability should be obtained. In addition, there is only a unidirectional communication channel between the master and the slave, i.e., the path is from the master to the slave. Such a system is obviously a causal system. Thus, the master and slave controllers can be separately

designed. Since the slave subsystem is a position tracker, as long as the two subsystems meet their respective performance requirements, the telemanipulation system will asymptotically realize its performance specifications in the face of time delays in the communication channel. The telemanipulation system configuration for this situation is shown in Figure 3.9.

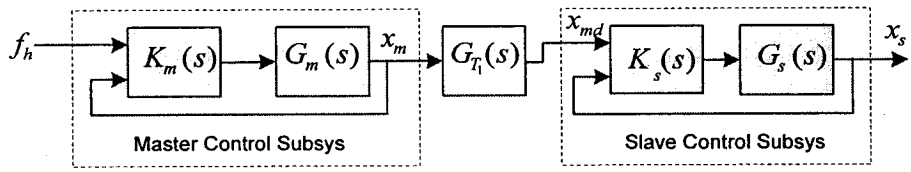


Fig. 3.9 The System Configuration for the Free Movement Case

3.3.1.1 Controller Design for the Master Manipulator

According to the objectives described in Section 3.1.2 and applying the H_∞-control method and the μ-synthesis theory introduced in Chapter 2, the master controller should be designed such that

- The closed-loop stability of the master control subsystem is achieved.
- Condition (3.4) is satisfied; i.e., a desired master compliance is achieved. Therefore, the compliance error $(x_m - G_{rm} f_h)$ should be weighted and included in the error signal set, in which the components (or the signals) are to be controlled, using the μ-synthesis machinery.
- The output of the controller u_m, which is the input to the master manipulator f_m, does not exceed saturation limits, which implies that it should also be weighted and included in the error signal set.

- A measurement noise n_m at the output end (x_m), which is viewed as a disturbance, is introduced to make the problem of general nature.

A diagram for the synthesis of the master controller $K_m = [K_{m1} \quad K_{m2}]$ can be configured as shown in Figure 3.10, where W_{em}, W_{um}, W_{nm} and W_{mul} are frequency dependent weighting functions.

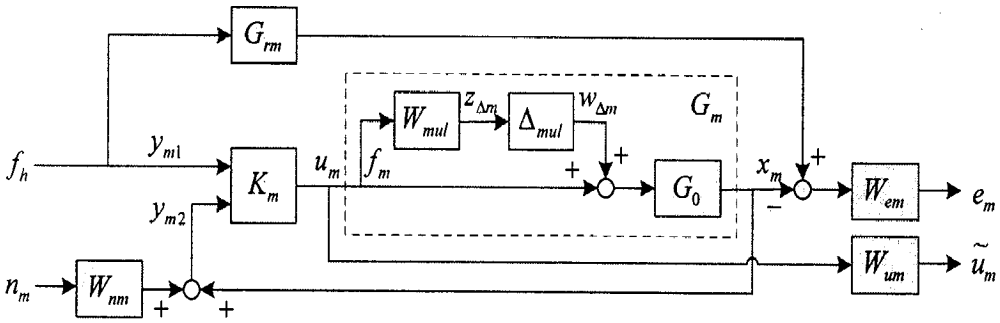


Fig. 3.10 Master Controller Synthesis Diagram

The Master Manipulator Model $G_m(s)$

The master manipulator dynamics $G_m(s)$ in Figure 3.10 is determined by using equations (3.6), (3.7) and (3.8), which has been discussed in Section 3.2.1.

The Reference Model $G_{rm}(s)$

$G_{rm}(s)$ in Figure 3.10 represents the reference model with ideal manipulating qualities. The controller design process attempts to shape the closed-loop response of the master control subsystem to match $G_{rm}(s)$.

For good command tracking response it is desired that the closed-loop system responds as a well-damped second-order system. To determine the optimal relationship between the force input and the displacement output for the PHANToM haptic device is beyond the scope of this thesis. Therefore, for simplicity, it is assumed that the steady-state value of master displacement must be equal to the force command. Specifically, by considering the bandwidth of the PHANToM haptic device, the reference model is now selected as

$$G_{rm}(s) = \frac{X_m(s)}{F_h(s)} = \frac{142^2}{s^2 + 2\times 0.7\times 142\ s + 142^2}. \tag{3.21}$$

The Controller $K_m(s)$

The controller can be designed as of either a 1-DOF or a 2-DOF configuration. The 1-DOF controllers for servo problems have been widely applied in practice. However, with these controllers resulting from μ-synthesis, the closed-loop systems often possess unsatisfactory transient properties: the overshoot exceeds the acceptable values and the oscillations in the responses are significant and excessive [41].

Indeed, before the 2-DOF controller structure is finally selected in this thesis, a 1-DOF controller for the master subsystem was synthesized with the same specifications as in the 2-DOF controller synthesis. Figure 3.11 shows the step response of the closed-loop system with the full order (without any model reduction) 1-DOF controller resulting from the μ-synthesis framework described in Chapter 2. Obviously such a large overshoot is not satisfactory and other controller design should be made. Thus, a 2-DOF structure is chosen although it results in higher order controllers. For comparison, the step responses

of the master subsystem with the 2-DOF controller generated later in this subsection can be found in Figure 3.14.

The controller $K_m(s) = [K_{m1}(s) \quad K_{m2}(s)]$ in Figure 3.10 is to be designed as a 2-DOF controller, where $K_{m1}(s)$ is acted as a command path filter to provide the closed-loop system with satisfied command response from f_h to x_m while $K_{m2}(s)$ provides stability and disturbance rejection. The synthesis process about $K_m(s)$ will be presented later.

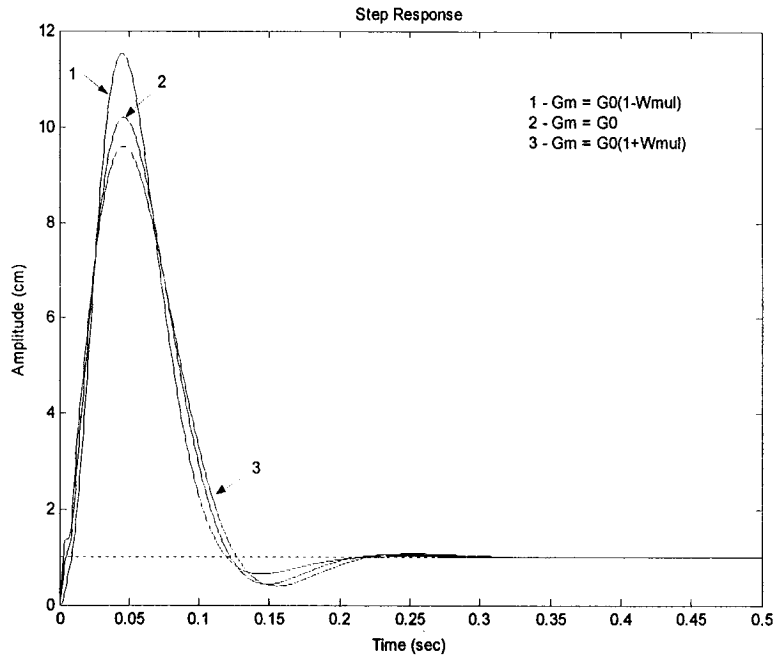


Fig. 3.11 Step Responses of the Master Subsystem with 1-DOF Controller (Δ_{mul} = 0, -1 and 1)

The Weighting Functions

The weighting function $W_{em}(s)$ is chosen for the master compliance error and can be thought of as a penalty function. It scales this performance error to unity; that is, it

weights the difference between the response of the reference model and the response of the closed-loop system. Since the error is scaled to unity, it has to stay below the inverse of W_{em} at every frequency. Therefore, the desired performance is achieved when

$$\| W_{em}(j\omega)(G_{rm}(j\omega) - H_{cl}(j\omega)) \|_{\infty} < 1. \tag{3.22}$$

The selection principle for W_{em} is that it should be large in the frequency range where small errors are desired and small where larger errors can be tolerated. The weight should be large at very low frequencies in order to achieve zero steady errors.

The final selection for W_{em} after a number of trial and errors and with considerations for the bandwidth of the plant yields

$$W_{em}(s) = \frac{0.01\ \ s+1}{s+0.01}, \tag{3.23}$$

which implies that the desired steady-state error is 0.01.

The control weighting function $W_{um}(s)$ is a performance criterion used to limit the control input; i.e., to penalize larger deflections and therefore minimize control activity. It scales the admissible control input to the plant to unity.

In the case of the master control subsystem, the choice for $W_{um}(s)$ is selected as

$$W_{um}(s) = \frac{10^{-3} s}{10^{-6} s + 1}, \tag{3.24}$$

which is close to a differentiator to penalize fast changes and large overshoot in the control input.

The noise weighting function $W_{nm}(s)$ scales the measurement noise or other noise entering the system. Since it is used to reflect high frequency sensor noise, it often takes a form of a high-pass filter. Typically, this weight is a first order transfer function with gain selected to produce the correct signal level and time constant selected to match the bandwidth of the signal. Here, $W_{nm}(s)$ is chosen as

$$W_{nm}(s) = \frac{0.1(s+0.01)}{s+200} \tag{3.25}$$

which implies that the master control subsystem has to be robust to a maximum 10% deviation from the measurement on x_m .

Since these weighting functions are responsible for the characteristics of the resulting controller, the success of the design depends on their selection. Therefore, design conditions and trade offs, such as robustness versus performance as well as performance versus control activity, are reflected in their selection.

Controller Synthesis Problem Formulation

According to the theory described in Chapter 2, if the robust performance is satisfied, the nominal stability, the robust stability and the nominal performance are also satisfied. Robust performance is achieved if the μ-plot stays below unity over the entire frequency range.

The master controller design problem can be translated to a H_∞-control and μ-synthesis problem, and the control subsystem in Figure 3.10 can be transformed to the general

formulation shown in Figure 3.12, by pulling out the input multiplicative uncertainty Δ_{mul} and the controller K_m, where

$$z_m = \begin{bmatrix} e_m \\ \tilde{u}_m \end{bmatrix},\ w_m = \begin{bmatrix} f_h \\ n_m \end{bmatrix},$$

$$y_m = \begin{bmatrix} y_{m1} \\ y_{m2} \end{bmatrix} = \begin{bmatrix} f_h \\ x_m + W_{nm} n_m \end{bmatrix} = \begin{bmatrix} f_h \\ G_0 W_{\Delta m} + W_{nm} n_m + G_0 u_m \end{bmatrix},$$

$u_m = u_m$, and $\Delta_m = \Delta_{mul}$.

With $\begin{bmatrix} z_{\Delta m} & z_m & y_m \end{bmatrix}^T = P_m \begin{bmatrix} w_{\Delta m} & w_m & u_m \end{bmatrix}^T$, it is not difficult to obtain the open-loop interconnection structure of the system, also called the augmented plant which includes the nominal model and the weighting functions, by disconnecting the controller K_m and the uncertainty Δ_m from the closed-loop structure shown in Figure 3.12(a):

$$P_m = \begin{bmatrix} P_{m_{11}} & P_{m_{12}} & P_{m_{13}} \\ P_{m_{21}} & P_{m_{22}} & P_{m_{23}} \\ P_{m_{31}} & P_{m_{32}} & P_{m_{33}} \end{bmatrix} = \left[\begin{array}{c|cc|c} 0 & 0 & 0 & W_{mul} \\ \hline -W_{em}G_0 & W_{em}G_{rm} & 0 & -W_{em}G_0 \\ 0 & 0 & 0 & W_{um} \\ \hline 0 & 1 & 0 & 0 \\ G_0 & 0 & W_{nm} & G_0 \end{array}\right] \tag{3.26}$$

and the closed- loop interconnection of the system

$$M_m = \begin{bmatrix} M_{m_{11}} & M_{m_{12}} \\ M_{m_{21}} & M_{m_{22}} \end{bmatrix} = F_l(P_m, K_m) = \begin{bmatrix} P_{m_{11}} & P_{m_{12}} \\ P_{m_{21}} & P_{m_{22}} \end{bmatrix} + \begin{bmatrix} P_{m_{13}} \\ P_{m_{23}} \end{bmatrix} K_m \left(I - P_{m_{33}} K_m\right)^{-1} \begin{bmatrix} P_{m_{31}} & P_{m_{32}} \end{bmatrix} \tag{3.27}$$

in Figure 3.12(b). The code *Master.m* in Appendix A is compiled to generate P_m in the form of Matlab *SYSTEM* matrix so that the routine *dkitgui* [11], a very good graphical user interface provided by Matlab, can be executed to carry out *D-K* iteration. Also the code *Pmu.m* in Apendix A is compiled to generate M_m and carry out μ-analysis on the closed-loop system.

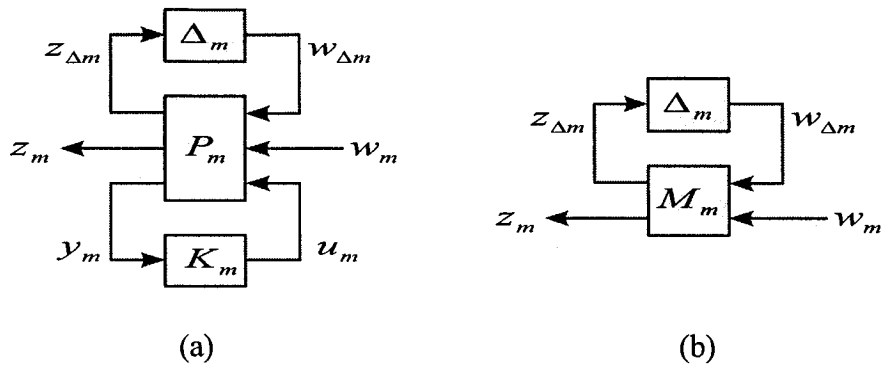


Fig. 3.12 Generalization of the Master Control Synthesis Problem

D-K Iteration

Once the generalized plant is formed, the next task is to solve the following *D-K* iteration problem:

$$\min_{\substack{K_m \\ \text{stabilizing}}} \min_{D \in \mathbf{D}} \| D^{-1} M_m(j\omega) D \|_\infty \qquad (3.28)$$

Routine *dkitgui* [11] can carry out automated, parameter adjustable or tunable, and visual iterations. It makes the designer monitor the progress quite easily. To make the order of the final global controller K_g as low as possible, and with the help of this useful tool and the model reduction command *schmr* [11], the following seventh-order controller $K_m = [K_{m1} \quad K_{m2}]$ with satisfied performance is generated and taken for the master subsystem, instead of the full 10-th order controller generated by *dkitgui*:

$$K_{m1} = \frac{-0.01451\ s^6 + 1606\ s^5 + 3.846\times10^6\ s^4 + 5.75\times10^9\ s^3 + 1.448\times10^{12}\ s^2 + 5.323\times10^{14}\ s + 8.462\times10^{16}}{s^7 + 3595\ s^6 + 6.349\times10^6\ s^5 + 4.848\times10^9\ s^4 + 1.231\times10^{12}\ s^3 + 4.189\times10^{14}\ s^2 + 4.979\times10^{16}\ s + 4.979\times10^{14}} \qquad (3.29)$$

and

$$K_{m2} = \frac{-0.8139\ s^6 - 8.156\times10^5\ s^5 - 5.04\times10^8\ s^4 - 4.6\times10^{11}\ s^3 - 9.729\times10^{13}\ s^2 - 4.179\times10^{15}\ s - 8.464\times10^{16}}{s^7 + 3595\ s^6 + 6.349\times10^6\ s^5 + 4.848\times10^9\ s^4 + 1.231\times10^{12}\ s^3 + 4.189\times10^{14}\ s^2 + 4.979\times10^{16}\ s + 4.979\times10^{14}} \qquad (3.30)$$

Figure 3.13 shows that only one iteration is needed for generating the desired K_m. Code *Srm.m* in Appendix A is compiled to draw the step responses of the closed-loop master control subsystem with the 7-th order K_m under three typical situations: $G_m = G_0$ (nominal model), $G_m = G_0(1+W_{mul})$, and $G_m = G_0(1-W_{mul})$, as shown in Figure 3.14. More details about this controller and the μ-analysis on it will be given in Chapter 4.

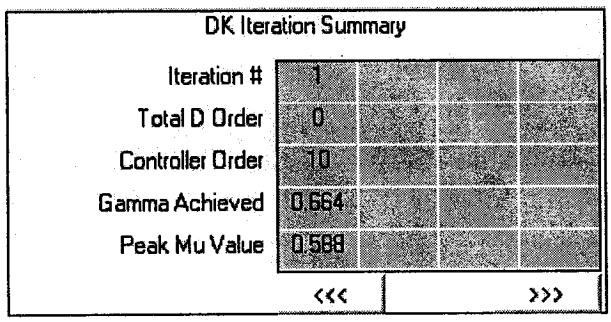


Fig. 3.13 *D-K* Iteration for the Synthesis of K_m

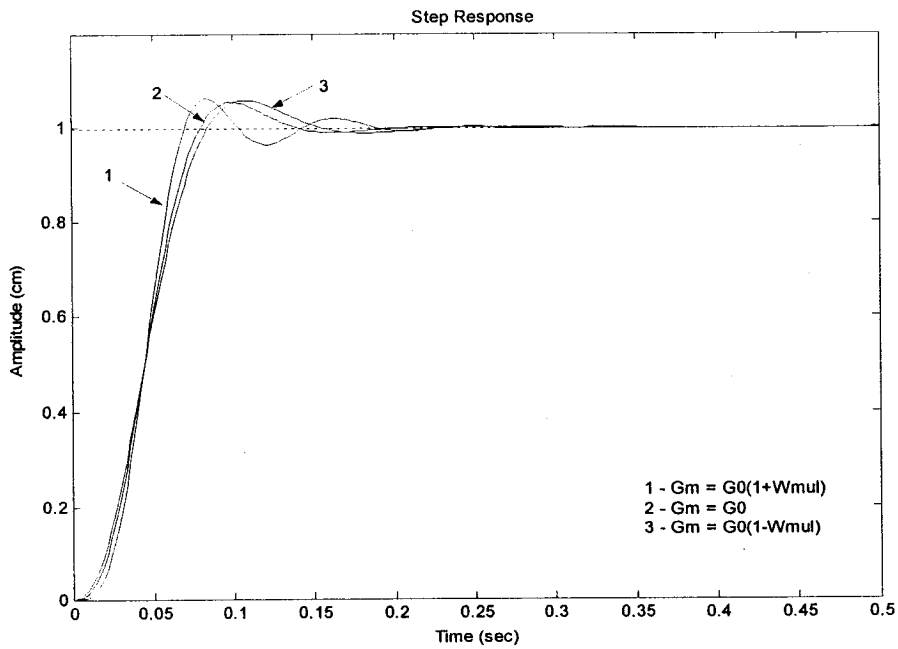


Fig 3.14 Step Response of the Closed-Loop Master Subsystem

3.3.1.2 Controller Design for the Slave Manipulator

The slave manipulator is treated as a position follower as shown in Figure 3.15. Obviously the slave subsystem structure is quite similar to the master's except for the input signal. W_{es}, W_{us} and W_{ns} are frequency dependent weighting functions, which take the same forms as those of W_{em}, W_{um} and W_{nm} in equations (3.23), (3.24) and (3.25). In addition, since the dynamics of both the master and the slave are assumed to be identical, the ideal reference model for the slave is taken as $G_{sr}(s) = 1$. Therefore, the requirements for the slave controller $K_s = [K_{s1} \quad K_{s2}]$ design are:

- The closed-loop stability of the slave control subsystem should be achieved.
- A desired slave tracking ability should be achieved; that is, the tracking error $(x_{md} - x_s)$ should be weighted and included in the error signal set of the μ-synthesis machinery.
- The output of the controller, u_s, acting as the input to the slave manipulator, should not exceed the saturation limits, which implies that it should also be weighted and included in the error signal set.
- A measurement noise n_s at the output end (x_s) is considered as a disturbance to make the problem more general so that the system robustness could be increased.

In the same way that was dealt with the master control subsystem, and with the relationship $[z_{\Delta s} \quad e_s \quad \tilde{u}_s \quad y_s]^T = P_s[w_{\Delta s} \quad x_{md} \quad n_s \quad u_s]^T$ where $y_s = [y_{s1} \quad y_{s2}]^T$, the open-loop interconnection structure P_s for Figure 3.16 (a) can be calculated as:

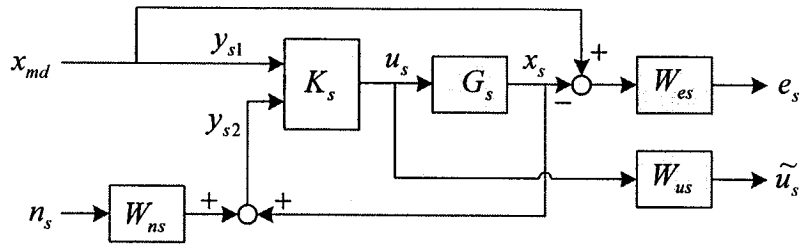
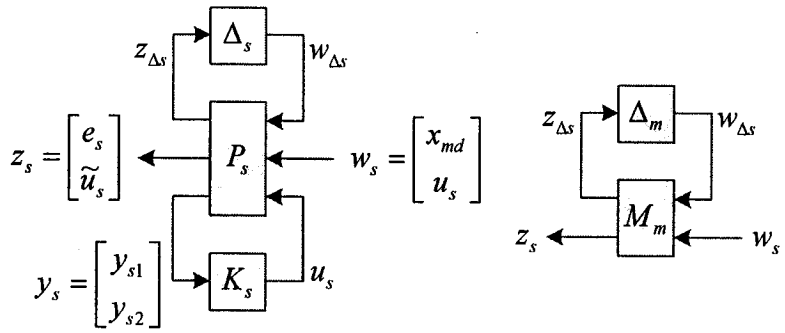


Fig. 3.15 Controller Synthesis Structure for the Slave



(a) **(b)**

Fig. 3.16 Generalization of the Slave Control Subsystem

$$P_s = \begin{bmatrix} P_{s_{11}} & P_{s_{12}} & P_{s_{13}} \\ P_{s_{21}} & P_{s_{22}} & P_{s_{23}} \\ P_{s_{31}} & P_{s_{32}} & P_{s_{33}} \end{bmatrix} = \left[\begin{array}{ccc|c} 0 & 0 & 0 & W_{mul} \\ -W_{es}G_0 & W_{es} & 0 & -W_{es}G_0 \\ \hline 0 & 0 & 0 & W_{us} \\ 0 & 1 & 0 & 0 \\ \hline G_0 & 0 & W_{ns} & G_0 \end{array}\right] \tag{3.31}$$

The lower LFT $F_l(P_s, K_s)$ is equal to the closed-loop interconnection M_s in Figure 3.16(b); therefore, it can be obtained as follows

$$M_s = \begin{bmatrix} M_{s_{11}} & M_{s_{12}} \\ M_{s_{21}} & M_{s_{22}} \end{bmatrix} = F_l(P_s, K_s) = \begin{bmatrix} P_{s_{11}} & P_{s_{12}} \\ P_{s_{21}} & P_{s_{22}} \end{bmatrix} + \begin{bmatrix} P_{s_{13}} \\ P_{s_{23}} \end{bmatrix} K_s \left(I - P_{s_{33}} K_s\right)^{-1} \begin{bmatrix} P_{s_{31}} & P_{s_{32}} \end{bmatrix}. \tag{3.32}$$

The reduced-order slave controller $K_s = [K_{s1} \quad K_{s2}]$ can be obtained using *dkitgui* and function *schmr* as follows:

$$K_{s1} = \frac{0.001508\ s^6 + 1513\ s^5 + 3.861\times10^6\ s^4 + 5.525\times10^9\ s^3 + 1.434\times10^{12}\ s^2 + 5.114\times10^{14}\ s + 8.344\times10^{16}}{s^7 + 3592\ s^6 + 6.341\times10^6\ s^5 + 4.833\times10^9\ s^4 + 1.224\times10^{12}\ s^3 + 4.173\times10^{14}\ s^2 + 4.921\times10^{16}\ s + 4.921\times10^{14}} \tag{3.33}$$

and

$$K_{s2} = \frac{-0.8056\ s^6 - 8.061\times10^5\ s^5 - 4.978\times10^8\ s^4 - 4.544\times10^{11}\ s^3 - 9.601\times10^{13}\ s^2 - 4.121\times10^{15}\ s - 8.348\times10^{16}}{s^7 + 3592\ s^6 + 6.341\times10^6\ s^5 + 4.833\times10^9\ s^4 + 1.224\times10^{12}\ s^3 + 4.173\times10^{14}\ s^2 + 4.921\times10^{16}\ s + 4.921\times10^{14}} \tag{3.34}$$

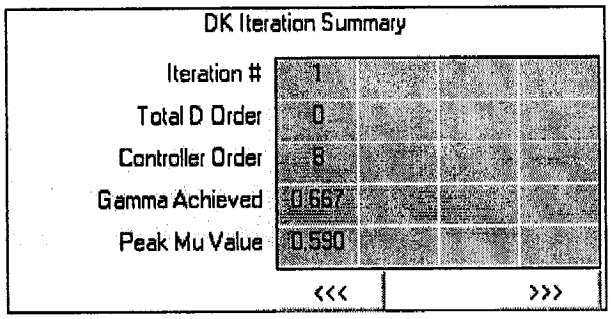


Fig. 3.17 *D-K* Iteration for the Synthesis of K_s

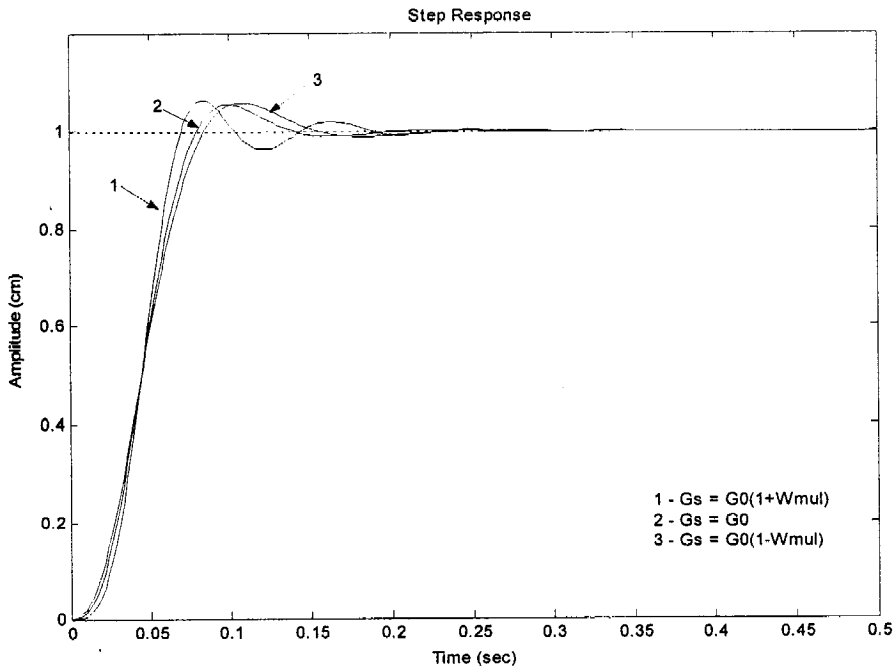


Fig. 3.18 Step Response of the Closed-loop Slave Subsystem

Similar to Figure 3.13, Figure 3.17 shows that only one iteration is needed for generating the desired K_s. With running Matlab code *Slave.m* in Appendix A, routine *dkitgui* and *Kmr.m* in Appendix A sequentially, code *Srs.m* in Appendix A is used to draw the step responses of the closed-loop slave control subsystem with the 7-th order controller K_s under three typical situations: $G_s = G_0$ (nominal model), $G_s = G_0(1+W_{mul})$, and $G_s = G_0(1-W_{mul})$, are now shown in Figure 3.18. More details about this controller design and the μ-analysis will also be given in Chapter 4.

3.3.1.3 The Telemanipulation for the Free Movement Case

Once the master and the slave controllers are designed, the remaining synthesis task for the free movement case is to build the telemanipulation system as shown in Figure 3. 9, where G_0 is denoted as G_1 / s^2 for the purpose of the simulations. Figure 3.19 shows the system simulation diagram when $G_m = G_s = G_0(1+W_{mul})$. Simulations for other typical situations, such as $G_m = G_0(1+W_{mul})$, $G_s = G_0(1-W_{mul})$, can also be made by setting $W_{mul} = 0$ or changing its sign (strictly speaking, by setting the uncertainty $\Delta_{mul} = 0,\ -1$ or $+1$) in Figure 3.19.

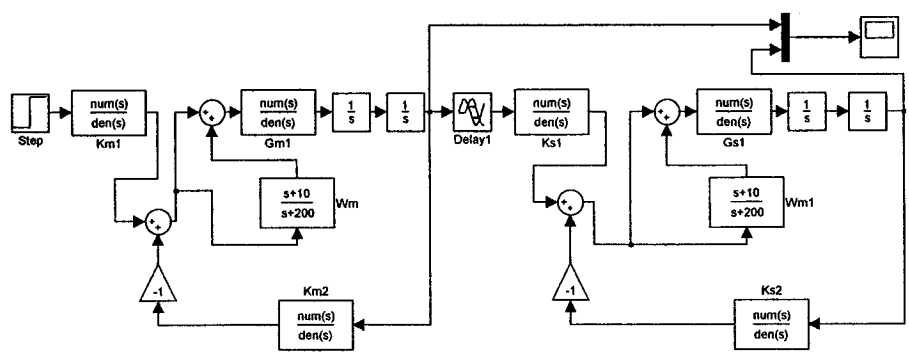
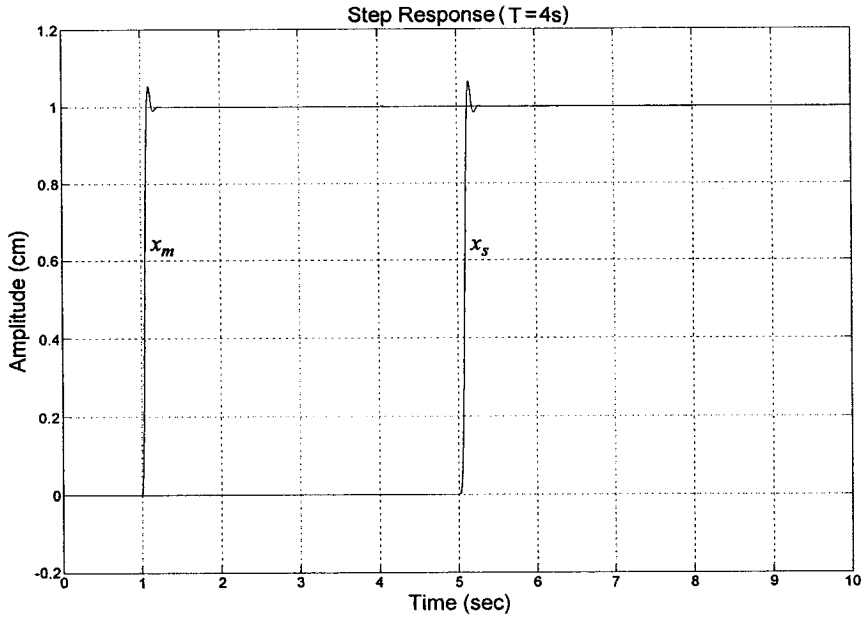
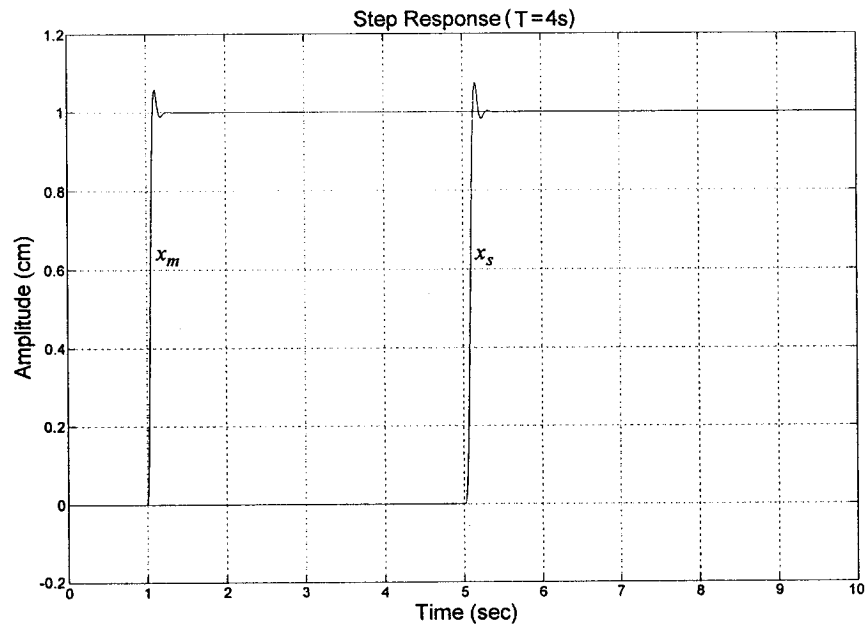


Fig. 3. 19 Simulation for the Free Movement Case



(a) $G_m = G_s = G_0$



(b) $G_m = G_s = G_0(1+W_m)$

Fig. 3.20 The Step Response of x_m and x_s : the Free Movement Case

The simulation results for the step responses of x_m and x_s, when $T_0 = 4$ (sec.), for cases $G_m = G_s = G_0$ and $G_m = G_s = G_0(1 + W_{mul})$ are shown respectively in Figure 3.20 (a) and (b).

In fact, with the controllers designed above, the telemanipulation system is always stable for all the values of the time delay.

3.3.2 Controller Design for the Constrained Movement Case

In this case, the slave manipulator will interact with the environment. The telemanipulator will encounter the resistance force from the environment. The overall control system is chosen as the one shown in Figure 3.2. Obviously, the requirements for the design of the global controller $K_g = [K_{g_1} \quad K_{g_2}]$ now become:

- The system should be stable in the presence of uncertainties in all plants including the uncertain time-delay of the communication channel;
- The design of K_g should not adversely affect the free movement already designed for; that is to say, K_m and K_s should be fixed in the system design and the compliance error $(x_m - G_{rm} f_h)$ should still be controlled so that it should be weighted and included in the error signal set; in addition, since the system is causal, the slave displacement x_s should be able to asynchronously track the master displacement x_m in the face of time delay in the communication channel, so that $(x_{md} - x_s)$ should also be weighted and included into the error signal set;

- The environment feedback force f_e should be felt by the master and the output of the global controller which is located in the master site (or local site), u_g, should reflect this force (in Chapter 4, we use $\tilde{f}_e$, instead of u_g, to denote this reflection). Therefore, the difference between these two factors, $(u_g - f_e)$, should be controlled; therefore, $(u_g - f_e)$ should be weighted and included in the error signal set;
- The output of the global controller u_g should not exceed a saturation limit, so it should be weighted and included into the error signal set in order to be controlled;
- Measurement noise n_g at the output end (x_s), which is viewed as a disturbance, is considered to make the problem more general and therefore make the system more robust.

Based on the above requirements, a block diagram for the synthesis of the global controller $K_g = [K_{g_1} \quad K_{g_2}]$ can be configured as shown in Figure 3.21, where W_{eg1}, W_{eg2} and W_{eg3} are performance weighting functions which reflect the design specifications on the system performance, while W_{ug} is a control weight on the output of the controller K_g and W_{ng} is the weight for modeling the noise signal. These weights are all frequency dependent. Since the requirements on the weighted signals are quite similar to the ones in the controller design for the free movement case, the selection of these weighting functions remain the same as described by equations (3.23) to (3.25); that is,

$$W_{eg1}(s) = W_{eg2}(s) = W_{eg3}(s) = \frac{0.01\ \ s+1}{s+0.01},\ W_{ug}(s) = \frac{10^{-3}s}{10^{-6}s+1} \text{ and } W_{ng}(s) = \frac{0.1(s+0.01)}{s+200}.$$

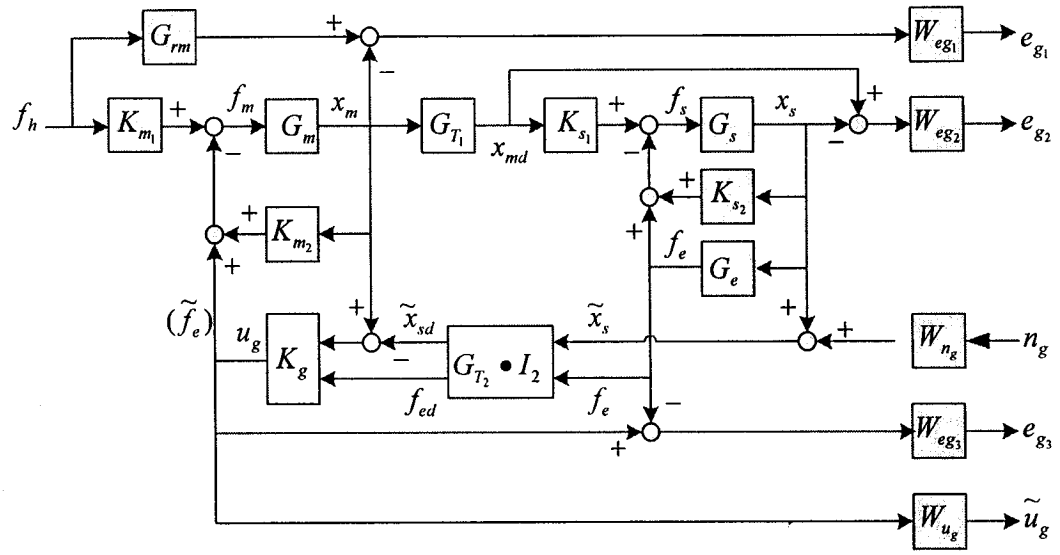


Fig. 3.21 Block Diagram for Synthesizing Global Controller K_g

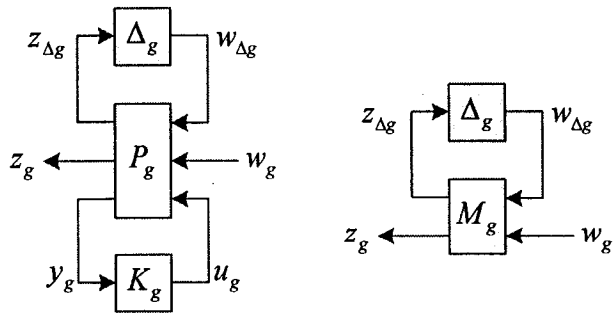
It should be noted that K_{m2} and K_{s2} in Figure 3.21 are respectively negative K_{m2} and K_{s2} in equations (3.30) and (3.34), in order to cope with the negative feedback nature.

In Figure 3.21, the blocks G_{T_1} and G_{T_2} represent the time-delayed communication channels from the master to the slave and from the slave to the master, respectively, which are modeled according to equations (3.10) to (3.17) as a second-order system with parametric uncertainty plus an input multiplicative uncertainty as described in Section 3.2.2, except for assuming that the amount of the delay in the two channels may not have to be the same.

The environment model, G_e, is assumed to consist of a spring and a damper with the structure expressed in equations (3.18)-(3.20).

Furthermore, there are altogether 16 uncertain blocks in the global systems: 5 dynamic (1 for the master, the slave, and the communication channel from the master to the slave (CCMS); 2 for the communication channel from the slave to the master (CCSM) and 11 parametric (3 for CCMS, 6 for CCSM, and 2 for the environment).

Based on the above considerations and the structures of the plants, it can be stated that for the constrained system based on the standard structure description (shown in Figure 3.22), the error set z_g in which the signals should be controlled include e_{g1}, e_{g2}, e_{g3}, and $\tilde{u}_g$; the exogenous input set w_g consists of the force applied by the human operator, f_h, and the sensor noise at the slave side, n_g; the measurement signal vector y_g includes the difference between the master displacement and the delayed and disturbed slave displacement $\tilde{x}_{sd}$, and the delayed environment reaction force; and finally, the control input to the system open-loop interconnection structure is the output of the global controller, u_g.



(a) (b)

Fig. 3.22 Generalized Global System for Constrained Movement Case

Pulling out all the uncertain sources and K_g, the open interconnection of the global system P_g can be obtained with the following relationship:

$$\begin{aligned}&\left[\begin{bmatrix} z_{g1} & \cdots & z_{g16} \end{bmatrix} \begin{bmatrix} e_{g1} & e_{g2} & e_{g3} & \tilde{u}_g \end{bmatrix} \begin{bmatrix} (x_m - \tilde{x}_{sd}) & f_{ed} \end{bmatrix}\right]^T \\ &= P_g \left[\begin{bmatrix} w_{g1} & \cdots & w_{g16} \end{bmatrix} \begin{bmatrix} f_h & n_g \end{bmatrix} u_g\right]^T\end{aligned} \tag{3.35}$$

Since the system is of high order, especially having a number of uncertainty blocks, it is not easy to write out P_g explicitly, but with Matlab command *sysic*, it is convenient to get the *SYSTEM* matrix form of P_g. Code *Globalpf.m* in Appendix A is compiled to compute the P_g matrix. Using command *minfo*, it can be concluded that the open system interconnection structure possesses 57 states, 22 outputs and 19 inputs.

Consequently, the lower LFT $F_l(P_g, K_g)$ is equal to the closed-loop interconnection M_g in Figure 3.22(b) and so can be obtained as

$$M_g = \begin{bmatrix} M_{g_{11}} & M_{g_{12}} \\ M_{g_{21}} & M_{g_{22}} \end{bmatrix} = F_l(P_g, K_g) = \begin{bmatrix} P_{g_{11}} & P_{g_{12}} \\ P_{g_{21}} & P_{g_{22}} \end{bmatrix} + \begin{bmatrix} P_{g_{13}} \\ P_{g_{23}} \end{bmatrix} K_g \left(I - P_{g_{33}} K_g\right)^{-1} \begin{bmatrix} P_{g_{31}} & P_{g_{32}} \end{bmatrix}. \tag{3.36}$$

The reduced-order global controller $K_g = [K_{g1} \quad K_{g2}]$ can be obtained using *dkitgui* and function *schmr*.

Note that by invoking *dkitgui* under current conditions it is impossible to get a γ-plot and a μ-plot below unity. Figure 3.23 and Table 3.1 show that the best controller generated after the seventh D-K iteration with γ = 2.8648 and μ = 2.8018 implying that the global system will not achieve robust performance.

In Figure 3.23, the maximum singular value of the frequency response of the closed-loop system, $\overline{\sigma}(M_g)$, where $M_g = F_l(P_g, K_g)$ is defined by equation (3.36), and the maximum value of the structured singular value, $\mu(M_g)$, after the 7-th D-K iteration are shown.

Fig. 3.23 γ- and μ-plot after 7-th D-K Iteration

Table 3.1 Summary of the D-K Iteration for K_g Design

Iteration	1	2	3	4	5	6	7
Total D-scale Order	0	44	80	84	74	72	72
Controller Order	57	101	137	141	131	129	129
γ	101.6455	4.9570	4.1687	6.3708	4.2863	10.3458	2.8648
Peak μ	22.2222	4.2806	2.7242	2.8572	3.0339	3.2815	2.8018

From Table 3.1, it can also be concluded that the generated global controller is of the129-th order which should be clearly reduced. By setting 0.01 as the error tolerance level and using the Matlab function *schmr*, a 9-th order controller can be obtained as follow:

$$K_{g1} = \frac{-0.002201\ s^8 + 0.01701\ s^7 + 0.04113\ s^6 + 0.02764\ s^5 + 0.009776\ s^4 + 0.0004461\ s + 3.791\times10^{-6}\ s^2 + 2.386\times10^{-8}\ s + 5.454\times10^{-11}}{s^9 + 1.475\ s^8 + 1.184\ s^7 + 0.5409\ s^6 + 0.1502\ s^5 + 0.02057\ s^4 + 0.0008155\ s^3 + 7.124\times10^{-6}\ s^2 + 4.456\times10^{-8}\ s + 1.075\times10^{-10}} \quad (3.37)$$

and

$$K_{g2} = \frac{2.354\times10^{-12}\ s^9 + 0.004531\ s^8 + 0.03471\ s^7 + 0.06112\ s^6 + 0.04162\ s^5 + 0.01563\ s^4 + 0.0007582\ s + 6.708\times10^{-6}\ s^2 + 4.294\times10^{-8}\ s + 1.047\times10^{-10}}{s^9 + 1.475\ s^8 + 1.184\ s^7 + 0.5409\ s^6 + 0.1502\ s^5 + 0.02057\ s^4 + 0.0008155\ s^3 + 7.124\times10^{-6}\ s^2 + 4.456\times10^{-8}\ s + 1.075\times10^{-10}} \tag{3.38}$$

It will be shown in Chapter 4 that with this global controller the telemanipulation system can achieve the nominal performance and the robust stability in the constrained movement case. Although it does not meet the robust performance requirement in frequency domain, it performs very well in time domain.

Figure 3.24 shows the simulation result of the global control system when the command f_h is set as a step input at time t = 1 second when $G_m = G_s = G_0(1+W_{mul})$, $G_e = B_0(1+db)s + K_0(1+dk)$ and the time delay of the communication channel is $T = 4$ seconds. It can be seen that the displacements x_m and x_s follow the command f_h quite well.

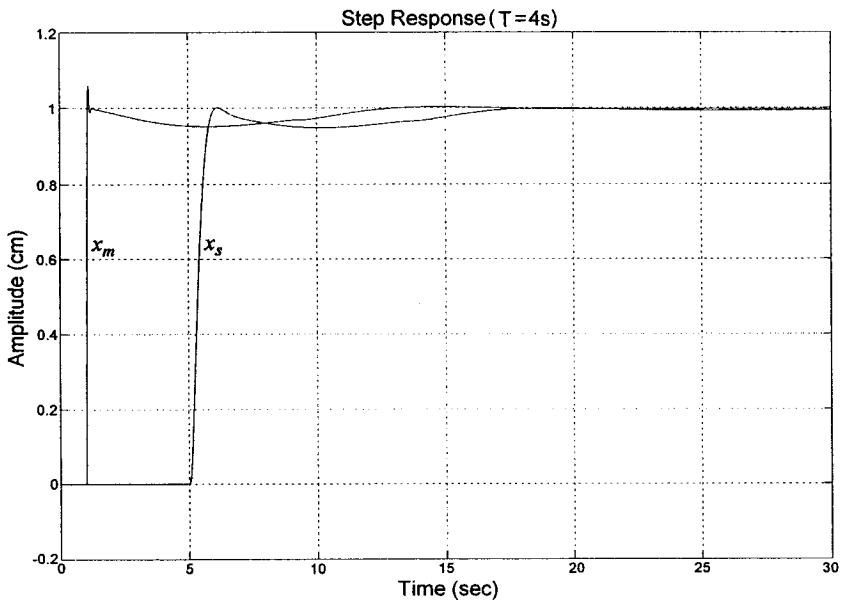


Fig. 3.24 Step Responses of x_m and x_s

More details about the performance analysis of the global closed-loop system with the resulting reduced-order controller will be given in Chapter 4.

Chapter 4

Analysis of the Resulting Telemanipulation Control System

In Chapter 3, the controller for the master manipulator, the controller for the slave manipulator, and the global controller for the case of interaction between the telemanipulation system and the environment have been designed with μ-synthesis method introduced in Chapter 2. In this chapter, analyses of the resulting proposed controllers are made.

The μ-analysis method used in this chapter will result in some relevant frequency plots, or μ-plots, in which a value below unity indicates achievement of the predefined design objectives. As described in Chapter 2, such frequency analysis plots will reveal

- Robust stability, an indicator which shows whether the system remains stable for a given set of perturbations to the nominal model;
- Nominal performance, which is used to indicate the compliance of a nominal, unperturbed system with the design requirements for its closed-loop performance; those requirements are implemented by the weighting functions which are incorporated in the interconnection structure of the system;
- Robust performance, which tests if the closed-loop system meets performance requirements and remains stable for the given set of perturbations to the nominal

model; the test is performed by a μ-value computation on the complete closed-loop system.

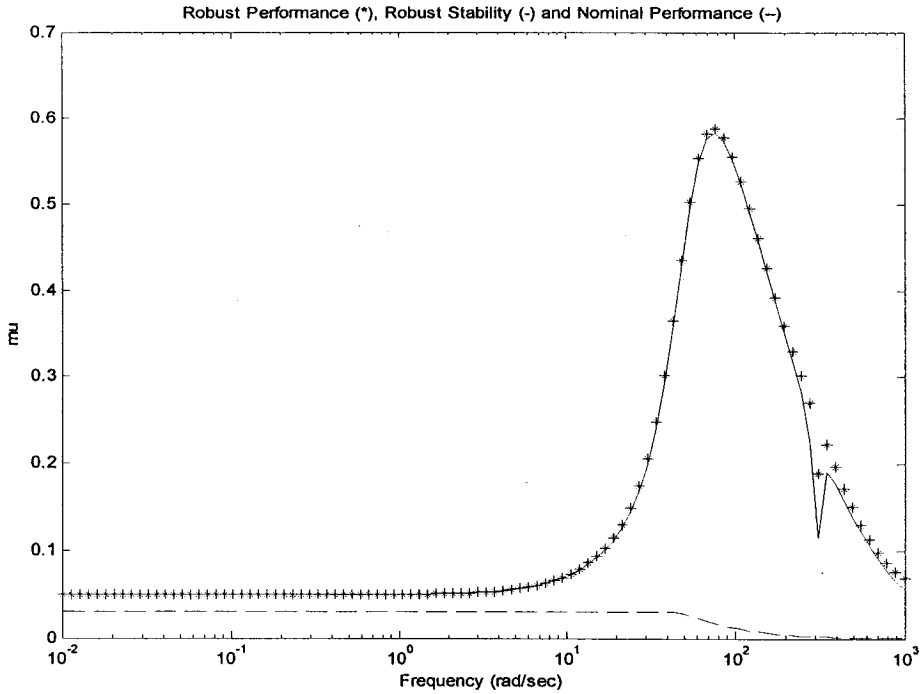
4.1 Analysis of the Master Control Subsystem

The analysis of the resulting master control subsystem is carried out both in the frequency domain and in the time domain.

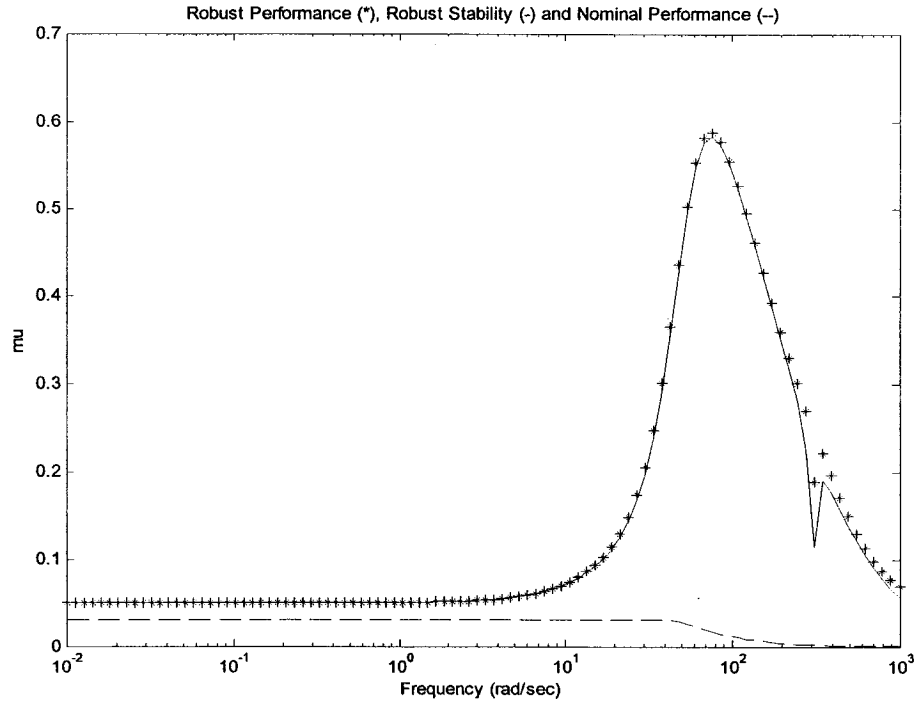
4.1.1 Analysis in the Frequency Domain

The design of the master controller is analyzed using the μ-theory frequency domain analysis tools. The interconnection structure of Figure 3.10 is first constructed, resulting in the generalized plant P_m in Figure 3.12(a). By closing the lower loop with the proposed master controller K_m results in the closed-loop master control subsystem as shown in Figure 3.12(b).

According to the μ-theory introduced in Section 2.5, Matlab code *Pmu.m*, is compiled in order to draw the robust stability, nominal performance and robust performance plots for this subsystem. After executing the *D-K* iteration program *dkitgui* and the controller order reduction and transformation subroutine *Kmr.m* or *K0.m*, running *Pmu.m* results in three plots that are depicted in Figures 4.1(a) and (b) for the full 10-th order controller and the reduced 7-th order controller, respectively.



(a) Full Order (10-th) Controller



(b) Reduced Order (7-th) Controller

Fig. 4.1 μ-analysis of the Master Subsystem

It is clear that with either the full order or the reduced-order controller, the robust stability is realized, which indicates that the controllers stabilize the complete sets of master manipulation subsystems.

Meanwhile, it can also be shown that for either controller with the weighting functions used and under the demanding set of requirements and large plant perturbations discussed in Chapter 3, the nominal performance and the robust performance over the complete frequency ranges are obtained. The performance is influenced if the uncertainty and the envelope perturbations are introduced, but the plot is still under unity, especially over the system bandwidth. This is indicated by the robust performance μ-plot which is larger than the nominal performance plot in Figure 4.1.

Note that there is no obvious change on the system robustness using the full order controller or the reduced order controller as can be seen by comparing the results in Figure 4.1(a) and Figure 4.1(b). This illustrates that the reduced-order controller can replace the full-order controller so that the control subsystem can be made more practical.

4.1.2 Analysis in the Time Domain

The step responses of the closed-loop master control subsystem with the reduced 7-th order K_m under three typical situations, $G_m = G_0$ (nominal model), $G_m = G_0(1+W_{mul})$, and $G_m = G_0(1-W_{mul})$, have been already shown in Figure 3.13, which illustrates that each of the three subsystems possesses good command-tracking ability and controller robustness.

In order to see how the closed-loop subsystem tracks the response of an ideal reference model, the step responses of both the ideal reference model and the nominal master control subsystem are plotted in Figure 4.2 by running the Matlab routine *Sridnom.m* given in Appendix A. The figure shows that the steady-state tracking is perfectly achieved by both models except for the reaction speed of the nominal subsystem being a little bit lower than the one of the ideal model.

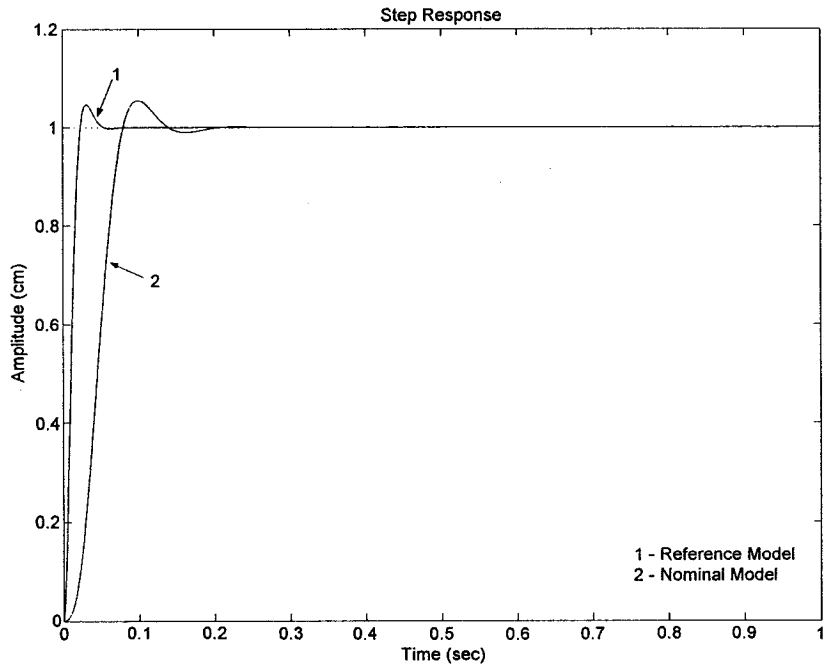


Fig. 4.2 The Ideal and the Nominal Step Response

4.2 Analysis of the Slave Control Subsystem

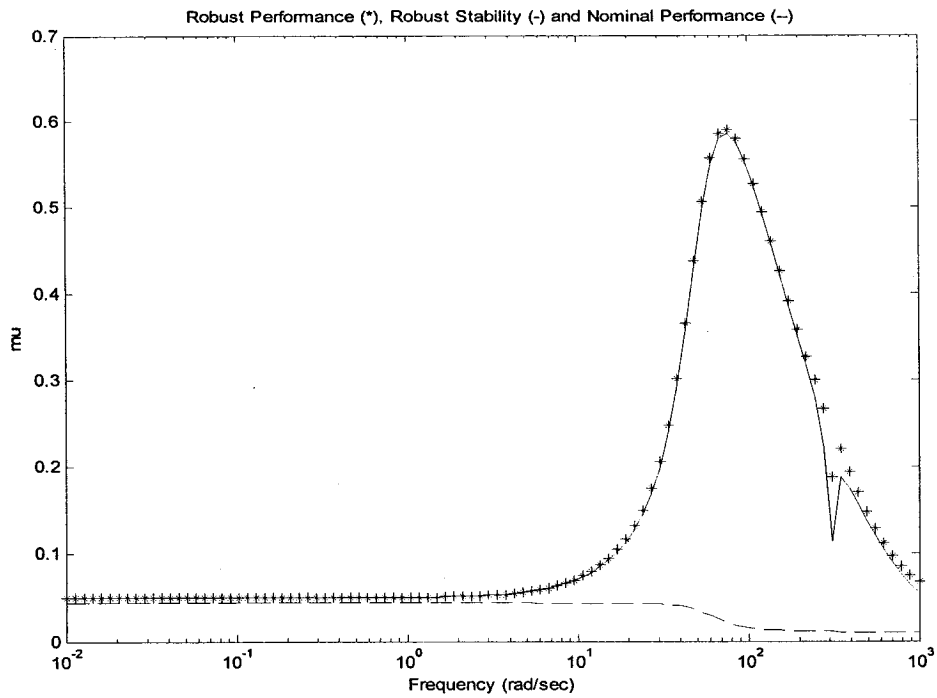
The analysis of the proposed slave controller as well as the closed-loop slave control subsystem can be made in a similar way that was described in Section 4.1.

4.2.1 Analysis in the Frequency Domain

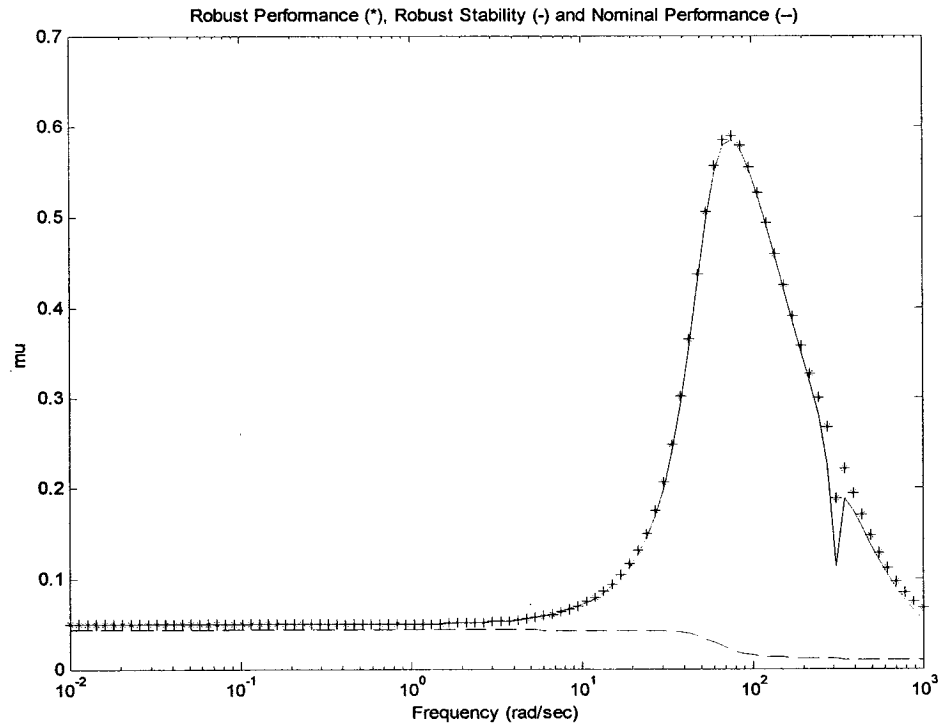
To carry out the μ-analysis on the slave controller, the interconnection structure of Figure 3.15 is firstly constructed, resulting in the generalized plant P_s shown in Figure 3.16(a). By closing the lower loop with the resulting master controller K_s yields the closed-loop master control subsystem as shown in Figure 3.16(b).

Code *Psu.m* is compiled and is run after executing *dkitgui* and *Kmr.m* or *K0.m*, to draw the μ-plots, which reflect robust stability, nominal performance and robust performance, for the slave control subsystem, respectively, with the full 8-th order controller and the reduced 7-th order controller. The plots are shown in Figures 4.3(a) and (b).

According to Figure 4.3, the proposed slave control subsystem with either the full 8-th order controller or the reduced 7-th order controller realizes robust stability, nominal performance and robust performance quite well since all the three plots remain beneath unity. Similar to the master control subsystem, the performance deteriorates if the model uncertainty is introduced, especially over the system bandwidth; however, the plot remains still under unity. Also, there is no noticeable change in the system robustness from using the full order controller or the reduced order controller. It can be shown that the 7-th order controller can replace the full 8-th order controller to obtain a simpler structure for realization.



(a) Full 8-th Order Controller



(b) Reduced 7-th Order Controller

Fig. 4. 3 μ-Analysis of the Slave Subsystem

4.2.2 Analysis in the Time Domain

The step responses of the closed-loop slave control subsystem with the reduced 7-th order K_s under three typical situations, $G_s = G_0$ (nominal model), $G_s = G_0(1+W_{mul})$, and $G_s = G_0(1-W_{mul})$, have been already shown in Figure 3.18, which illustrates that each of the three subsystems possesses good command-tracking ability and the controller robustness. Since the ideal reference model for the slave manipulator is taken as $G_{sr}(s) = 1$ in Chapter 3, for the closed-loop slave subsystem, tracking the ideal response is the same as tracking the command input. Therefore, Figure 3.18 reflects truly the tracking ability of the closed-loop slave subsystem.

4.3 Analysis of the Global Controller – the Constrained Case

In this section, the global controller for the constrained movement is analyzed in both frequency domain and time domain.

4.3.1 Analysis in the Frequency Domain

To carry out μ-analysis on the reduced-order global controller K_g for the constrained movement case which is constructed in Section 3.3.2 and represented by equations (3.37) and (3.38), the interconnection structure of Figure 3.21 is first constructed, resulting in the generalized plant P_g that is shown in Figure 3.22(a). By closing the lower loop with K_g, the closed-loop system shown in Figure 3.22(b) is obtained.

The code *Pgu.m* in Appendix A is compiled and is run after executing *dkitgui* and *Kgr.m* to draw the μ-plots which reflect robust stability, nominal performance and robust performance, for the global control system with the reduced 9-th order global controller. The plots are shown in Figure 4.4.

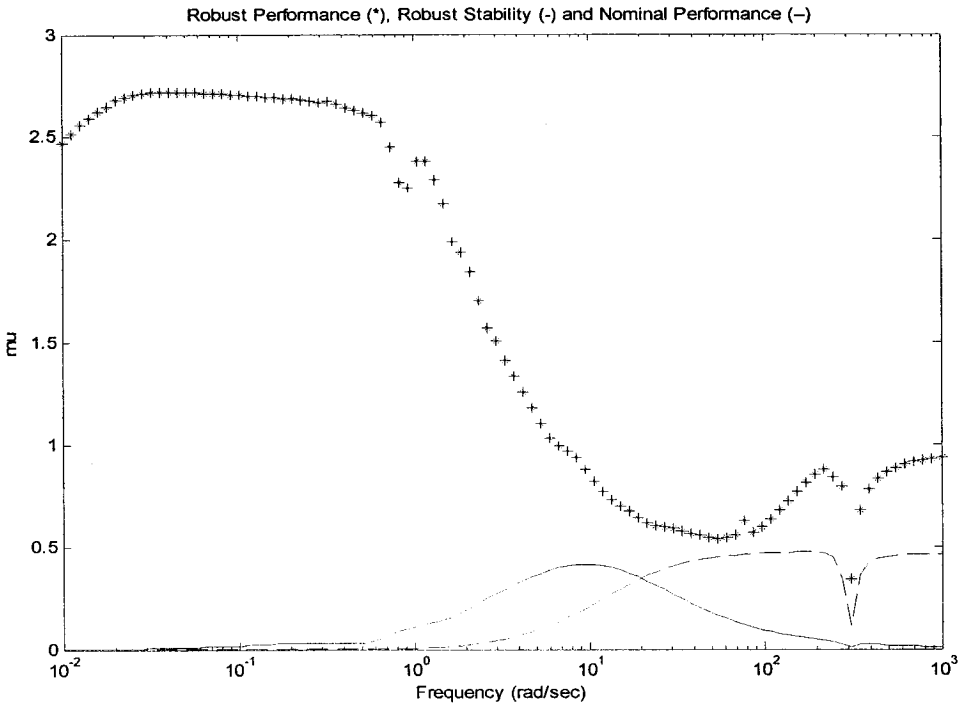


Fig. 4.4 μ-Analysis of the Closed-Loop System with Reduced Controller K_g

From Figure 4.4, it can be seen that the system with the reduced controller achieves the nominal performance and the robust stability although the robust performance is not achieved in the whole frequency range (0.01~1000 rad/s). Indeed, the performance of the closed-loop system with reduced order controller (9-th order instead of 129-th order) shows good robustness since the μ-plots of the robust performance, the robust stability and the nominal performance of the closed-loop system with this controller are all below unity when $\omega > 5$ rad/sec, which is even better than the one of the system with the full

129-th order controller comparing Figure 4.4 with Figure 3.23 where the μ-plot is still above unity at some frequencies higher than 100 rad/sec.

Although some of the μ-values in Figure 4.4 are larger than 1, indicating that the design objectives are not fully achieved, from the time domain simulation results shown in the next section it can be stated that the properties of the controller are satisfactory. This may indicate that some of the weighting functions might not reflect the actual requirements to the full extent and should be retuned. It might also indicate that the design method is conservative or that it might be impossible to take all the actual requirements into account without making the design too conservative.

4.3.2 Analysis in the Time Domain

The closed-loop global system corresponding to the reduced-order global controller K_g for the constrained case is simulated under following typical model cases.

4.3.2.1 Case Nominal System : $G_m = G_s = G_0$, $G_e = B_0 s + K_0$ and T = 4 sec

In this case, the system simulation layout is shown in Figure 4.5 and the simulation results for the closed-loop telemanipulation control system are shown in Figures 4.6 to Figure 4.9.

Figure 4.6 shows that both the master manipulator displacement x_m and the slave manipulator displacement x_s of the nominal system in the constrained movement case

can well follow the step input command f_h which is exerted at t = 1 second. Figure 4.7 illustrates that x_s tracks x_m very well except for a delay caused by the communication channel.

Figures 4.8 and 4.9 illustrate how well the transparency of the system is reached. They show that the output of the global controller, i.e., the reaction force felt at the master side, $\tilde{f}_e$, is almost the same as the real environment reaction force at the remote site, f_e, when ignoring the delay caused by the communication channel.

These simulation results illustrate that all the performance objectives described in Chapter 3 for the constrained case have been reached by the nominal system with the resulting reduced-order controllers.

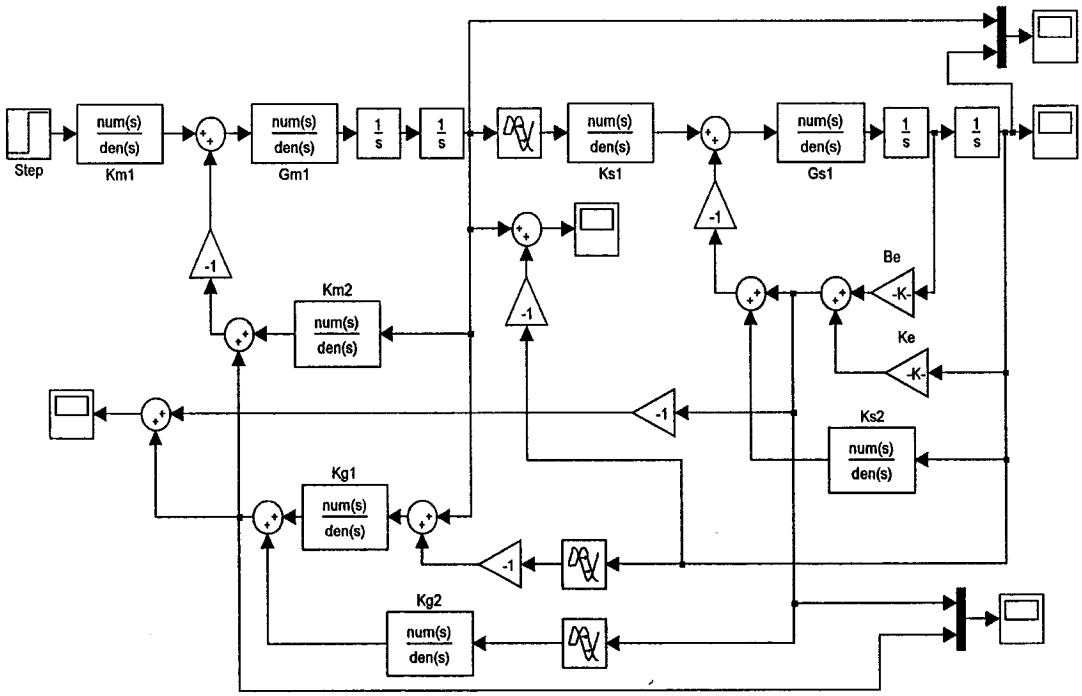


Fig.4 .5 Simulation Block Diagram Schematic: Nominal System

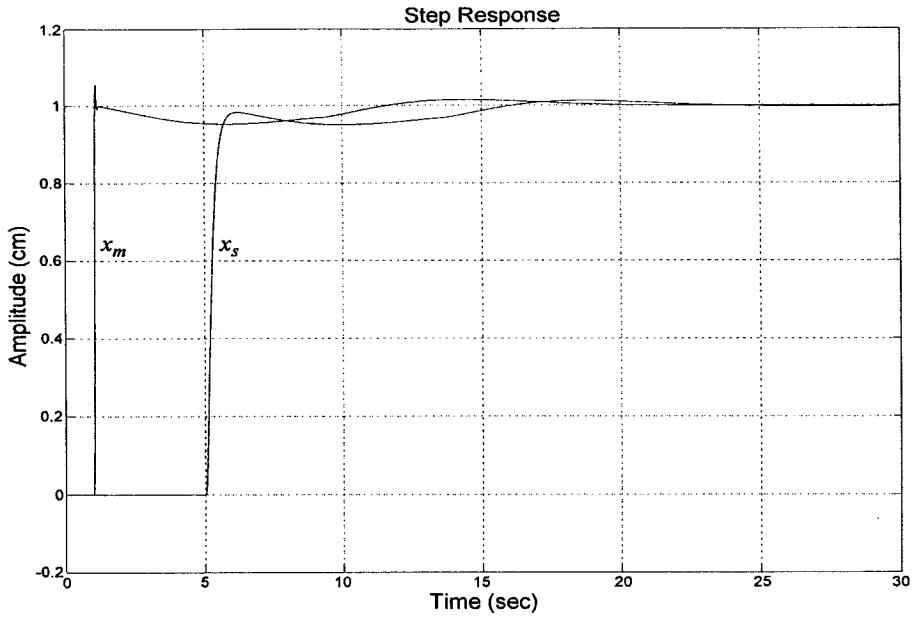


Fig. 4.6 Step Responses of x_m and x_s with $G_m = G_s = G_0, G_e = B_0 s + K_0$ and $T = 4$ sec

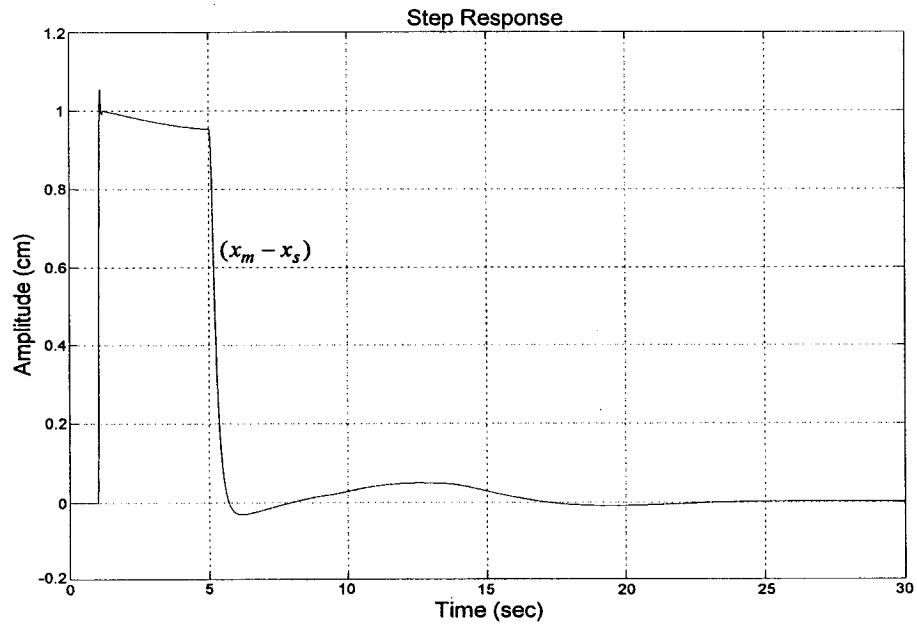


Fig. 4.7 Step Response of (x_m - x_s) with $G_m = G_s = G_0, G_e = B_0 s + K_0$ and $T = 4$ sec

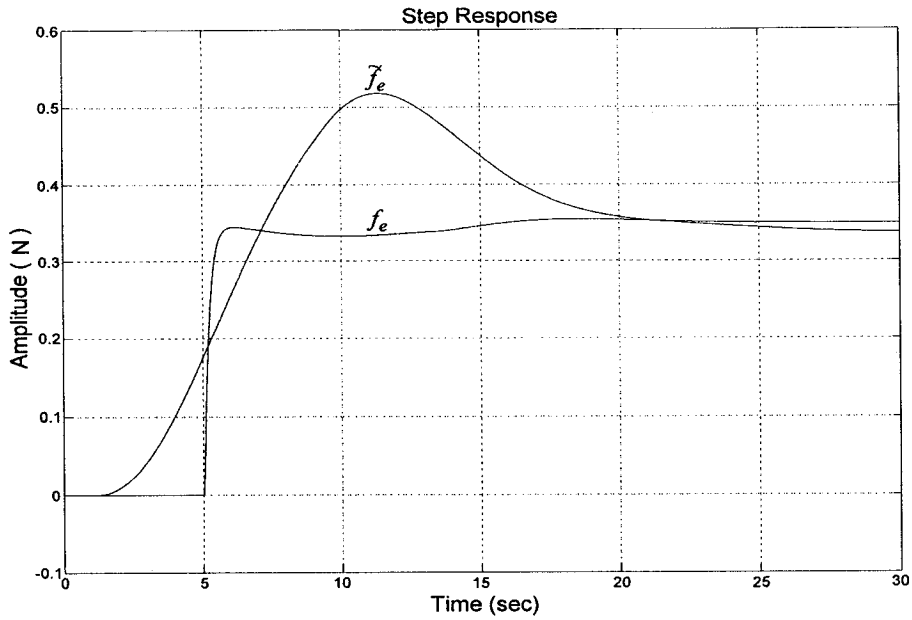


Fig. 4.8 Step Response of f_e and $\tilde{f}_e$ with $G_m = G_s = G_0, G_e = B_0 s + K_0$ and $T = 4$ sec

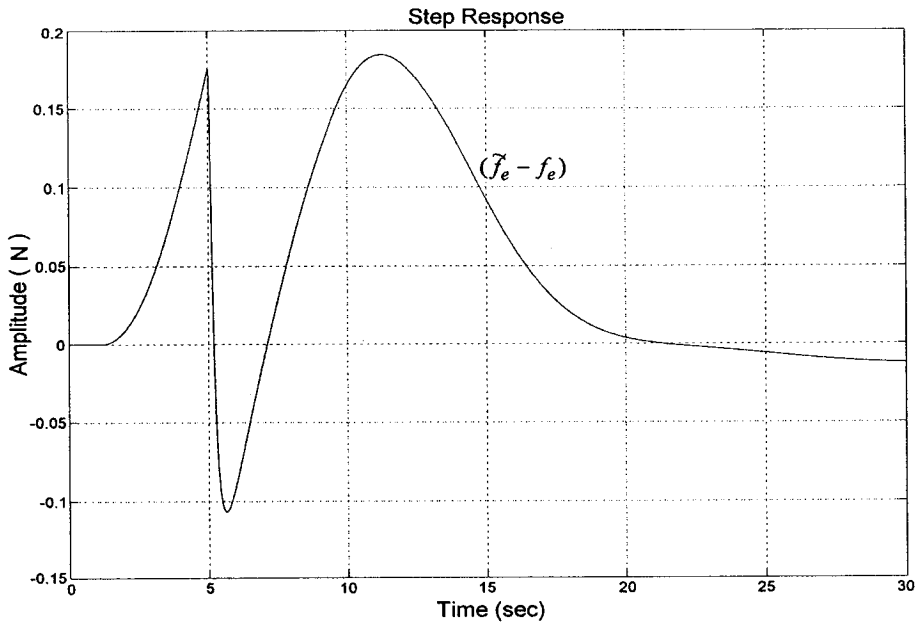


Fig. 4.9 Step Response of ($\tilde{f}_e$ - f_e) with $G_m = G_s = G_0, G_e = B_0 s + K_0$ and $T = 4$ sec

4.3.2.2 Case $G_m = G_s = G_0(1+W_{mul})$, $G_e = B_0(1+db)s + K_0(1+dk)$ and T = 4 sec

In this case, the simulation layout of the closed-loop telemanipulation control system is shown in Figure 4.10 and the simulation results are shown in Figures 4.11 to 4.14.

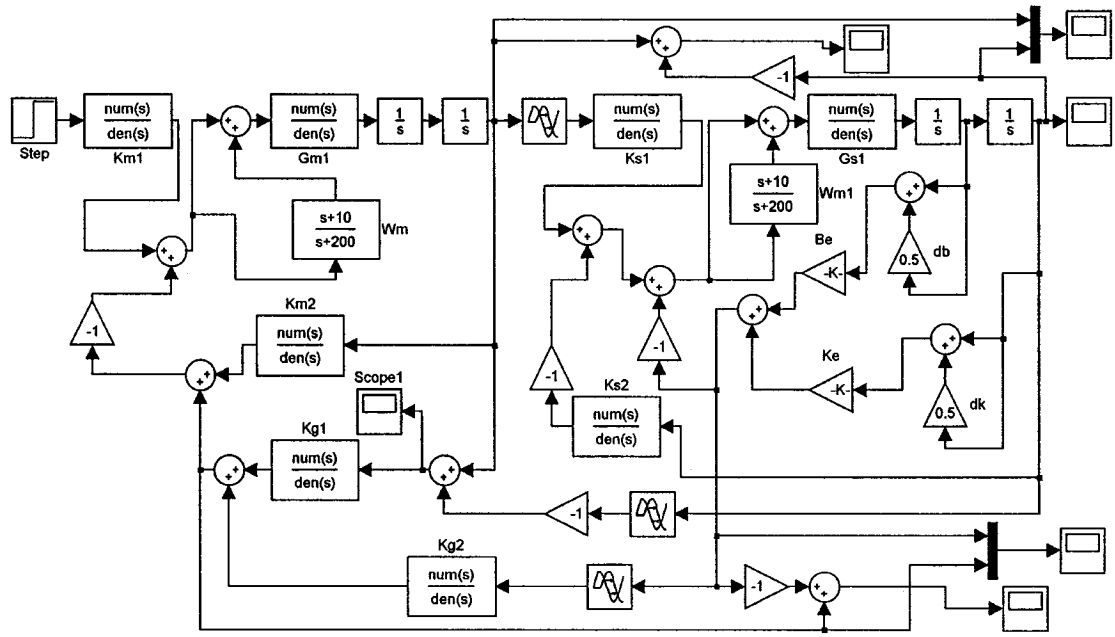


Fig. 4.10 Simulation Block Diagram Schematic: $T = 4$ sec and
$G_m = G_s = G_0(1+W_{mul}), G_e = B_0(1+db)s + K_0(1+dk)$

Figure 4.11 shows that for the plants $G_m = G_s = G_0(1+W_{mul})$ and $G_e = B_0(1+db)s + K_0(1+dk)$ in the constrained case, both the master manipulator displacement x_m and the slave manipulator displacement x_s can follow the step input command f_h quite well which is exerted at t = 1 second. Figure 4.12 indicates that x_s tracks x_m very well except for a delay caused by the communication channel.

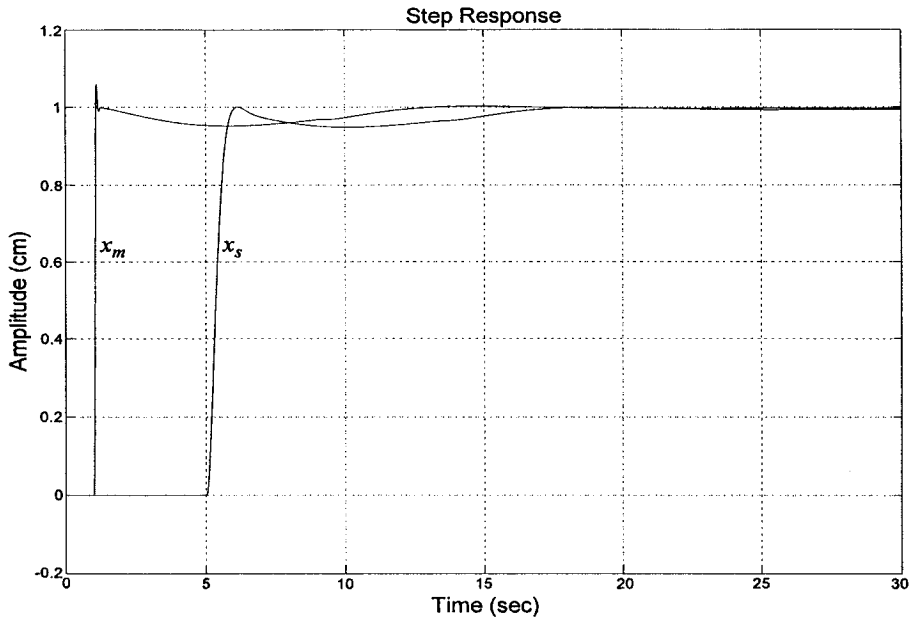


Fig. 4.11 Step Responses of x_m and x_s When $T = 4$ sec and

$$G_m = G_s = G_0(1 + W_{mul}),\ G_e = B_0(1 + db)s + K_0(1 + dk)$$

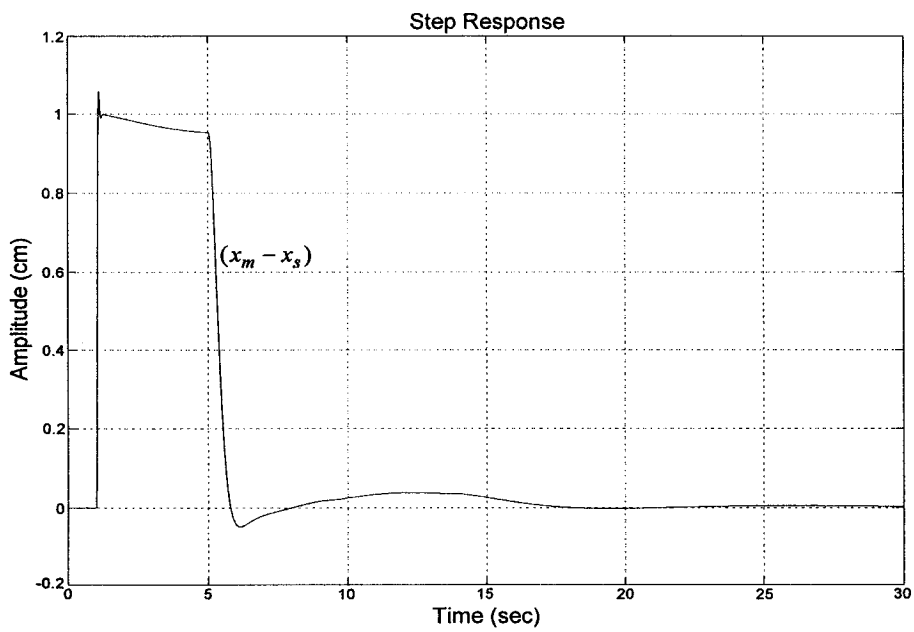


Fig. 4.12 Step Response of (x_m - x_s) When $T = 4$ sec and

$$G_m = G_s = G_0(1 + W_{mul}),\ G_e = B_0(1 + db)s + K_0(1 + dk)$$

Figures 4.13 and 4.14 illustrate how well the transparency of the system possessing largest uncertainties has been reached. They show that $\tilde{f}_e$, the reaction force felt at the master side, is almost the same as f_e, the real environment reaction force at the remote site, when ignoring the delay caused by the communication channel.

Comparing Figure 4.6 with Figure 4.11 and Figure 4.7 with Figure 4.12, it is not difficult to see that there is no noticeable change in the responses of the displacement x_m and x_s when the system switches from nominal case to the case of largest uncertainties.

However, there is a change in the step response of f_e and $\tilde{f}_e$ when this switching takes place as seen by comparing Figure 4.8 with Figure 4.13. Indeed, the amplitude of the reaction force of the environment at steady state is increased from approximately 0.35 to approximately 0.53. This is due to the fact that the coefficients of the environment have been changed by 50%, which results in the change of the steady-state value of the reaction force.

Note that x_s reaches its steady-state value $X_s = 1$, and with $K_0 = 0.35$, $dk = 0.5$, which are specified in Chapter 3, the steady-state value of f_e is equal to

$$F_e = K_0(1+dk)X_s = 0.35 \times (1+0.5) \times 1 = 0.525.$$

which illustrates that the system possesses good transparency and robustness.

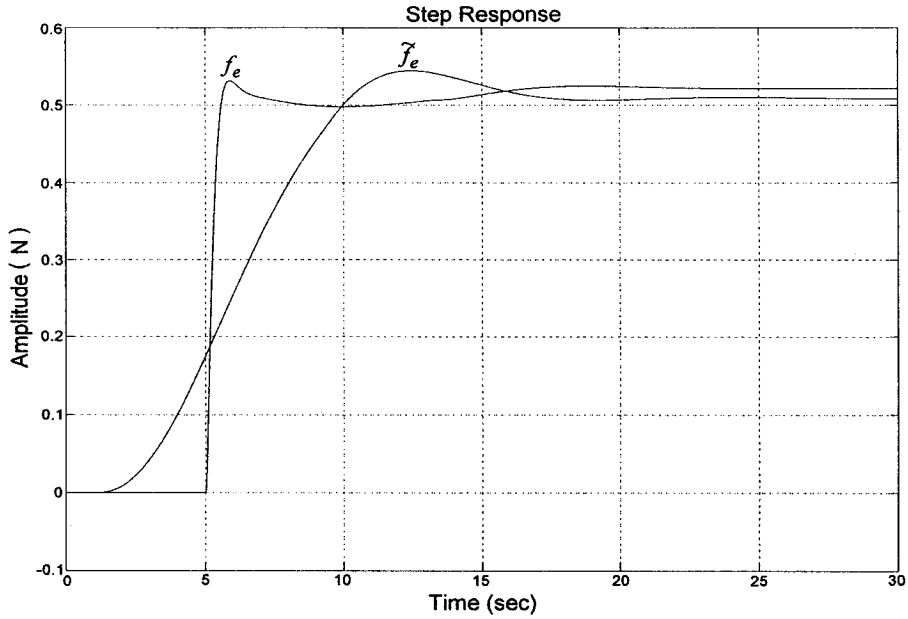


Fig. 4.13 Step Responses of f_e and $\tilde{f}_e$ When $T = 4$ sec and

$$G_m = G_s = G_0(1 + W_{mul}),\ G_e = B_0(1 + db)s + K_0(1 + dk)$$

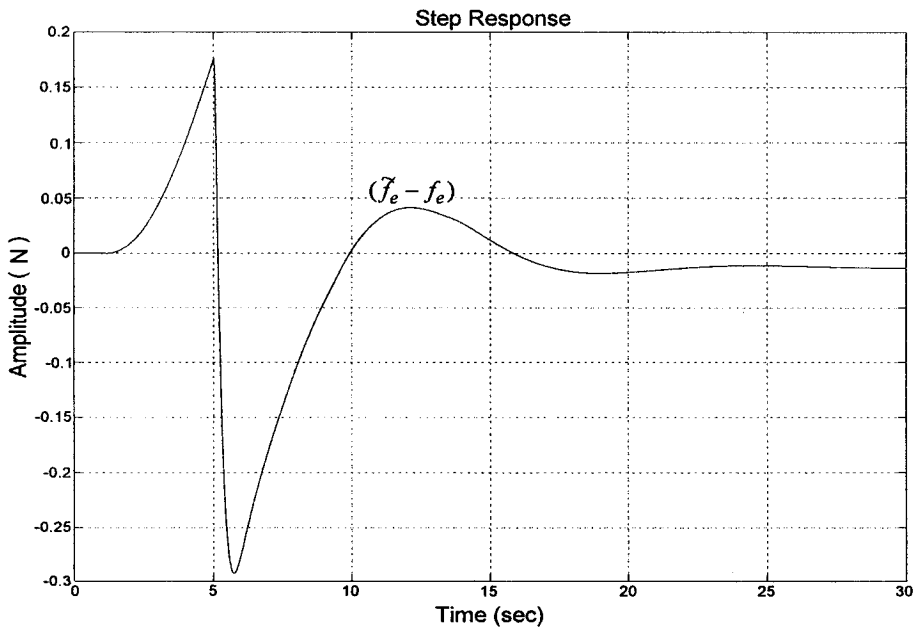


Fig. 4.14 Step Response of ($\tilde{f}_e$ - f_e) When $T = 4$ sec and

$$G_m = G_s = G_0(1 + W_{mul}),\ G_e = B_0(1 + db)s + K_0(1 + dk)$$

4.3.2.3 Case $G_m = G_s = G_0(1-W_{mul})$, $G_e = B_0(1-db)s + K_0(1-dk)$ and T = 4 sec

In this typical case, the system schematic simulation layout is shown in Figure 4.15 and the simulation results are shown in Figure 4.16 to Figure 4.19.

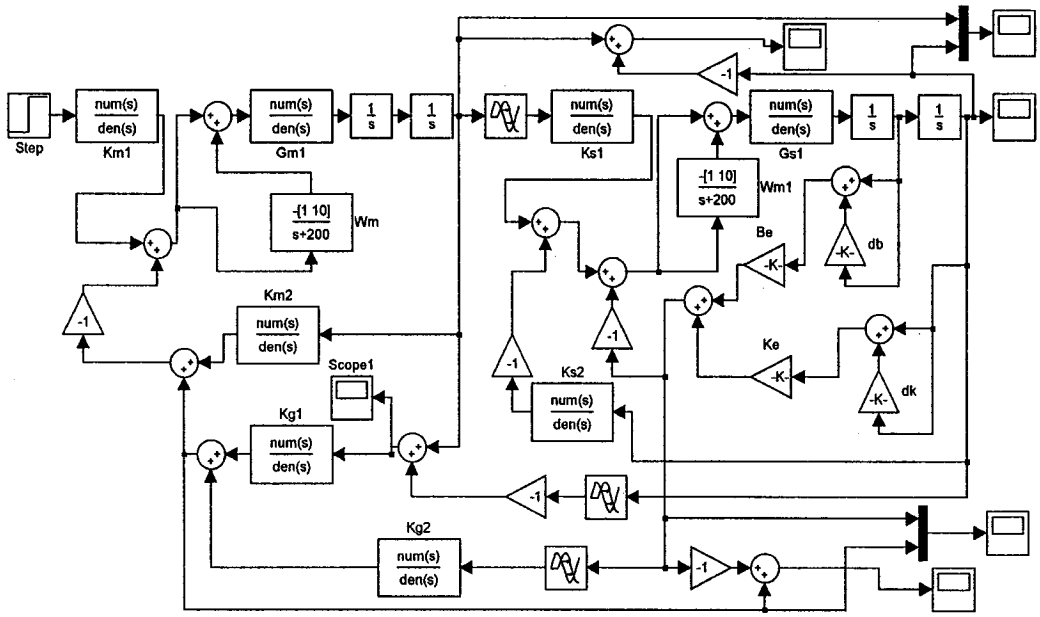


Fig. 4 .15 Simulation Block Diagram Schematic: T = 4 sec and $G_m = G_s = G_0(1-W_{mul})$, $G_e = B_0(1-db)s + K_0(1-dk)$

Figure 4.16 shows that for the plants $G_m = G_s = G_0(1-W_{mul})$ and $G_e = B_0(1-db)s + K_0(1-dk)$ in the constrained movement case, both the master manipulator displacement x_m and the slave manipulator displacement x_s can follow the step input command f_h which is exerted at t = 1 second quite well, while Figure 4.17 indicates that x_s tracks x_m quite well except for a delay caused by the communication channel.

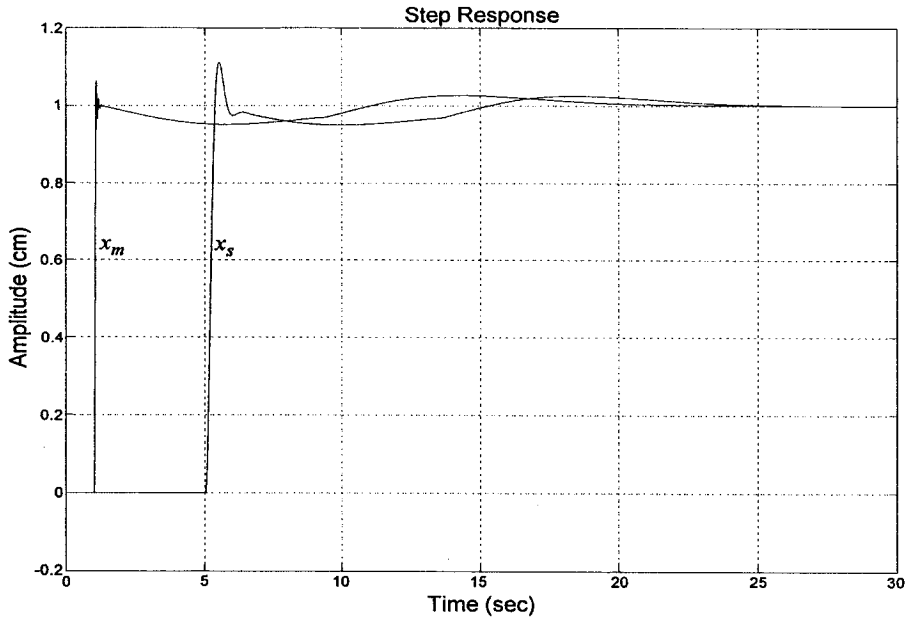


Fig. 4.16 Step Responses of x_m and x_s When $T = 4$ sec and

$$G_m = G_s = G_0(1 - W_{mul}),\ G_e = B_0(1 - db)s + K_0(1 - dk)$$

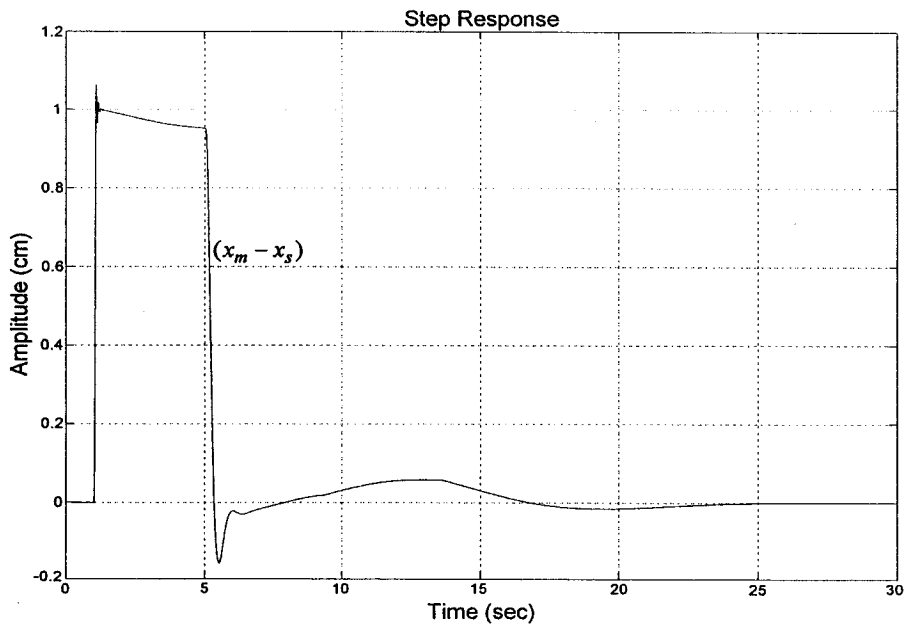


Fig. 4.17 Step Response of (x_m - x_s) When $T = 4$ sec and

$$G_m = G_s = G_0(1 - W_{mul}),\ G_e = B_0(1 - db)s + K_0(1 - dk)$$

There is again no noticeable change in the responses of the displacement x_m and x_s when the system switches from nominal case to the other case of largest uncertainties.

Figures 4.18 and 4.19 illustrate the transparency of the system in this case. It shows that, at steady state, the reaction force felt at the master side, $\tilde{f}_e$, is almost the same as the real environment reaction force f_e at the remote site if one ignores the delay caused by the communication channel and the large overshoot of $\tilde{f}_e$.

The amplitude of the reaction force of the environment at steady state is decreased from 0.35 in Section 4.3.2.1 or 0.525 in Section 4.3.2.2 to approximately 0.18. This is due to the fact that the coefficients of the environment are 50% less than the nominal case (Section 4.3.2.1).

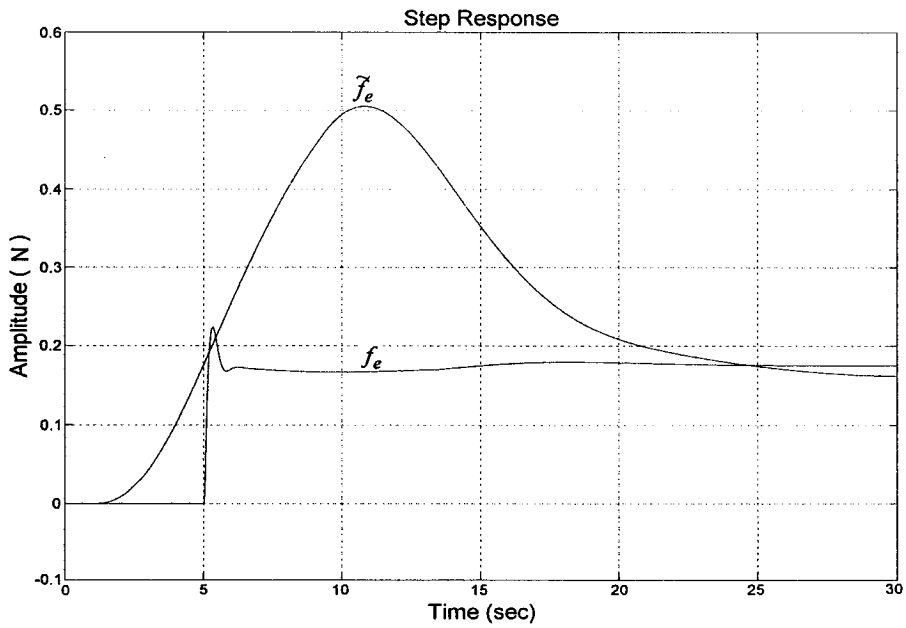


Fig. 4.18 Step Responses of f_e and $\tilde{f}_e$ When $T = 4$ sec and

$$G_m = G_s = G_0(1 - W_{mul}),\ G_e = B_0(1 - db)s + K_0(1 - dk)$$

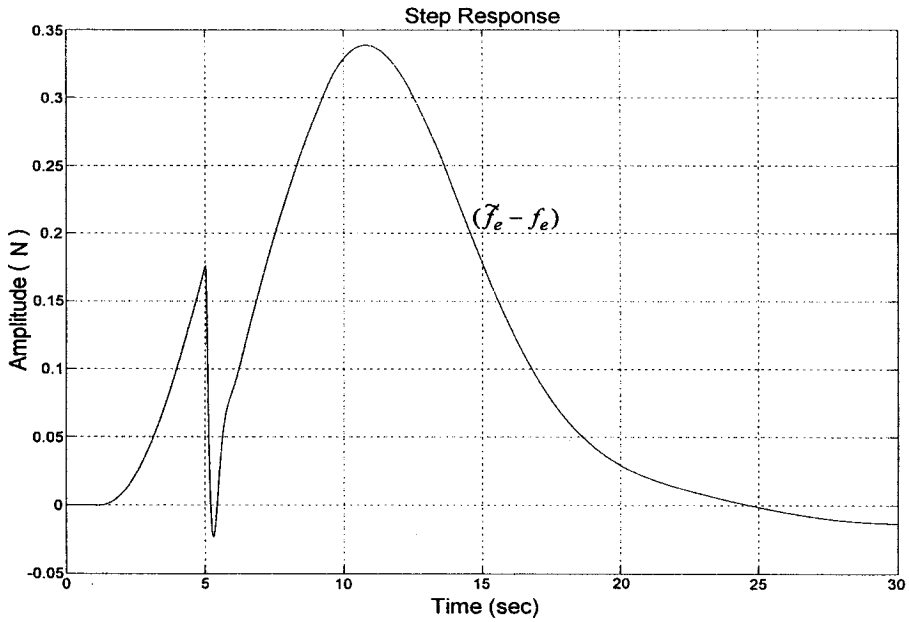


Fig. 4.19 Step Response of ($\tilde{f}_e$ - f_e) When T = 4 sec and

$$G_m = G_s = G_0(1 - W_{mul}),\ G_e = B_0(1 - db)s + K_0(1 - dk)$$

4.3.2.4 Other Scenarios

Similar results can be obtained from simulations in other typical cases as follows:

1) $G_m = G_0(1+W_{mul})$, $G_s = G_0(1+W_{mul})$ and $G_e = B_0(1+db)s + K_0(1-dk)$;

2) $G_m = G_0(1+W_{mul})$, $G_s = G_0(1+W_{mul})$ and $G_e = B_0(1-db)s + K_0(1+dk)$;

3) $G_m = G_0(1+W_{mul})$, $G_s = G_0(1+W_{mul})$ and $G_e = B_0(1-db)s + K_0(1-dk)$;

4) $G_m = G_0(1+W_{mul})$, $G_s = G_0(1-W_{mul})$ and $G_e = B_0(1+db)s + K_0(1+dk)$;

5) $G_m = G_0(1+W_{mul})$, $G_s = G_0(1-W_{mul})$ and $G_e = B_0(1+db)s + K_0(1-dk)$;

6) $G_m = G_0(1+W_{mul})$, $G_s = G_0(1-W_{mul})$ and $G_e = B_0(1-db)s + K_0(1+dk)$;

7) $G_m = G_0(1+W_{mul})$, $G_s = G_0(1-W_{mul})$ and $G_e = B_0(1-db)s + K_0(1-dk)$;

8) $G_m = G_0(1-W_{mul})$, $G_s = G_0(1+W_{mul})$ and $G_e = B_0(1+db)s + K_0(1+dk)$;

9) $G_m = G_0(1-W_{mul})$, $G_s = G_0(1+W_{mul})$ and $G_e = B_0(1+db)s + K_0(1-dk)$;

10) $G_m = G_0(1-W_{mul})$, $G_s = G_0(1+W_{mul})$ and $G_e = B_0(1-db)s + K_0(1+dk)$;

11) $G_m = G_0(1-W_{mul})$, $G_s = G_0(1+W_{mul})$ and $G_e = B_0(1-db)s + K_0(1-dk)$;

12) $G_m = G_0(1-W_{mul})$, $G_s = G_0(1-W_{mul})$ and $G_e = B_0(1+db)s + K_0(1+dk)$.

13) $G_m = G_0(1-W_{mul})$, $G_s = G_0(1-W_{mul})$ and $G_e = B_0(1+db)s + K_0(1-dk)$;

14) $G_m = G_0(1-W_{mul})$, $G_s = G_0(1-W_{mul})$ and $G_e = B_0(1-db)s + K_0(1+dk)$.

The simulation results for the above conditions, which are included in Appendix B, show that the bilateral haptic telemanipulation control system configured in Chapter 3 possesses very good robustness.

4.4 Analysis of the Effects of the Communication Time Delay

In the previous sections, simulation results for some typical system configurations have been analyzed. In this section, the influences of the uncertain transmission time-delay on the performance of the bilateral haptic telemanipulation system are discussed.

Time delay, caused by the communication channel, has received much attention by researchers as it may influence the stability and performance of a bilateral telemanipulation system.

4.4.1 The Free Movement Case

In this case, the time delay of the communication channel has no influence on the system robust stability, nominal performance and robust performance, no matter how large or small it is. The following simulation results show this.

The schematic simulation layouts for the nominal system with $G_m = G_s = G_0$ and the system with $G_m = G_s = G_0(1 + W_{mul})$ are shown in Figures 4.20 and 4.21, respectively.

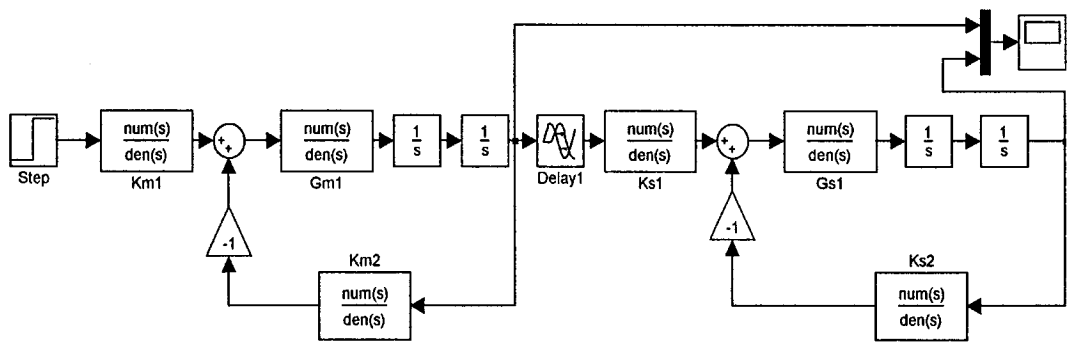


Fig. 4. 20 Simulation Schematic for the Nominal System: Free Movement

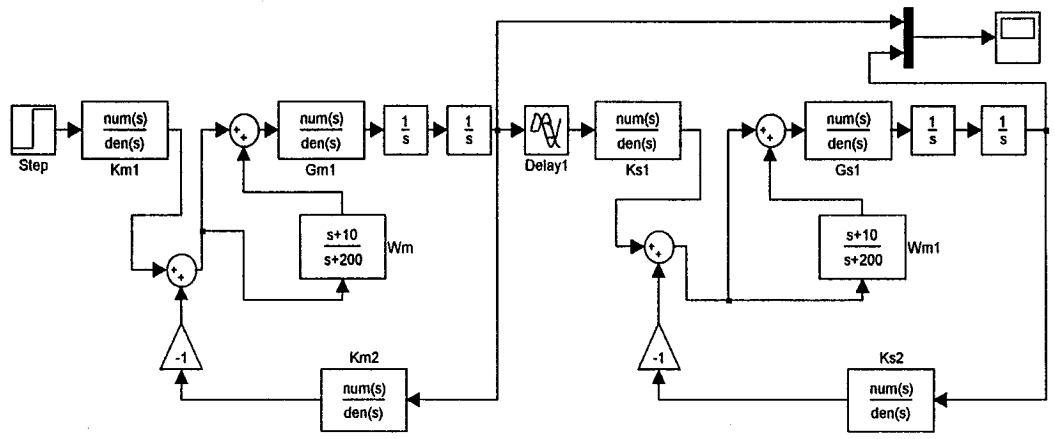
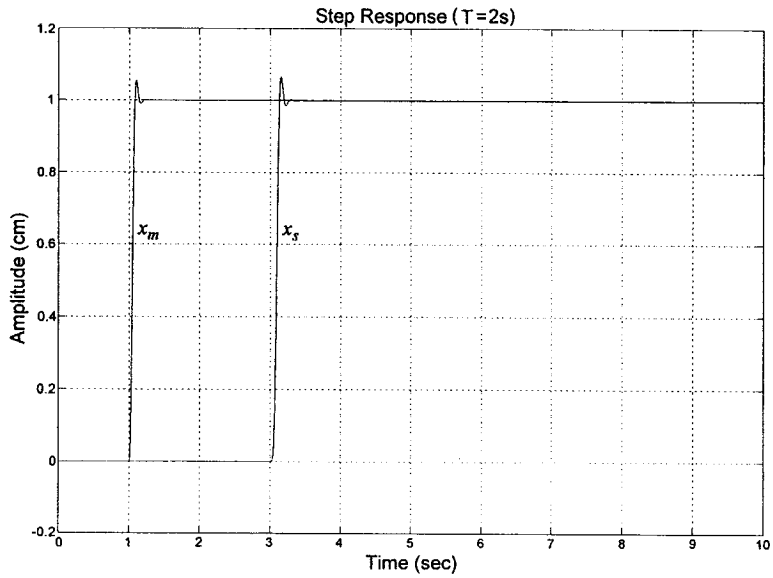
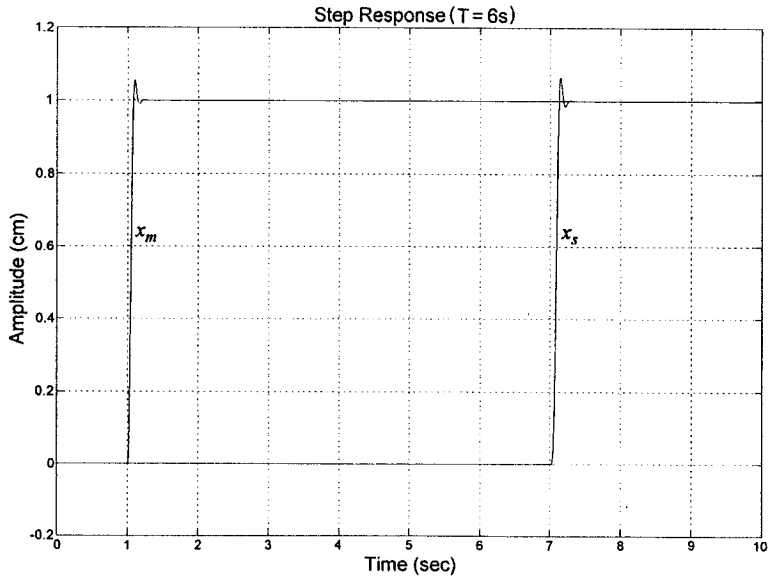


Fig. 4. 21 Simulation Schematic for the System with $G_m = G_s = G_0(1 + W_{mul})$: Free Movement

Figure 4.22 shows the step response of the nominal system with $G_m = G_s = G_0$ for the time delay of T = 2 and 6 seconds respectively, while Figure 4.23 presents the step response of the system with $G_m = G_s = G_0(1 + W_{mul})$ for the time delay of T = 2 and 6 seconds respectively.

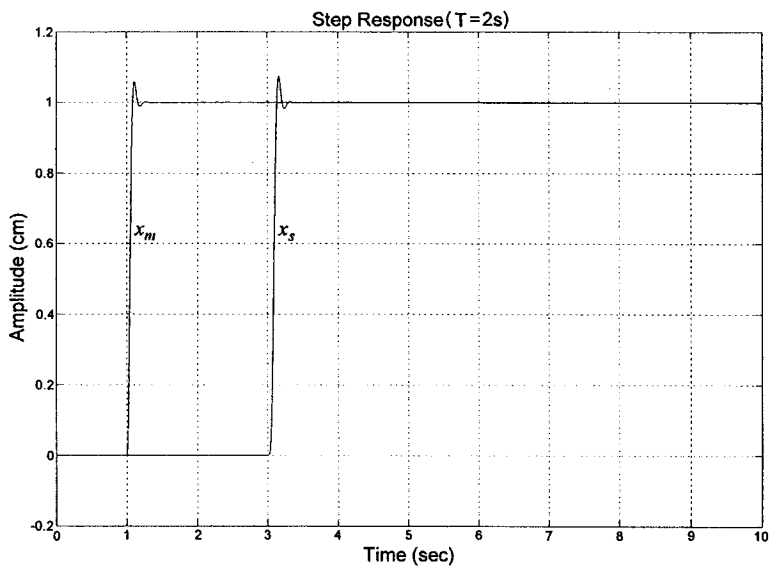


(a) T = 2 sec

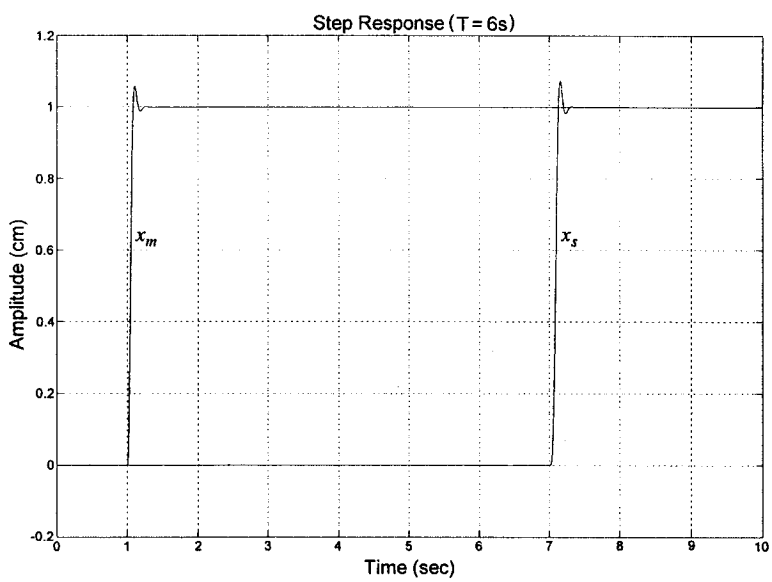


(b) T = 6 sec

Fig. 4.22 Step response of the nominal system



(a) T = 2 sec



(b) T = 6 sec

Fig. 4.23 Step responses of the system with $G_m = G_s = G_0(1 + W_{mul})$

4.4.2 The Constrained Movement Case

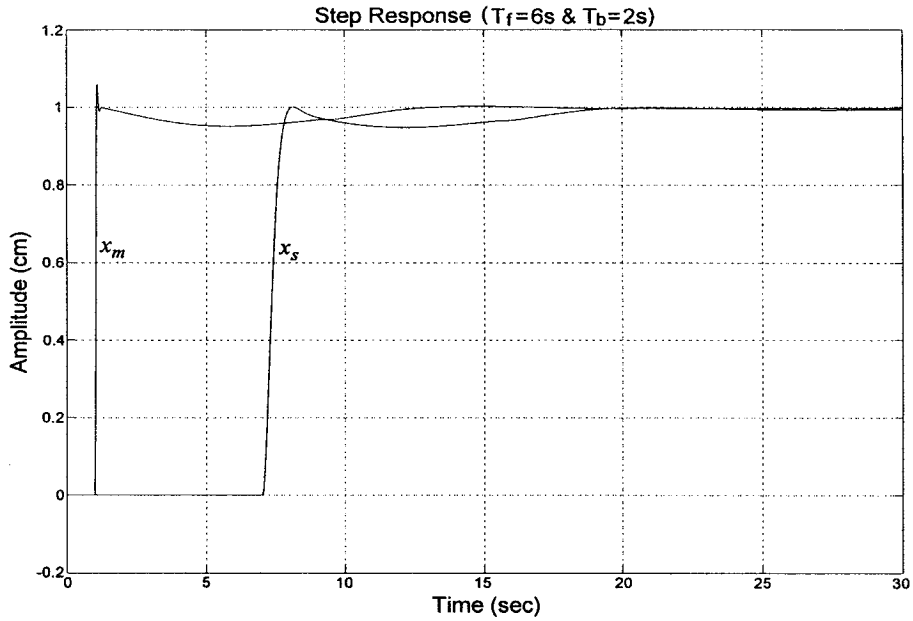
In this case, the telemanipulator interacts with an environment. Two situations are considered here: the first one is when the time delay in the forward path (i.e., from the master to the slave) is larger than the time delay in the backward path (i.e., from the slave to the master), and the second one is the opposite situation.

Based on the μ-controllers constructed in Chapter 3, the simulation results obtained for the whole envelope of systems are similar to each other. Therefore, only the system with $G_m = G_s = G_0(1+W_{mul})$ and $G_e = B_0(1+db)s + K_0(1+dk)$, where Δ_{mul}, Δ_b and Δ_k take their maximum value of 1, is considered here. In other words, Figure 4.10 is adopted again.

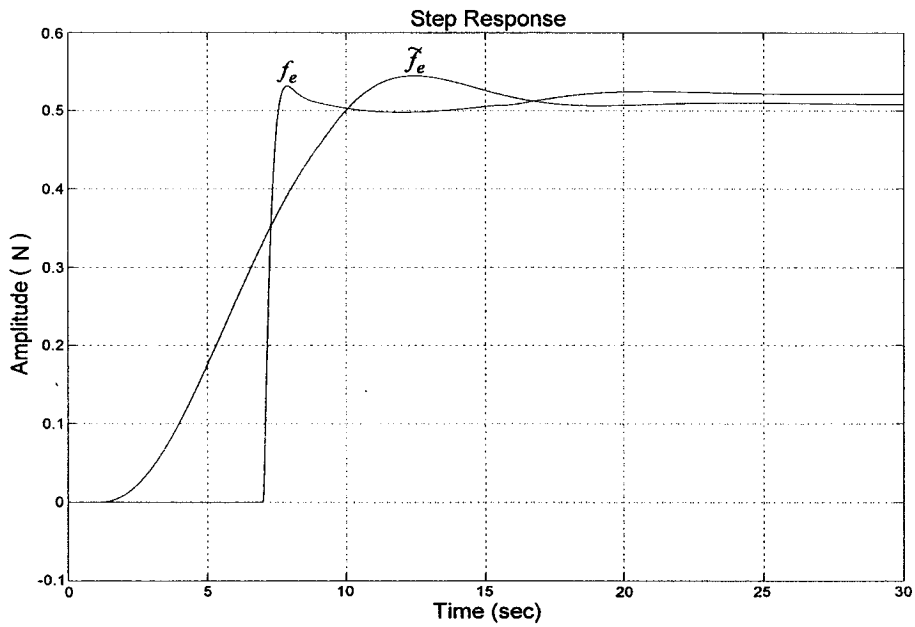
Figure 4.24 and Figure 4.25 show the step responses of the system considered in two situations, respectively:

1) the forward time delay $T_f = 6$ seconds and the backward time delay $T_b = 2$ seconds (Figure 4.24), and
2) the forward time delay $T_f = 2$ seconds and the backward time delay $T_b = 6$ second (Figure 4.25).

The simulation results illustrate that the time delay of the communication channel in both situations has also no influence on the system stability and performance.



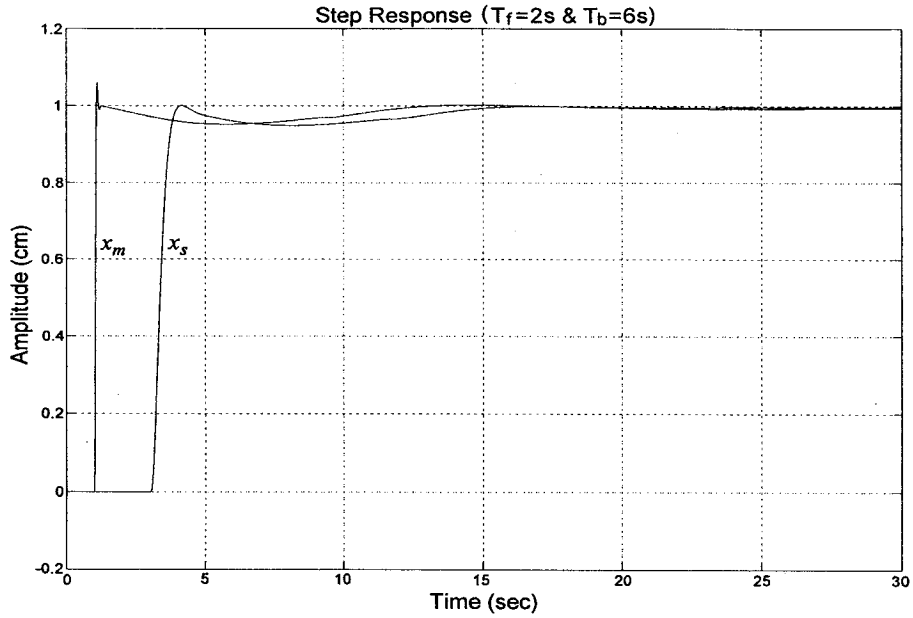
(a) Step Responses of x_m and x_s



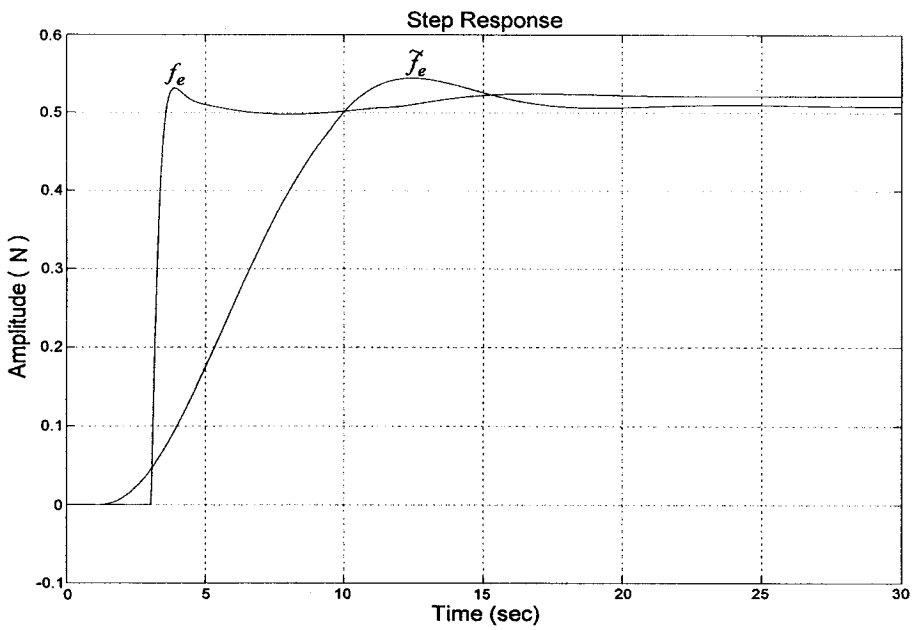
(b) Step Responses of f_e and $\tilde{f}_e$

Fig. 4.24 Step Responses of the System: $T_f = 6$ s & $T_b = 2$ s

$$G_m = G_s = G_0(1 + W_{mul}),\ G_e = B_0(1 + db)s + K_0(1 + dk)$$



(a) Step Responses of x_m and x_s



(b) Step Responses of f_e and $\tilde{f}_e$

Fig. 4.25 Step Responses of the System: $T_f = 2$ s & $T_b = 6$ s

$$G_m = G_s = G_0(1 + W_{mul}),\ G_e = B_0(1 + db)s + K_0(1 + dk)$$

Chapter 5

Conclusions and Future Research Directions

The structured singular value, μ, can be applied to evaluate robust stability and performance robustness of a controlled system or plant in face of uncertainties. Any linear controller which is designed for a system can be connected with the generalized plant to form the closed-loop system and can therefore to be analyzed.

In this chapter, some conclusions on the research work presented in this thesis are made and some future research problems are proposed.

5.1 Conclusions on the Presented Framework

In this thesis, a design framework using LFT technique and μ-synthesis and analysis theories for a bilateral haptic telemanipulation control system was presented. The results were accompanied and substantiated with frequency domain analysis and time domain simulations, both of which show the proposed controllers yield satisfactory performance requirement.

The first feature of this design framework is that every component in the considered system is assumed to possess uncertainties of multiplicative dynamic or parametric in

nature. All the uncertainties are simultaneously accounted for during the design process such that the resulting control system can be ensured to be more robust.

The thesis has shown how all the plants (i.e., the master and the slave manipulators, the communication channel and the environment), controllers, modeled uncertainties, and various performance weights are incorporated in a single framework that is of a general interconnection structure and equipped with a suitable matrix measure (the structured singular value, μ). It has also illustrated that this framework can easily handle all system performance specifications simultaneously. The robust stability of the system is guaranteed subject to modeling uncertainties and bounded perturbations, and the performance of the proposed controllers are satisfactory in terms of tracking responses although robust performance is not fully met in the whole frequency range for the constrained movement case, which implies that either the performance and control weights need further tuning or the ranges of the uncertainties predefined for the plants in the considered system are too broad to be possible to make it acquiring the robust performance.

Input multiplicative uncertainty models for both the master and the slave manipulators, and parametric uncertain models for both the communication channel and the environment, are considered to sufficiently cover variations in the models of these plants.

Much effort has been put in choosing weighting functions in order to achieve robust stability and good performance. Right choice of the weighting functions which reflect the performance specifications is the critical factor for successfully designing such bilateral haptic telemanipulation control systems.

The realization of robust stability and performance specifications is mainly reflected by the performance weighting functions, control weighting functions and noise weighting functions. In addition, a weighting function can also be chosen for a model uncertainty. If the model uncertainty is unspecified, then the selection of the weighting function is mainly decided by the robustness requirements.

The second feature of the design framework is that the structures of the proposed and designed controllers are different form other methods in the literature. In this thesis, both the master and the slave controllers adopt the 2-DOF architecture in order to avoid large overshoots from the system responses and therefore improve system performance while many other methods in the literature take only a 1-DOF controller for the master and a 2-DOF controller for the slave. Moreover, in the constrained movement case, a global controller in the master side (or local site) takes both the environment reaction force and the difference between the displacement of the master manipulator and the delayed displacement of the slave manipulator as its input to achieve stability and transparency of the closed-loop system.

The third feature of the design framework is that the time delays in both directions, forward and backward, of the communication channel are viewed as being different. The channel is treated as a parametric uncertain second-order system with an input multiplicative uncertainty in both directions such that a more reasonable communication channel model is considered. The simulation results show that, in this manner and with the use of the proposed controllers, variations in the time delay do not threat the stability and performance of the system.

μ-synthesis and analysis theories, which the proposed design framework is based on, enable the designer to evaluate the performance and robustness level of a controller in face of the modeling uncertainties. The same framework can be used for the design and at the same time for the frequency domain analysis of the controller. It also allows designers to trade off between performance and robustness objectives or among different performance objectives. Performance and stability requirements such as tracking error magnitudes, control activity level and disturbance rejection can be easily expressed using LFT techniques. Although the design process consists of tuning weighting functions, the controller synthesis can be automated with excellent Matlab μ-Synthesis and Analysis Toolbox.

5.2 Future Research Directions

The applicability of μ-analysis and synthesis is conditioned by the availability of adequate uncertainty models based on LFT techniques. LFT modeling is a time-consuming task for a complex system and requires the plants in the system to be all proper; otherwise, it cannot be carried out. There are some other disadvantages with the μ-analysis and synthesis method. First, the resulting controllers are of the orders corresponding to the scaled interconnection structures and consequently their orders are often very high although it is possible to reduce the orders of the controllers using model reduction techniques. Second, it is often necessary to select a number of weighting functions, which is usually non-trivial and a time-consuming task. The third disadvantage is that the D-K iteration does not necessarily guarantee finding the global optimum of

equation (2.48). There is also the difficulty in defining a stopping criterion for the optimization routine.

Therefore, one of the tasks in the future is to work out an efficient method of LFT modeling order reduction and apply it into the system general open interconnection structure before the controller synthesis such that the number of the resulting controller order can be greatly reduced. In addition, to discover a set of general, optimal and efficient rules for the tuning of weighting functions is also an important issue.

The recent trend of applying Linear Matrix Inequality (LMI) techniques [42] into the controller design has also received attention. This is due to the fact that many design objectives and constraints can be expressed as LMIs. In addition, formulated as LMIs, a control problem can be solved by using efficient convex optimization algorithms (such as Matlab LMI toolbox). Furthermore, problems with multiple constraints and objectives that have no analytical solutions in terms of matrix equations may be solvable using LMI framework. Therefore, developing a controller design framework which is based on LMI method for a time delayed bilateral haptic telemanipulation system in which every plant possesses uncertainty becomes a topic of in the future research.

Finally, much effort should be spent on the study, design, analysis and control of multi-DOF as well as nonlinear and parametric time-varying haptic telemanipulation systems.

References

[1] R.J. Anderson and M.W. Spong, "*Bilateral Control of Teleoperators with Time Delay*", IEEE Transactions on Automatic Control, 34 (5), 1989, pp. 494-501.

[2] K. Hashtrudi-Zaad and S.E. Salcudean, "*On the Use of Local Force Feedback for Transparent Teleoperation*", in: IEEE International Conference on Robotics and Automation, ICRA'99, Detroit, MI, May 1999, pp. 1863-1869.

[3] D.A. Lawrence, "*Stability and Transparency in Bilateral Teleoperation*", IEEE Transactions on Robotics and Automation, 9 (5), 1993, pp. 624-637.

[4] M. C. Cavusoglu, A. Sherman and F. Tendick, "*Bilateral Controller Design for Telemanipulation in Soft Environments*", IEEE Intl. Conf. Robotics and Automation, 2001, pp. 1045-1052.

[5] J. E. Colgate, "*Robust Impedance Shaping Telemanipulation*", IEEE Transactions on Robotics and Automation, 9(4), 1993, pp. 374-384.

[6] A. Sherman, M. C. Cavusoglu and F. Tendick, "*Comparison of Teleoperator Control Architectures for Palpation Task*", Proc. IMECE'00 Symp. On Haptic Interfaces for VE & Teleoperator System, November 2000, Orlando, Florida, USA.

[7] K. Hashtrudi-Zaad and S.E. Salcudean, "*Transparency in Time-Delayed Systems and the Effect of Local Force Feedback for Transparent Teleoperation*", IEEE Transactions on Robotics and Automation, 18(1), 2002, pp. 108-114.

[8] J. C. Doyle, "*Analysis of Feedback Systems with Structured Uncertainties*", Proc. Inst. Electr. Eng., Pt. D, vol. 129, 1982, pp. 242-250.

[9] A. Packard, J. Doyle, and G. Balas, "*Linear, Multivariable Robust Control With a μ Perspective*", Trans. of the ASME, vol. 115, June 1993, pp.426-438.

[10] K. Zhou and J.C. Doyle, "*Essentials of Robust Control*", Prentice Hall, UpperSaddle River, 1998.

[11] G. J. Balas, J.C. Doyle, K. Glover, and R. Smith, "*μ-Analysis and Synthesis Toolbox User's Guide*", The MathWorks Inc., 2001.

[12] T. Massie and J. Salisbury, "*The PHANToM Haptic Interface: A Device for Probing Virtual Objects*", Proc. ASME, Symposium on Haptic Interfaces for VE & Teleoperation System, Chicago, IL, 1994, pp. 295-302.

[13] Y. Yokokohji, T. Yoshikawa, "*Bilateral Control of Master–Slave Manipulators for Ideal Kinesthetic Coupling — Formulation and Experiment*", IEEE Transactions on Robotics and Automation, 10 (5), 1994, pp. 605-620.

[14] G. M. H. Leung and B. A. Francis, "Bilateral Control of Teleoperators with Time Delay through A Digital Communication Channel", Proc. 30th Ann. Allerton Conf. Commun., Contr., Compu., 1992, pp. 692-701.

[15] B. Hannaford, "*Stability and Performance Tradeoffs in Bilateral Telemanipulation*", Proc. IEEE International Conference on Robotics and Automation, ICRA'89, Scottsdale, AZ, May 1989, pp. 1764-1767.

[16] H. Kazerooni, B. J. Waibel, and S. Kim, "*On the Stability of Robot Compliant Motion Control: Theory and Experiments*", Journal of Dynamic Systems, Measurement, and Control, vol. 112, 1990, pp. 417-426.

[17] W.S. Kim, B. Hannaford, A.K. Bejczy, "*Force-Reflection and Shared Compliant Control in Operating Telemanipulators with Time Delay*", IEEE Transactions on Robotics and Automation, 8 (2), 1992, pp.176-185.

[18] G. Niemeyer, J.E. Slotine, "*Stable Adaptive Teleoperation*", Proc. American Control Conference, 1990, pp. 1186-1191.

[19] G. Niemeyer, J.E. Slotine, "*Towards Force-Reflecting Teleoperation Over The Internet*", Proc. IEEE Intl. Conf. Robotics & Automation, Leuven, Belgium, May 1998, pp.1909-1915.

[20] W.S. Kim, "Developments of New Force Reflecting Control Schemes and an Application to a Teleoperation Training Simulator", Proc. IEEE International Conference on Robotics and Automation, ICRA'92, Nice, France, May 1992, pp. 1412-1419.

[21] J.H. Park and H.C. Cho, "*Sliding-Mode Controller for Bilateral Teleoperation with Varying Time Delay*", Proc. IEEE International Conference on Advanced Intellectual Mechanics, AIM'99, Atlanta, GA, September 1999, pp. 311-316.

[22] J. Yan and S. E. Salcudean, "*Teleoperation Controller Design using H_∞-Optimization with Application to Motion-Scaling*", IEEE Trans. Contr. Syst. Technol., vol. 4, May 1996, pp. 244-258.

[23] Z. Hu, S. E. Salcudean, and P. D. Loewen, "*Optimization-Based Controller Design for Teleoperation Systems*", Proc. 13th IFAC World Congress, San Francisco, CA, 1996, pp. 405-410.

[24] W.H. Zhu and S.E. Salcudean, "*Teleoperation with Adaptive Motion/Force Control*", IEEE Int. Conf. on Rob. and Aut., Detroit, MI, May 1999, pp. 231-237.

[25] W.H. Zhu and S.E. Salcudean, "*Stability Guaranteed Teleoperation: An Adaptive Motion/Force Control Approach*", IEEE Trans. on Automatic Control, 45(11), Nov. 2000, pp. 1951-1969.

[26] H. Kazerooni, T. I. Tsay and K. Hollerbach, "*A Controller Design Framework for Telerobotic Systems*", IEEE Trans. Contr. Syst. Tech., 1(1), 1993, pp. 50-62.

[27] T.I. Tsay and H. Kazerooni, "*A Design Framework for Telerobotics Using the H_∞ Approach*", Proc. American Control Conference, 1992, pp. 2931-2935.

[28] G. M. H. Leung, B. A. Francis, and J. Apkarian, "*Bilateral Controller for Teleoperators with Time Delay via μ-Synthesis*", IEEE Trans. Robotics & Automation, Vol. 11, No. 1, Feb. 1995, pp. 105-116.

[29] J. Doyle, A. Packard and K. Zhou, "*Reviews of LFTs, LMIs, and μ*", Proc. 30th Conf. on Decision and Control, Brighton, England, Dec. 1991, pp. 1227-1232.

[30] A. Packard and J. Doyle, "The Complex Structured Singular Value", Automatica, vol. 29, 1994, pp. 71-109.

[31] P. Lambrechts, J. Terlouw, S. Bennani and M. Steinbuch, "*Parametric Uncertainty Modeling Using LFTs*", Proc. American Control Conference, San Francisco, CA, June 1993.

[32] C. Scherer, "*Theory of Robust Control*", Lecture Notes, 2001. (available at www.dcsc.tudelft.nl/~cscherer/, last access in Sept. 2004)

[33] R. S. Sanchez-Pena and M. Sznaier, "*Robust Systems Theory and Applications*", Wiley, 1998.

[34] S. Toffner-Clausen, P. Andersen and J. Stoustrup, "*Robust Control*", 4th edition, Aalborg University, Denmark, 2001

[35] P. Lundstrom, S. Skogestad and Z. Q. Wang, "*Performance Weight Selection for H-infinity and μ-Control Methods*", Trans. Inst. MC. 13(5), 1991, pp. 241-252.

[36] J. Markerink, S. Bennani and B. Mulder, "*Design of a Robust, Scheduled Controller for the HIRM Using μ-Synthesis*", GARTEUR/TP-088-29, 1997

[37] J. Schuring and R. M. P. Groverde, "*A μ-Synthesis Approach (2)*", from Robust Flight Control: A Design Challenge, Edited by J.F. Magni, S. Bennani and J. Terlouw, Springer Verlag, Lecture Notes in Control and Information Sciences, No. 224, 1997, pp.341-359.

[38] M. C. Cavusoglu, D. Feygin, and F. Tendick, "*A Critical Study of the Mechanical and Electrical Properties of the PHANToM Haptic Interface and Improvements for High Performance Control*", Presence, Vol. 11, No. 6, Dec. 2002.

[39] M. C. Cavusoglu, "*Telesurgery and Surgical Simulation: Design, Modeling, and Evaluation of Haptic Interfaces to Real and Virtual Surgical Environments*", Ph. D. Dissertation, U. of CA, Berkeley, 2000.

[40] P.Lundstrom, S. Skogestad and J. C. Doyle, "*Two-Degree-of-Freedom Controller Design for an Ill-Conditioned distillation Process Using μ-Synthesis*", IEEE Trans. on Control System Technology, Vol. 7, No. 1, January 1999, pp. 12-21.

[41] I. Szaszi and P. Gaspar, "Robust Servo Control Design Using The H_∞/μ Method, Periodica Polytechnica Ser. Transp. Eng. Vol. 27, No. 1-2, 1999, pp. 3-16.

[42] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan, "*Linear Matrix Inequalities in System and Control Theory*", SIAM, 1994.

[43] S. Bennani, G. Looye and C. Scherer, "*μ-Synthesis*", from Robust Flight Control: A Design Challenge, Edited by J.F. Magni, S. Bennani and J. Terlouw, Springer

Verlag, Lecture Notes in Control and Information Sciences, No. 224, 1997, pp. 81-101.

Appendix A

Matlab Codes for the Controller Synthesis and Analysis

A.1 Communication Channel Modeling

A.1.1 Wd.m

```
% Generate Multiplicative Weight Wd(s) for Comm. Channel

t0=4;
t=[t0-0.2;t0-0.1;t0;t0+0.1;t0+0.2;];
w=logspace(-1,3,100);

mat=[];

for i=1:5
   gd0=(1-0.5.*t(i).*j.*w-0.125.*t(i).^2.*w.^2)./...
          (1+0.5.*t(i).*j.*w-0.125.*t(i).^2.*w.^2);
   gd=exp(-t(i).*j.*w);
   Del=(gd-gd0)./gd0;
   mat=[mat;20*log10(abs(Del))];
end

figure(1); clf;
semilogx(w,mat);

mf= ginput(10);
magg=vpck(10.^(mf(:,2)./20),mf(:,1));

figure(2);
wdel=fitmag(magg);

[a,b,c,d]=unpck(wdel);
[num,den]=ss2tf(a,b,c,d);
[z,p,k]=ss2zp(a,b,c,d);
```

A.1.2 Wd2.m

```
% Drawing the weight curve and delay uncertainty: Fig. 3.7

t0=4;
t=[t0-0.2;t0-0.1;t0;t0+0.1;t0+0.2;];      % deltat=0.2
w=logspace(-1,3,100);

mat=[];

for i=1:5
   gd0=(1-0.5.*t(i).*j.*w-0.125.*t(i).^2.*w.^2)./...
            (1+0.5.*t(i).*j.*w-0.125.*t(i).^2.*w.^2);
   gd=exp(-t(i).*j.*w);
   Del=(gd-gd0)./gd0;
   mat=[mat;20*log10(abs(Del))];
end

figure(1);
semilogx(w,mat);

wnum=-2.3458.*w.^2+0.5671*j.*w+0.0320;
wden=-w.^2+1.7263.*j.*w+2.3478;
wd=wnum./wden;
wdlog=20.*log10(abs(wd));

hold on; ishold;
semilogx(w,wdlog,'m*:');
hold off; ishold
```

A.1.3 Gdp.m

```
% Comm. Channel : 2nd-order Pade approximation

t0=4;
deltat=0.2;  % Deviation Coefficient

matt11=[0 1; deltat 1];
matt12=[0 1; deltat 1];
matt13=[0 1; deltat 1];

c11=[-t0];
c12=[t0/2];
c13=[t0^2/8];

intt11=nd2sys([1], [1 0]);          % 1/s
intt12=nd2sys([1], [1 0]);

systemnames='matt11 matt12 matt13 intt11 intt12 c11 c12 c13';
sysoutname='Gdp1';
```

```
inputvar='[wt11; wt12; wt13; ud1]';

input_to_matt11='[wt11; c11]';
input_to_matt12='[wt12; c12+matt13(2)]';
input_to_matt13='[wt13; c13]';

input_to_intt11='[intt12]';
input_to_intt12='[ud1-matt12(2)]';

input_to_c11='[intt12]';
input_to_c12='[intt12]';
input_to_c13='[intt11]';

outputvar='[matt11(1); matt12(1); matt13(1); ud1+matt11(2)+intt12]';
cleanupsysic='yes';
sysic
```

A.1.4 Gd.m

```
% Delay Link with mixed uncertainty
% The generated System Matrix is Gdd

Gdp; %Get Gdp1, System Matrix of the model with parametric uncertainty

% Define delay uncertainty weight: Wd(s)
wnum1=[2.3458 0.5671 0.0320];
wden1=[1 1.7263 2.3478];
Wd1=nd2sys(wnum1,wden1);

% Interconnection
systemnames='Gdp1 Wd1';   % Wd: weight of multiplicative input
uncertainty
sysoutname='Gdd1';

inputvar='[w11; w12; w13; w14; ud11]'; %

input_to_Gdp1='[w11; w12; w13; ud11+w14]';
input_to_Wd1='[ud11]';

outputvar='[Gdp1(1:3); Wd1; Gdp1(4)]';
cleanupsysic='yes';
sysic
```

A.2 Interconnection Structure

A.2.1 Gmm.m

```
% Master and slave manipulator with multiplicative uncertainty
% System output is Gm

Gm0=nd2sys([3.004e005 1.717e006 2.764e010],[1 368.3 4.614e005 0 0]);
Wm=nd2sys(1*[1 10], [1 200]);

systemnames='Gm0 Wm';
sysoutname='Gm';

inputvar='[wp; um]';

input_to_Gm0='[um+wp]';
input_to_Wm='[um]';

outputvar='[Wm; Gm0]';
cleanupsysic='yes';
sysic
```

A.2.2 Master.m

```
% Master Open-loop Interconnection Pm
% 2-DOF controller

Gmm; % nominal model with Multiplicative uncertainty: Gmmul

% Ideal Reference Model Grm(s)
zeta=0.7;
wn=142;
Grm=nd2sys(wn^2, [1 2*zeta*wn wn^2]);

% Performance Weight
Wem=nd2sys([0.01 1],[1 0.01]);

% Control Weight
Wum=nd2sys([10^(-3) 0],[10^(-6) 1]);

% Noise Weight
Wnm=nd2sys(0.1*[1 0.01],[1 200]);

% Interconnection

systemnames='Gm Grm Wnm Wum Wem';
```

```
sysoutname='Pm';
inputvar='[wp; fh; nm; um]';

input_to_Grm='[fh]';
input_to_Wnm='[nm]';
input_to_Wum='[um]';
input_to_Wem='[Grm-Gm(2)]';
input_to_Gm='[wp; um]';

outputvar='[Gm(1); Wem; Wum; fh; Gm(2)+Wnm]';
cleanupsysic='yes';
sysic
```

A.2.3 Slave.m

```
% Slave Open-loop Interconnection Ps
% 2-DOF controller

Gmm; % nominal model with Multiplicative uncertainty: Gmmul

% Ideal Reference Model Grs(s)
Grs=1;

% Performance Weight
Wes=nd2sys([0.01 1],[1 0.01]);

% Control Weight
Wus=nd2sys([10^(-3) 0],[10^(-6) 1]);

% Noise Weight
Wns=nd2sys(0.1*[1 0.01],[1 200]);

% Interconnection

systemnames='Gm Grs Wns Wus Wes';
sysoutname='Ps';
inputvar='[wss; xmd; ns; us]';

input_to_Grs='[xmd]';
input_to_Wns='[ns]';
input_to_Wus='[us]';
input_to_Wes='[Grs-Gm(2)]';
input_to_Gm='[wss; us]';

outputvar='[Gm(1); Wes; Wus; xmd; Gm(2)+Wns]';
cleanupsysic='yes';
sysic
```

A.2.4 Gloabalpf.m

```
% Global System Open-Interconnection: Pg
% min||Weg(Grg*fh-Xs)|| &  min||Weg(Xm-Xs)|| & min||Weg(feg-fe)||

Gmm;  % master nominal model with Multiplicativ uncertainty: Gm
Gd; % Generate Comm. Channel: Gdd1
Gdd2=Gdd1; Gdd3=Gdd1;

Gs10=nd2sys([3.004e005 1.717e006 2.764e010],[1 368.3 4.614e005]);
Ws=nd2sys(1*[1 10], [1 200]);

% nominal slave with Environment Model
bbar=0.0175;
kbar=0.35;
db=0.5;
dk=0.5;

int1=nd2sys([1], [1 0]);          % 1/s
int2=nd2sys([1], [1 0]);
matb=[0 1; bbar*db bbar];
matk=[0 1; kbar*dk kbar];

% Master Controller
% forward
Ka1=tf([-0.01451 1606 3.846e006 5.75e009...
            1.448e012 5.323e014 8.462e016 ], ...
    [1 3595 6.349e006 4.848e009 1.231e012...
            4.189e014 4.979e016 4.979e014]);
% Negative Feedback
Ka2=tf([0.8139 8.156e005 5.04e008 4.6e011...
            9.729e013 4.179e015 8.464e016],...
    [1 3595 6.349e006 4.848e009 1.231e012...
            4.189e014 4.979e016 4.979e014]);

[aka1,bka1,cka1,dka1]=ssdata(Ka1);
Km1=pck(aka1,bka1,cka1,dka1);

[aka2,bka2,cka2,dka2]=ssdata(Ka2);
Km2=pck(aka2,bka2,cka2,dka2);

% Slave Controller
% forward
Kb1=tf([0.001508 1513 3.861e006 5.525e009...
            1.434e012 5.114e014 8.344e016],...
    [1 3592 6.341e006 4.833e009 1.224e012...
            4.173e014  4.921e016 4.921e014]);
% Negative Feedback
Kb2=tf([0.8056 8.061e005 4.978e008 4.544e011...
            9.601e013 4.121e015 8.348e016],...
    [1 3592 6.341e006 4.833e009 1.224e012...
            4.173e014  4.921e016 4.921e014]);

[akb1,bkb1,ckb1,dkb1]=ssdata(Kb1);
Ks1=pck(akb1,bkb1,ckb1,dkb1);
```

```
[akb2,bkb2,ckb2,dkb2]=ssdata(Kb2);
Ks2=pck(akb2,bkb2,ckb2,dkb2);

% Ideal Reference Model Grg(s) for Path from Fh to Xs
zeta=0.7; wn=142;
Grg=nd2sys(wn^2, [1 2*zeta*wn wn^2]);

% Performance Weight
Weg1=nd2sys([0.01 1],[1 0.01]);
Weg2=nd2sys([0.01 1],[1 0.01]);
Weg3=nd2sys([0.01 1],[1 0.01]);

% Control Weight
Wug=nd2sys([10^(-3) 0],[10^(-6) 1]);

% Noise Weight
Wng=nd2sys(0.1*[1 0.01],[1 200]);

% Interconnection

systemnames='Gm Gs10 Ws Grg Km1 Km2 Ks1 Ks2 Wug Wng ...
               Weg1 Weg2 Weg3 matb matk int1 int2 Gdd1 Gdd2 Gdd3';

sysoutname='Pg';

inputvar='[wg1; wg2; wg3; wg4; wg5; wg6; wg7; wg8; wg13; wg14; ...
               wg15; wg9; wg10; wg16; wg11; wg12; fh; ng; ug]';
% wg1, wg2: Ke, Be;
% wg3, wg4, wg5; wg6, wg7, wg8; wg13; wg14; wg15: Gdp1, Gdp2, Gdp3;
% wg9, wg10, wg16: Wd1, Wd2, Wd3
% wg11, wg12: Gm, Gs

input_to_matk='[wg1; int1]';
input_to_matb='[wg2; int2]';

input_to_int1='[int2]';
input_to_int2='[Gs10]';

input_to_Gm='[wg11; Km1-Km2-ug]';
input_to_Gs10='[wg12+Ks1-Ks2-matb(2)-matk(2)]';
input_to_Ws='[Ks1-Ks2-matb(2)-matk(2)]';

input_to_Gdd1='[wg3; wg4; wg5; wg9; Gm(2)]';
input_to_Gdd2='[wg6; wg7; wg8; wg10; int1+Wng]';
input_to_Gdd3='[wg13; wg14; wg15; wg16; matb(2)+matk(2)]';

input_to_Wug='[ug]';
input_to_Wng='[ng]';
input_to_Weg1='[Grg-int1]';
input_to_Weg2='[Gdd1(5)-int1]';
input_to_Weg3='[ug-matb(2)-matk(2)]';

input_to_Grg='[fh]';

input_to_Km1='[fh]';
input_to_Km2='[Gm(2)]';
```

```
input_to_Ks1='[Gdd1(5)]';
input_to_Ks2='[int1+Wng]';

outputvar='[matk(1); matb(1); Gdd1(1:3); Gdd2(1:3); Gdd3(1:3); ...
       Gdd1(4); Gdd2(4); Gdd3(4); Gm(1); Ws; Weg1; Weg2; Weg3; ...
       Wug; Gm(2)-Gdd2(5); Gdd3(5)]';

cleanupsysic='yes';
sysic
```

A.2.5 Master1dof.m

```
% Master Open-loop Interconnection Pm
% 1-DOF controller

Gmm; % nominal model with Multiplicativ uncertainty: Gmmul

% Ideal Reference Model Grm(s)

zeta=0.7; wn=142;
Grm=nd2sys(wn^2, [1 2*zeta*wn wn^2]);

% Performance Weight Wem
Wem=nd2sys([0.01 1],[1 0.01]);

% Control Weight
Wum=nd2sys([10^(-3) 0],[10^(-6) 1]);

% Noise Weight
Wnm=nd2sys(0.1*[1 0.01],[1 200]);

% Interconnection

systemnames='Gm Grm Wnm Wum Wem';    %
sysoutname='Pm';
inputvar='[wp; fh; nm; um]';

input_to_Grm='[fh]';
input_to_Wnm='[nm]';
input_to_Wum='[um]';
input_to_Wem='[Grm-Gm(2)]';
input_to_Gm='[wp; fh-um]';

outputvar='[Gm(1); Wem; Wum; Gm(2)+Wnm]';
cleanupsysic='yes';
sysic
```

A.3 Controller Order Reduction and Transformation

A.3.1 K0.m

```
% Full order Controller

[a,b,c,d]=unpck(K1ss); % Master and Slave
%[a,b,c,d]=unpck(K7ss); % Global
sys=ss(a,b,c,d);
k=tf(sys)
```

A.3.2 Kmr.m

```
% Order-Reduced master and slave Controller

[ka,kb,kc,kd]=unpck(K1ss);
[kar,kbr,kcr,kdr,ktb,ksv]=schmr(ka,kb,kc,kd,1,7);
sys=ss(kar,kbr,kcr,kdr);
kr=pck(kar,kbr,kcr,kdr);
k=tf(sys)
```

A.3.3 Kgr.m

```
% Order-Reduced Global Controller

[ka,kb,kc,kd]=unpck(K7ss);
[kar,kbr,kcr,kdr,ktb,ksv]=schmr(ka,kb,kc,kd,2,1e-2);
sys=ss(kar,kbr,kcr,kdr);
kr=pck(kar,kbr,kcr,kdr);
k=tf(sys)
```

A.4 Step Response

A.4.1 Sridnom.m

```
% ideal and nominal model (Master subsys): Figure 4.2

g=tf([3.004e005 1.717e006 2.764e010],[1 368.3 4.614e005 0 0]);

% Master controller
k(1)=tf([-0.01451 1606 3.846e006 5.75e009 ...
          1.448e012 5.323e014 8.462e016],...
    [1 3595 6.349e006 4.848e009 1.231e012 ...
          4.189e014 4.979e016 4.979e014]);
k(2)=tf([0.8139 8.156e005 5.04e008 4.6e011 ...
          9.729e013 4.179e015 8.464e016],...
    [1 3595 6.349e006 4.848e009 1.231e012 ...
          4.189e014 4.979e016 4.979e014]);

ssys0=feedback(g,k(2));
sys0=series(k(1),ssys0);  %  Nominal master subsys

sysr=tf(142^2, [1 2*0.7*142 142^2]); % Reference model:  Eq. (3.21)

figure(1);
step(sys0, 1);
hold on; ishold;
step(sysr,1);
hold off; ishold
```

A.4.2 Sr1dof.m

```
% For drawing Fig. 3.11: Closed-loop Master step Response: 1-dof

g=tf([3.004e005 1.717e006 2.764e010],[1 368.3 4.614e005 0 0]);

% full 20th order controller
k=tf([124.9 1.251e008 1.556e011 1.377e014 7.326e016 ...
        2.236e019 4.203e021 5.088e023 3.952e025 1.94e027 ...
        5.486e028 8.596e029 4.874e030 8.421e030 4.033e030 ...
        5.624e029 2.559e028 4.342e026 3.676e024 1.506e022],...
     [1 2.868e004 1.517e008 4.139e011 3.199e014 1.412e017...
        4.759e019 1.15e022 1.849e024 1.92e026 1.185e028 ...
        3.934e029 3.266e030 9.328e030 1.062e031 4.357e030 ...
        5.793e029 2.597e028 4.386e026 3.704e024 1.514e022]);

sys0=feedback(g,k); % closed-loop: nominal system
```

```
% G=G0(1+W)
s1=1;
Wm1=tf([1 10],[1 200]);
wmu1=parallel(Wm1,s1);
gu1=series(wmu1,g);
sys1=feedback(gu1,k);

% G=G0(1-W)
wmu2=parallel(-Wm1,s1);
gu2=series(wmu2,g);
sys2=feedback(gu2,k);

figure(1);
step(sys0, 1);
hold on; ishold;
step(sys1,1);
hold on; ishold;
step(sys2,1);
hold off; ishold
```

A.4.3 Srm.m

```
% Fig. 3.14: Master Subsys Step Response

g=tf([3.004e005 1.717e006 2.764e010],[1 368.3 4.614e005 0 0]);

k(1)=tf([-0.01451 1606 3.846e006 5.75e009 ...
           1.448e012 5.323e014 8.462e016],...
    [1 3595 6.349e006 4.848e009 1.231e012 ...
           4.189e014 4.979e016 4.979e014]);
k(2)=tf([-0.8139 -8.156e005 -5.04e008 -4.6e011 ...
           -9.729e013 -4.179e015 -8.464e016],...
    [1 3595 6.349e006 4.848e009 1.231e012 ...
           4.189e014 4.979e016 4.979e014]);

ssys0=feedback(g,-k(2));
sys0=series(k(1),ssys0);  % Nominal closed-loop master subsys

s1=1;
Wm1=tf([1 10],[1 200]);

% G=G0(1+W)
wmu1=parallel(Wm1,s1);
gu1=series(wmu1,g);
ssys1=feedback(gu1,-k(2));
sys1=series(k(1),ssys1);

% G=G0(1-W)
```

```
wmu2=parallel(-Wm1,s1);
gu2=series(wmu2,g);
ssys2=feedback(gu2,-k(2));
sys2=series(k(1),ssys2);

figure(1);
step(sys0, sys1, sys2, 1);
```

A.4.4 Srs.m

```
% Fig. 3.18: Slave Subsys Step Response

g=tf([3.004e005 1.717e006 2.764e010],[1 368.3 4.614e005 0 0]);

k(1)=tf([0.001508 1513 3.861e006 5.525e009 ...
            1.434e012 5.114e014 8.344e016],...
          [1 3592 6.341e006 4.833e009 1.224e012 ...
             4.173e014 4.921e016 4.921e014]);
k(2)=tf([-0.8056 -8.061e005 -4.978e008 -4.544e011 ...
            -9.601e013 -4.121e015 -8.348e016],...
          [1 3592 6.341e006 4.833e009 1.224e012 ...
             4.173e014 4.921e016 4.921e014]);

ssys0=feedback(g,-k(2));
sys0=series(k(1),ssys0);  % Nominal

s1=1;
Wm1=tf([1 10],[1 200]);

% G=G0(1+W)
wmu1=parallel(Wm1,s1);
gu1=series(wmu1,g);
ssys1=feedback(gu1,-k(2));
sys1=series(k(1),ssys1);

% G=G0(1-W)
 wmu2=parallel(-Wm1,s1);
gu2=series(wmu2,g);
ssys2=feedback(gu2,-k(2));
sys2=series(k(1),ssys2);

figure(1);
step(sys0, 1);
hold on; ishold;
step(sys1,1);
hold on;ishold;
step(sys2,1);
hold off; ishold
```

A.5 Analysis in Frequency Domain

A.5.1 Pmu.m

```
% Master Robustness Analysis in Freq. Domain: Fig. 4.1
% Running after executing Master.m DKITGUI and Kmr or K0

M=starp(Pm,kr);

om=logspace(-2,3,100);

Mg=frsp(M,om);

uncblk=[1 1];  % uncertainty block
fictblk=[2 2];  % fivtious uncertainty block
deltaset=[uncblk; fictblk];
bnds=mu(Mg, deltaset);
clf;

figure(10);
vplot('liv,m',sel(bnds,1,1),'*w',sel(Mg,2,2),'--w',sel(Mg,1,1),'-w');
xlabel('Frequency (rad/sec)')
ylabel('mu');
title('Robust Performance (*), Robust Stability (-)...
          and Nominal Performance (--)');
```

A.5.2 Psu.m

```
% Slave Robustness Analysis in Freq. Domain: Fig. 4.3
% Running after executing Slave.m DKITGUI and Kmr or K0

M=starp(Ps,kr);

om=logspace(-2,3,100);

Mg=frsp(M,om);

uncblk=[1 1];  % uncertainty block
fictblk=[2 2];  % fivtious uncertainty block
deltaset=[uncblk; fictblk];

bnds=mu(Mg, deltaset);
clf;

figure(10);
vplot('liv,m',sel(bnds,1,1),'*w',sel(Mg,2,2),'--w',sel(Mg,1,1),'-w');
```

```
xlabel('Frequency (rad/sec)')
ylabel('mu');
title('Robust Performance (*), Robust Stability (-)...
          and Nominal Performance (--)');
```

A.5.3 Pgu.m

```
% Global System Robustness Analysis: Fig.4.4
% Running after executing Globalpf.m, DKITGUI, Kmr.m or K0.m

M=starp(Pg,kr);

om=logspace(-2,3,100);

Mg=frsp(M,om);

% uncertainty block
uncblk=[-1 1; -1 1; -1 1; -1 1; -1 1; -1 1; -1 1; ...
        1 1; 1 1; 1 1; 1 1; -1 1; -1 1; -1 1; 1 1];
fictblk=[2 4];  % fivtious uncertainty block
deltaset=[uncblk; fictblk];
bnds=mu(Mg, deltaset);
clf;

figure(10);
vplot('liv,m',sel(bnds,1,1),'*w',sel(Mg,2,2),'--w',sel(Mg,1,1),'-w');
xlabel('Frequency (rad/sec)')
ylabel('mu');
title('Robust Performance (*), Robust Stability (-)...
          and Nominal Performance (--)');
```

Appendix B

Simulation Results of Other Scenarios

For the constrained movement case, referring to Section 4.3.2, there are some other simulation results shown in the figures B.1 to B.14 for the following typical scenarios, respectively:

1) $G_m = G_0(1+W_{mul})$, $G_s = G_0(1+W_{mul})$ and $G_e = B_0(1+db)s + K_0(1-dk)$;

2) $G_m = G_0(1+W_{mul})$, $G_s = G_0(1+W_{mul})$ and $G_e = B_0(1-db)s + K_0(1+dk)$;

3) $G_m = G_0(1+W_{mul})$, $G_s = G_0(1+W_{mul})$ and $G_e = B_0(1-db)s + K_0(1-dk)$;

4) $G_m = G_0(1+W_{mul})$, $G_s = G_0(1-W_{mul})$ and $G_e = B_0(1+db)s + K_0(1+dk)$;

5) $G_m = G_0(1+W_{mul})$, $G_s = G_0(1-W_{mul})$ and $G_e = B_0(1+db)s + K_0(1-dk)$;

6) $G_m = G_0(1+W_{mul})$, $G_s = G_0(1-W_{mul})$ and $G_e = B_0(1-db)s + K_0(1+dk)$;

7) $G_m = G_0(1+W_{mul})$, $G_s = G_0(1-W_{mul})$ and $G_e = B_0(1-db)s + K_0(1-dk)$;

8) $G_m = G_0(1-W_{mul})$, $G_s = G_0(1+W_{mul})$ and $G_e = B_0(1+db)s + K_0(1+dk)$;

9) $G_m = G_0(1-W_{mul})$, $G_s = G_0(1+W_{mul})$ and $G_e = B_0(1+db)s + K_0(1-dk)$;

10) $G_m = G_0(1-W_{mul})$, $G_s = G_0(1+W_{mul})$ and $G_e = B_0(1-db)s + K_0(1+dk)$;

11) $G_m = G_0(1-W_{mul})$, $G_s = G_0(1+W_{mul})$ and $G_e = B_0(1-db)s + K_0(1-dk)$;

12) $G_m = G_0(1-W_{mul})$, $G_s = G_0(1-W_{mul})$ and $G_e = B_0(1+db)s + K_0(1+dk)$.

13) $G_m = G_0(1-W_{mul})$, $G_s = G_0(1-W_{mul})$ and $G_e = B_0(1+db)s + K_0(1-dk)$;

14) $G_m = G_0(1-W_{mul})$, $G_s = G_0(1-W_{mul})$ and $G_e = B_0(1-db)s + K_0(1+dk)$.

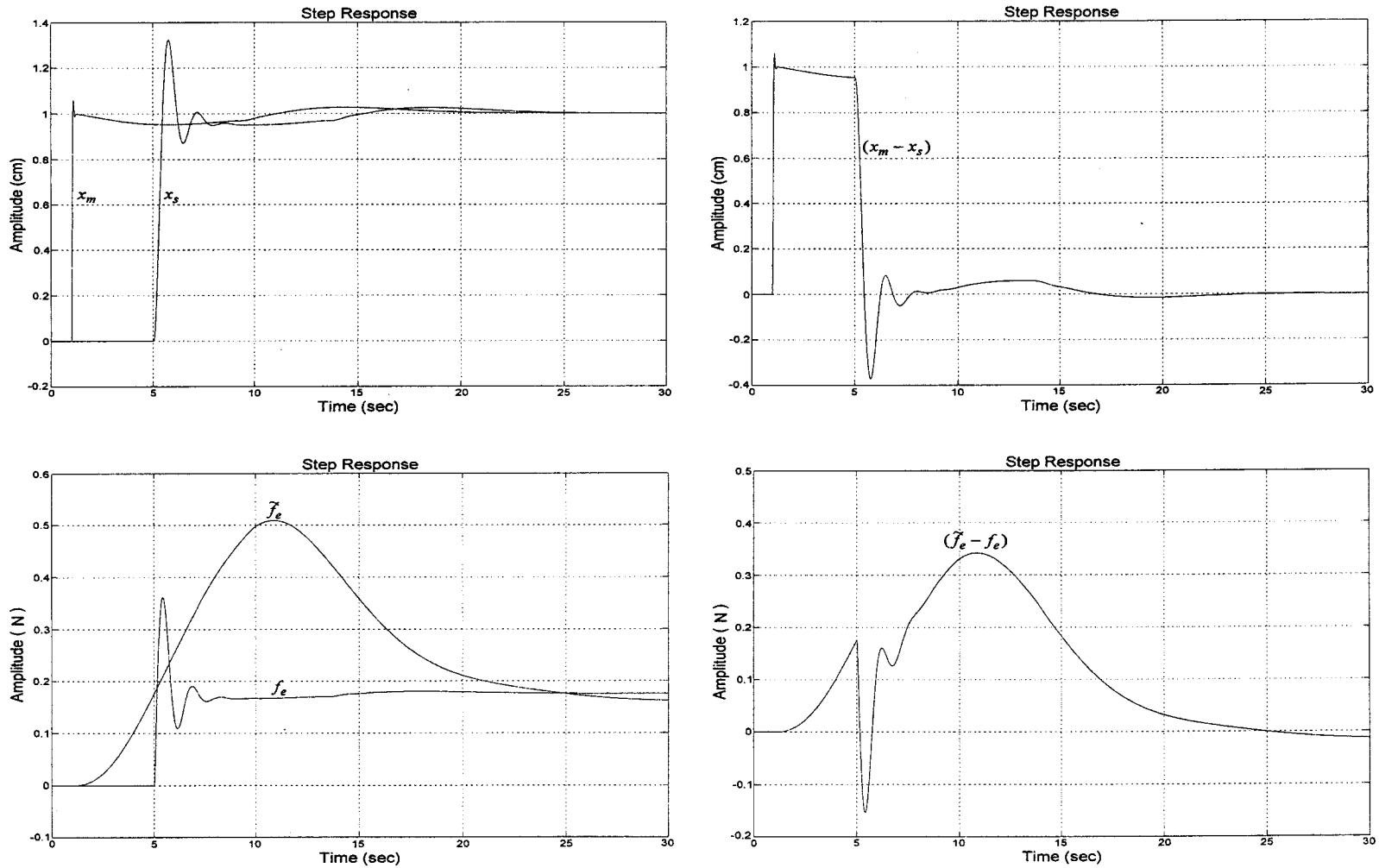


Fig. B.1 Case 1: $G_m = G_0(1+W_{mul})$, $G_s = G_0(1+W_{mul})$ **and** $G_e = B_0(1+db)s + K_0(1-dk)$

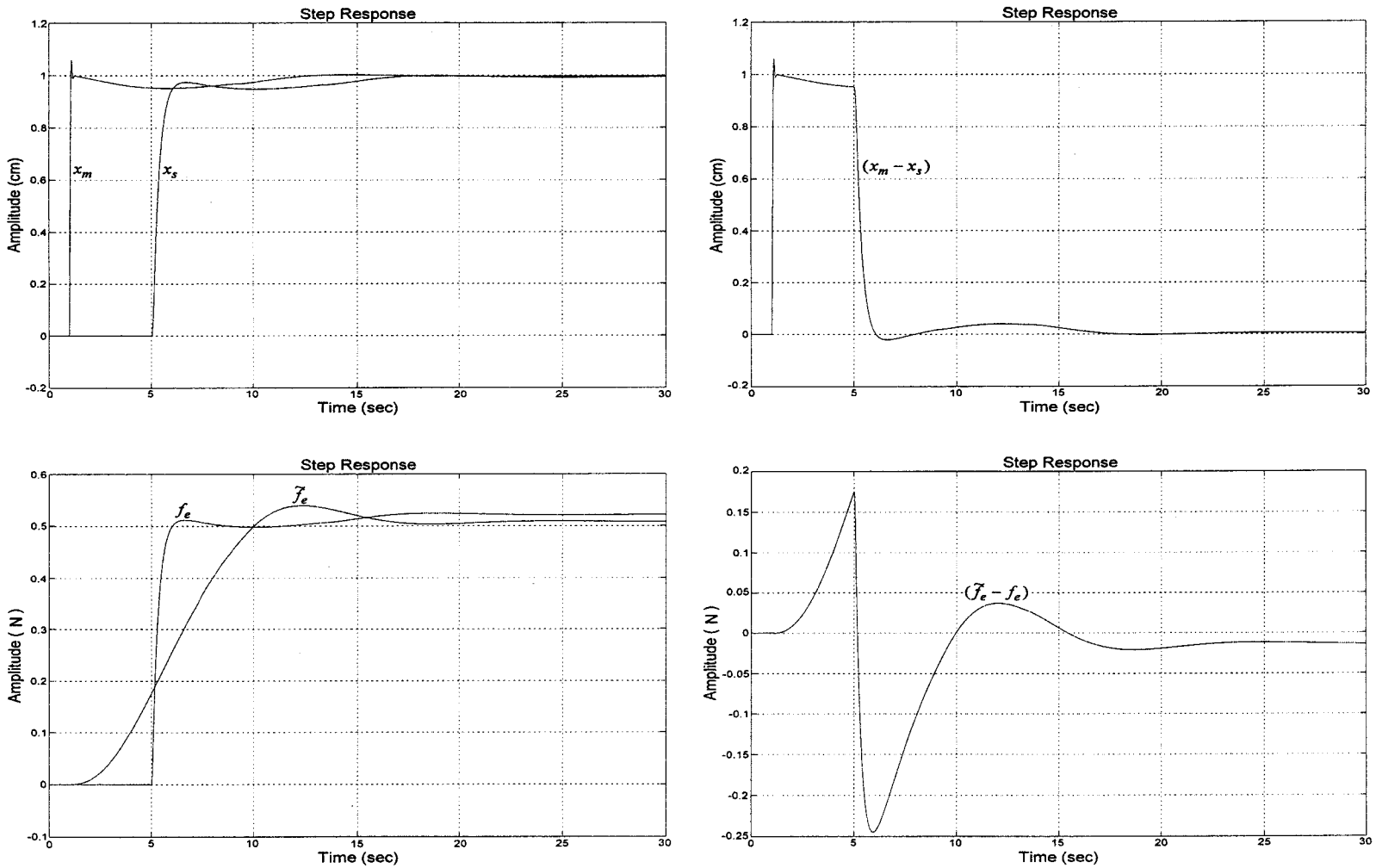


Fig. B.2 Case 2: $G_m = G_0(1+W_{mul})$, $G_s = G_0(1+W_{mul})$ **and** $G_e = B_0(1-db)s + K_0(1+dk)$

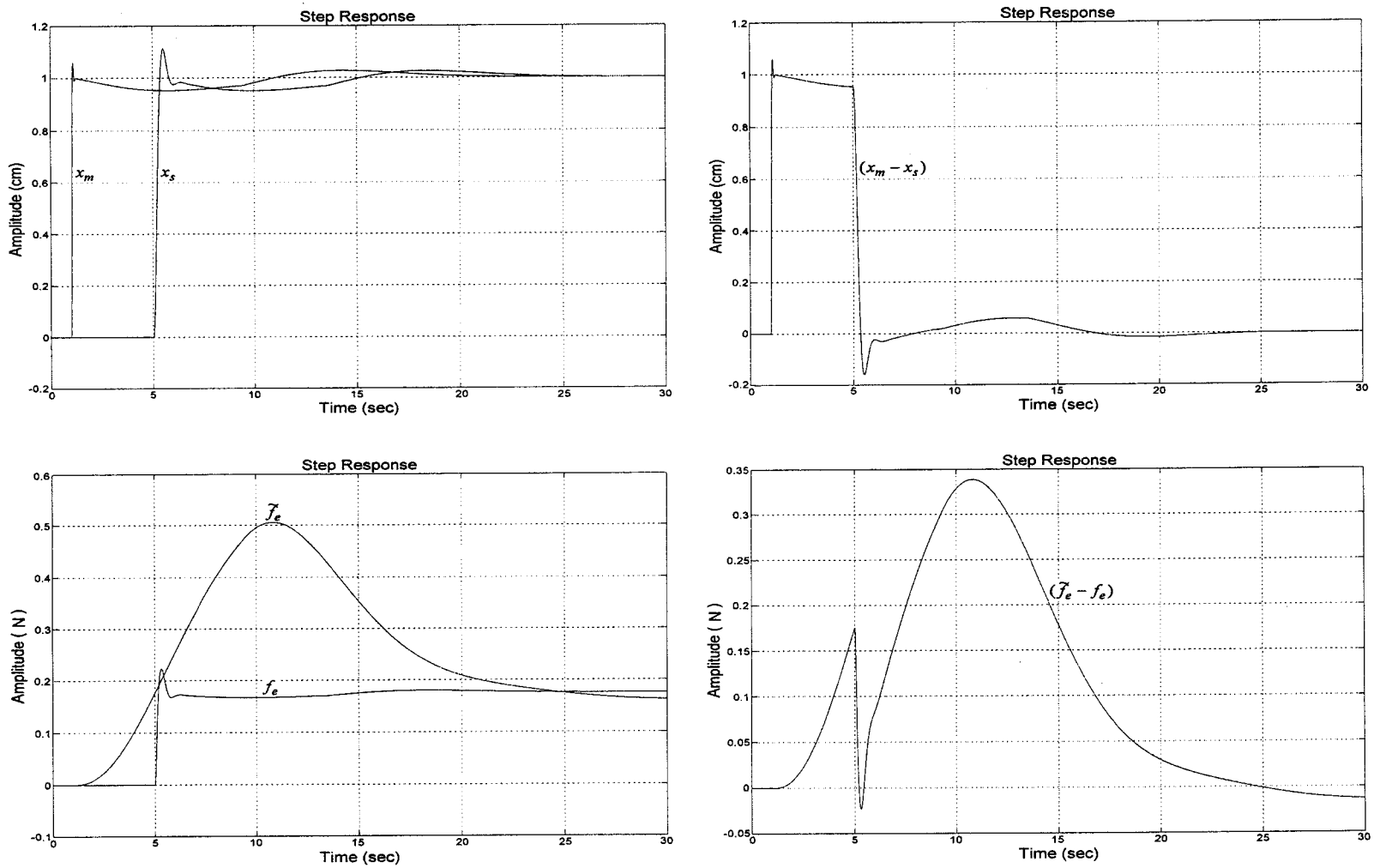


Fig. B.3 Case 3: $G_m = G_0(1+W_{mul})$, $G_s = G_0(1+W_{mul})$ **and** $G_e = B_0(1-db)s + K_0(1-dk)$

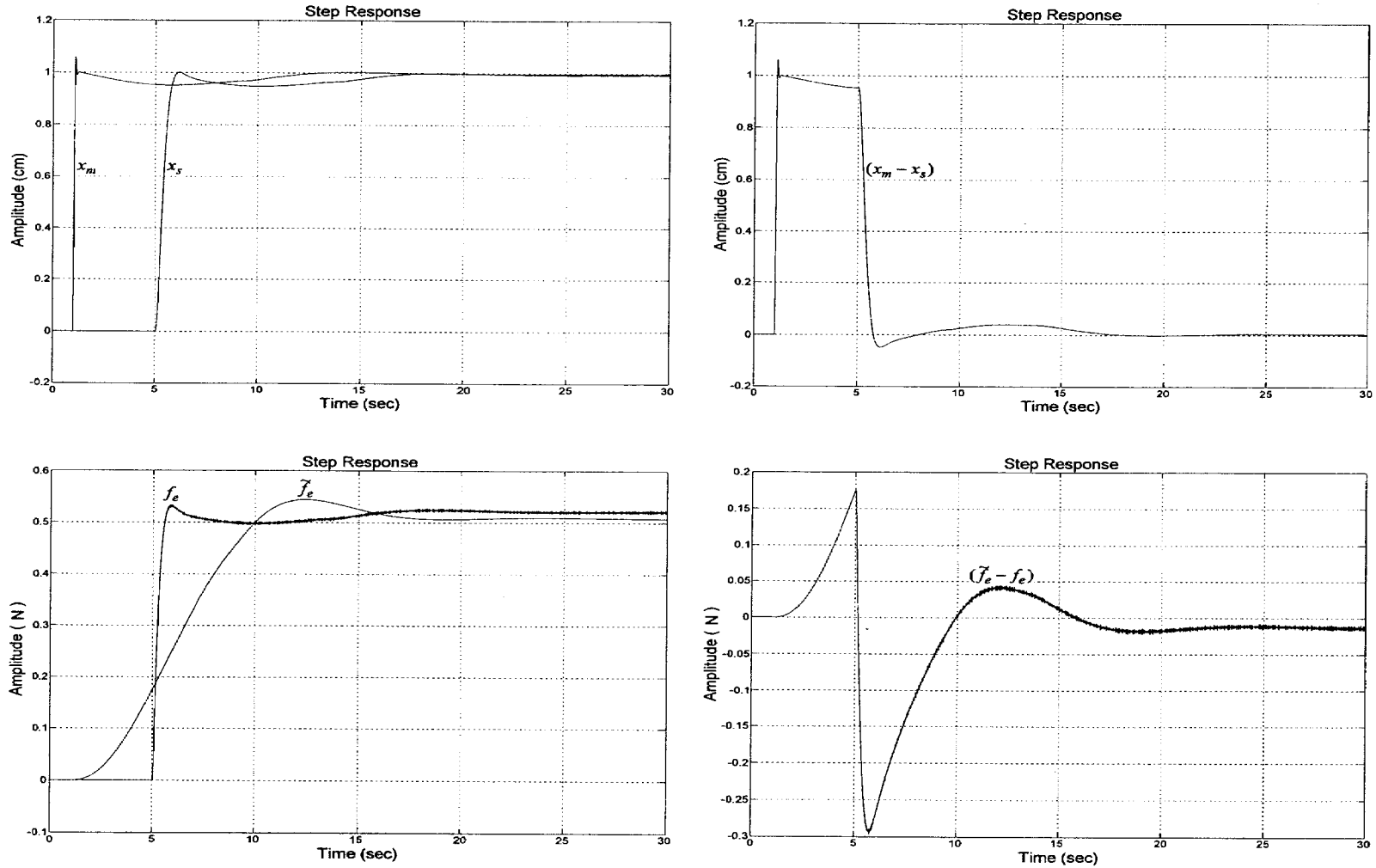


Fig. B.4 Case 4: $G_m = G_0(1+W_{mul})$, $G_s = G_0(1-W_{mul})$ **and** $G_e = B_0(1+db)s + K_0(1+dk)$

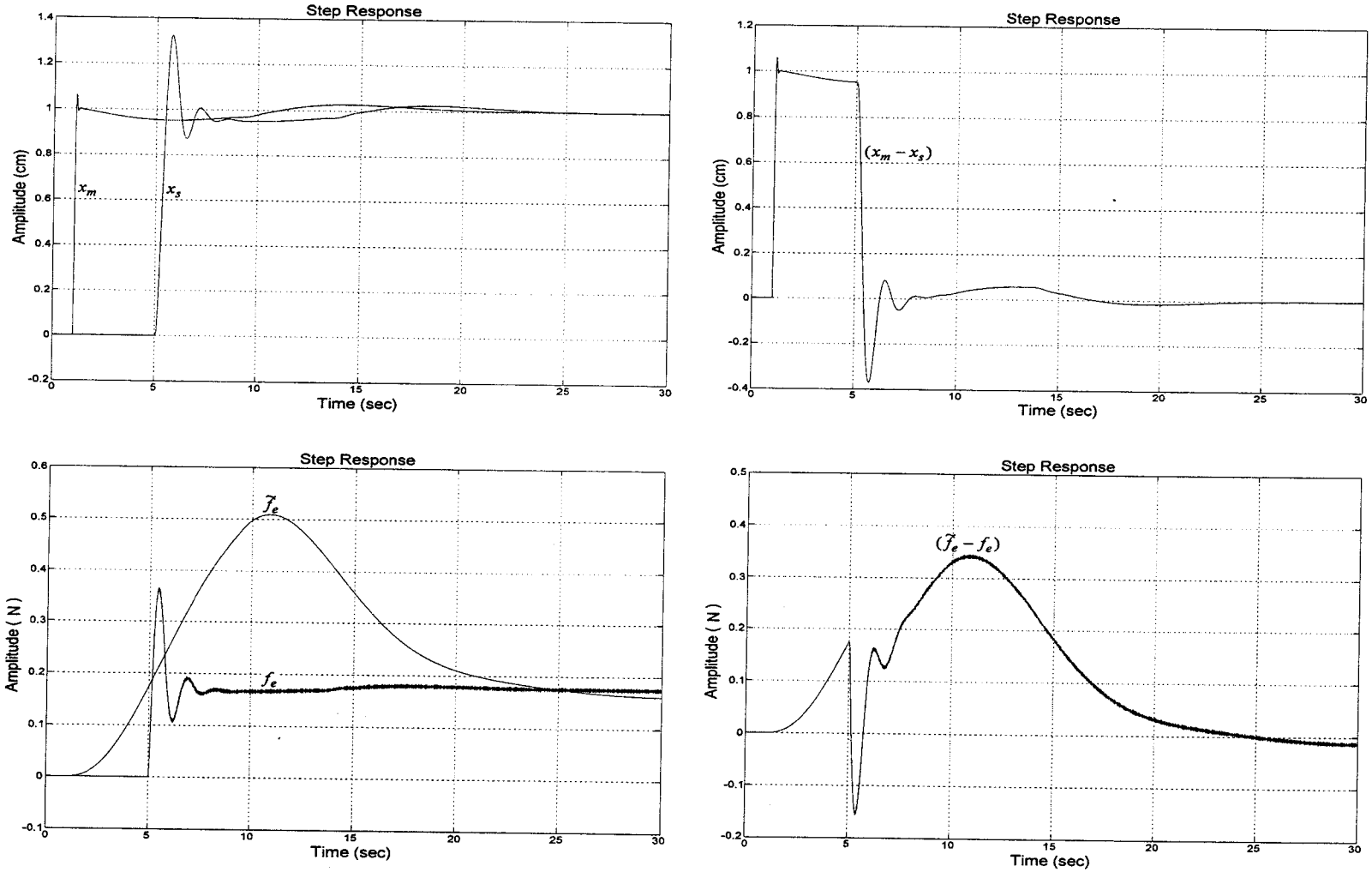


Fig. B.5 Case 5: $G_m = G_0(1+W_{mul})$, $G_s = G_0(1-W_{mul})$ **and** $G_e = B_0(1+db)s + K_0(1-dk)$

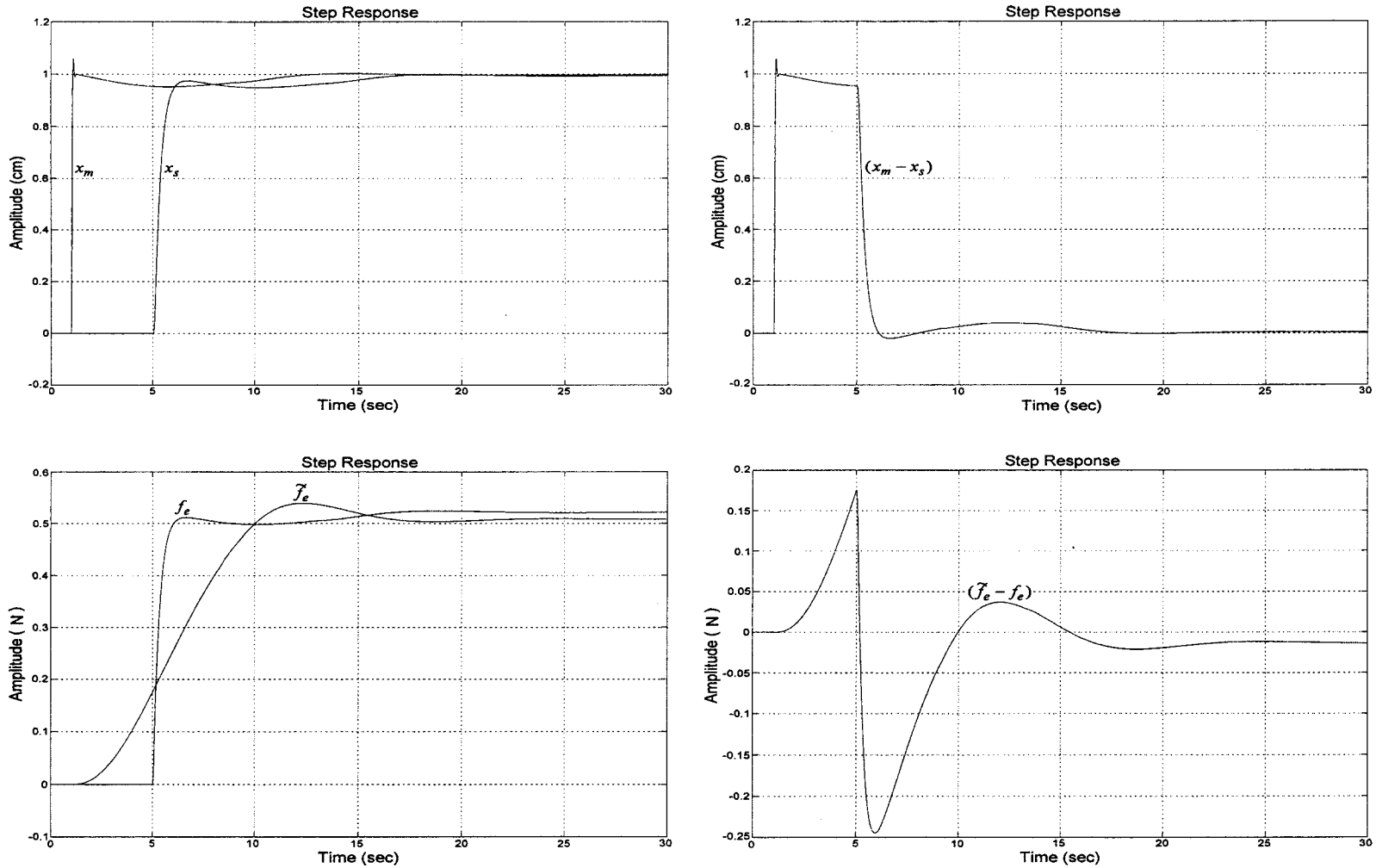


Fig. B.6 Case 6: $G_m = G_0(1+W_{mul})$, $G_s = G_0(1-W_{mul})$ **and** $G_e = B_0(1-db)s + K_0(1+dk)$

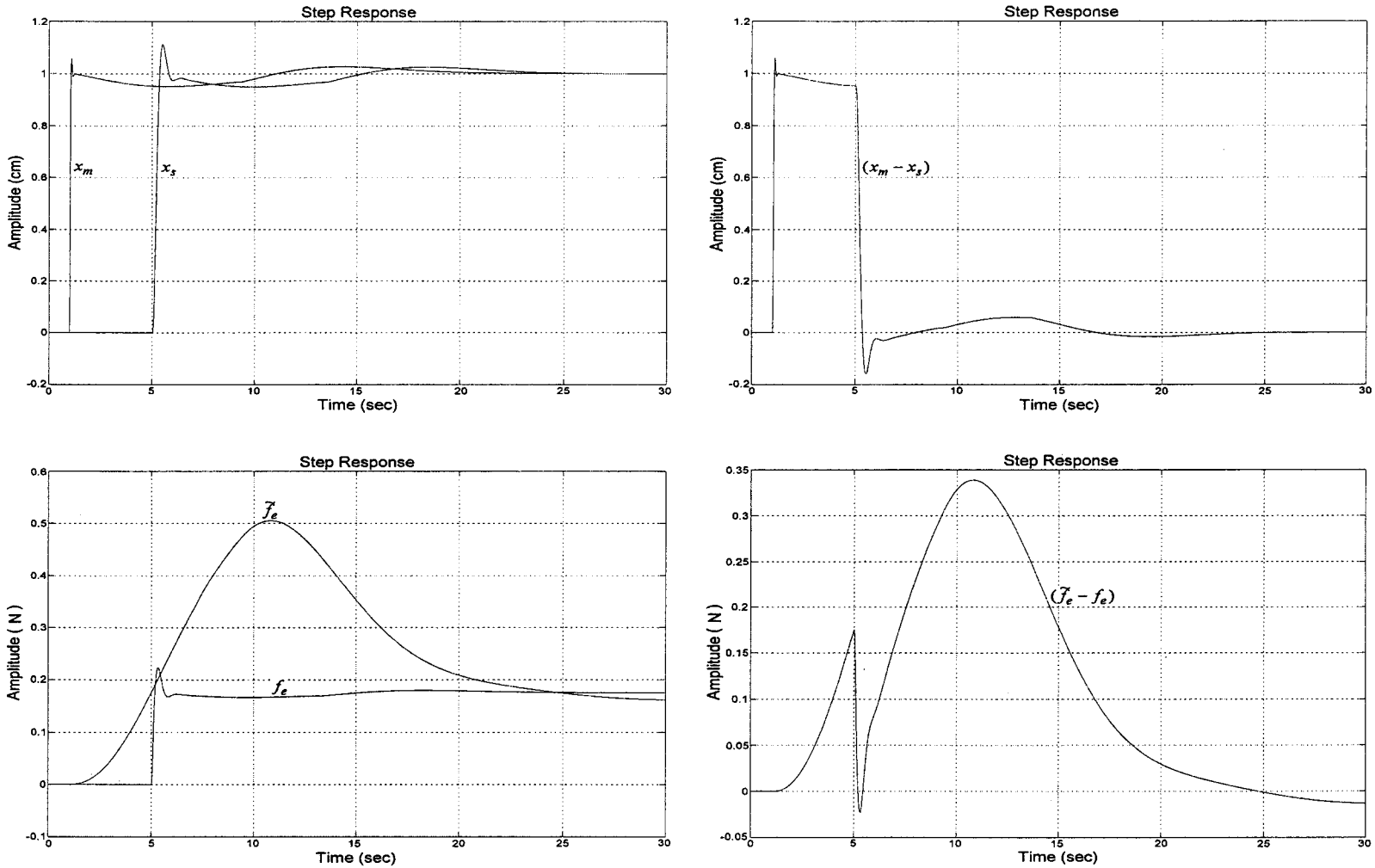


Fig. B.7 Case 7: $G_m = G_0(1+W_{mul})$, $G_s = G_0(1-W_{mul})$ **and** $G_e = B_0(1-db)s + K_0(1-dk)$

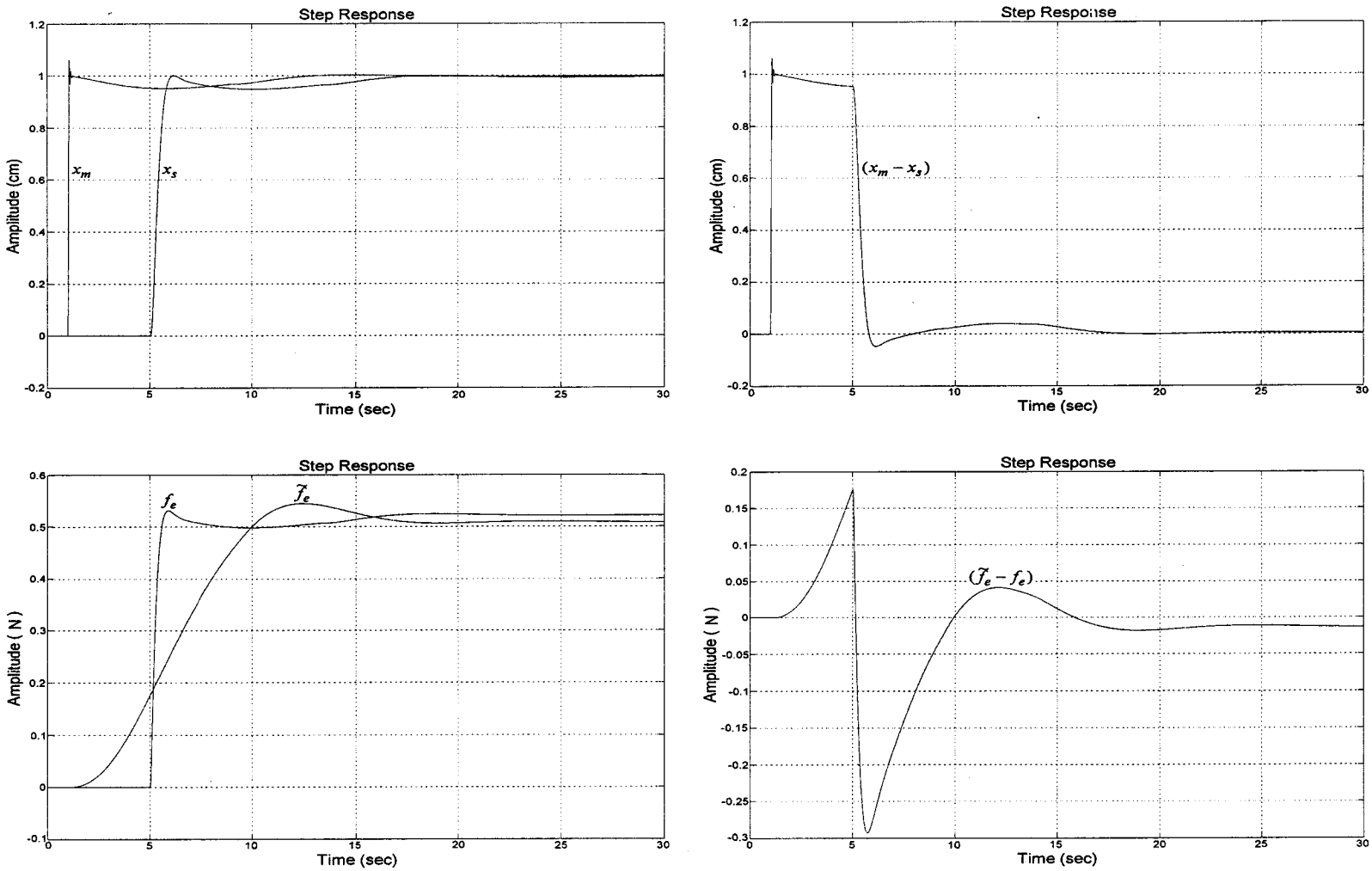


Fig. B.8 Case 8: $G_m = G_0(1-W_{mul})$, $G_s = G_0(1+W_{mul})$ **and** $G_e = B_0(1+db)s + K_0(1+dk)$

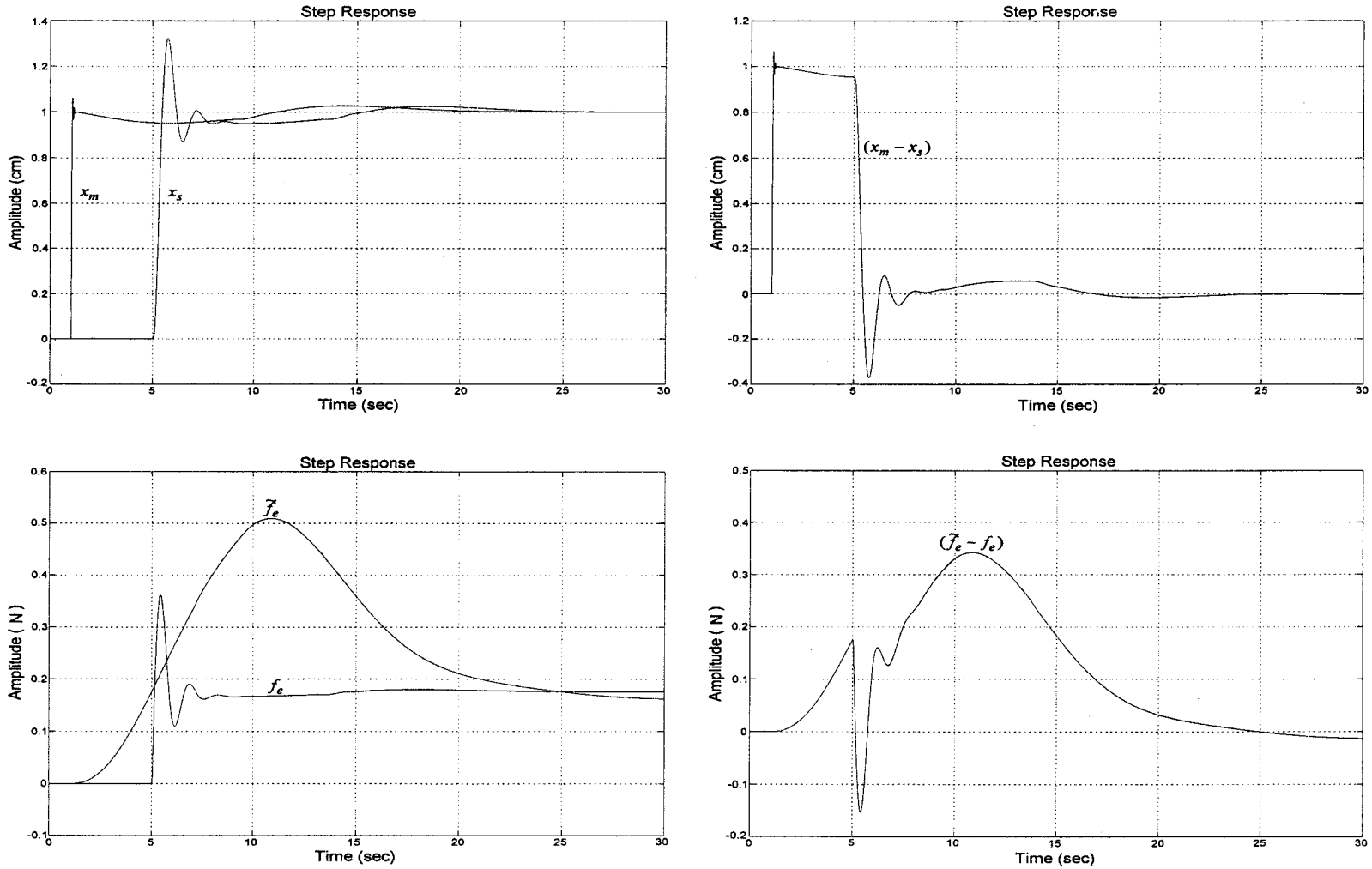


Fig. B.9 Case 9: $G_m = G_0(1-W_{mul})$, $G_s = G_0(1+W_{mul})$ **and** $G_e = B_0(1+db)s + K_0(1-dk)$

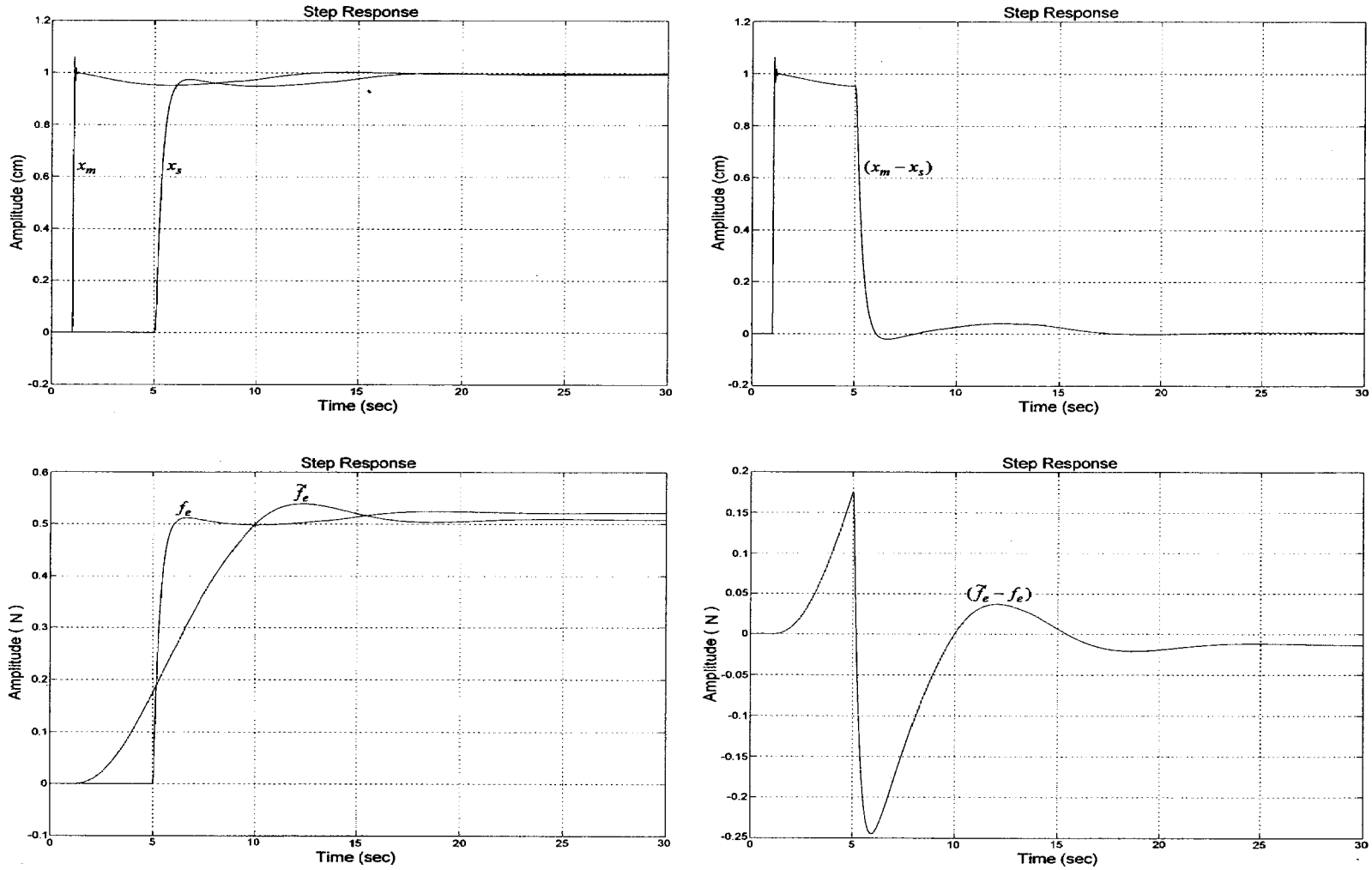


Fig. B.10 Case 10: $G_m = G_0(1 - W_{mul})$, $G_s = G_0(1 + W_{mul})$ **and** $G_e = B_0(1 - db)s + K_0(1 + dk)$

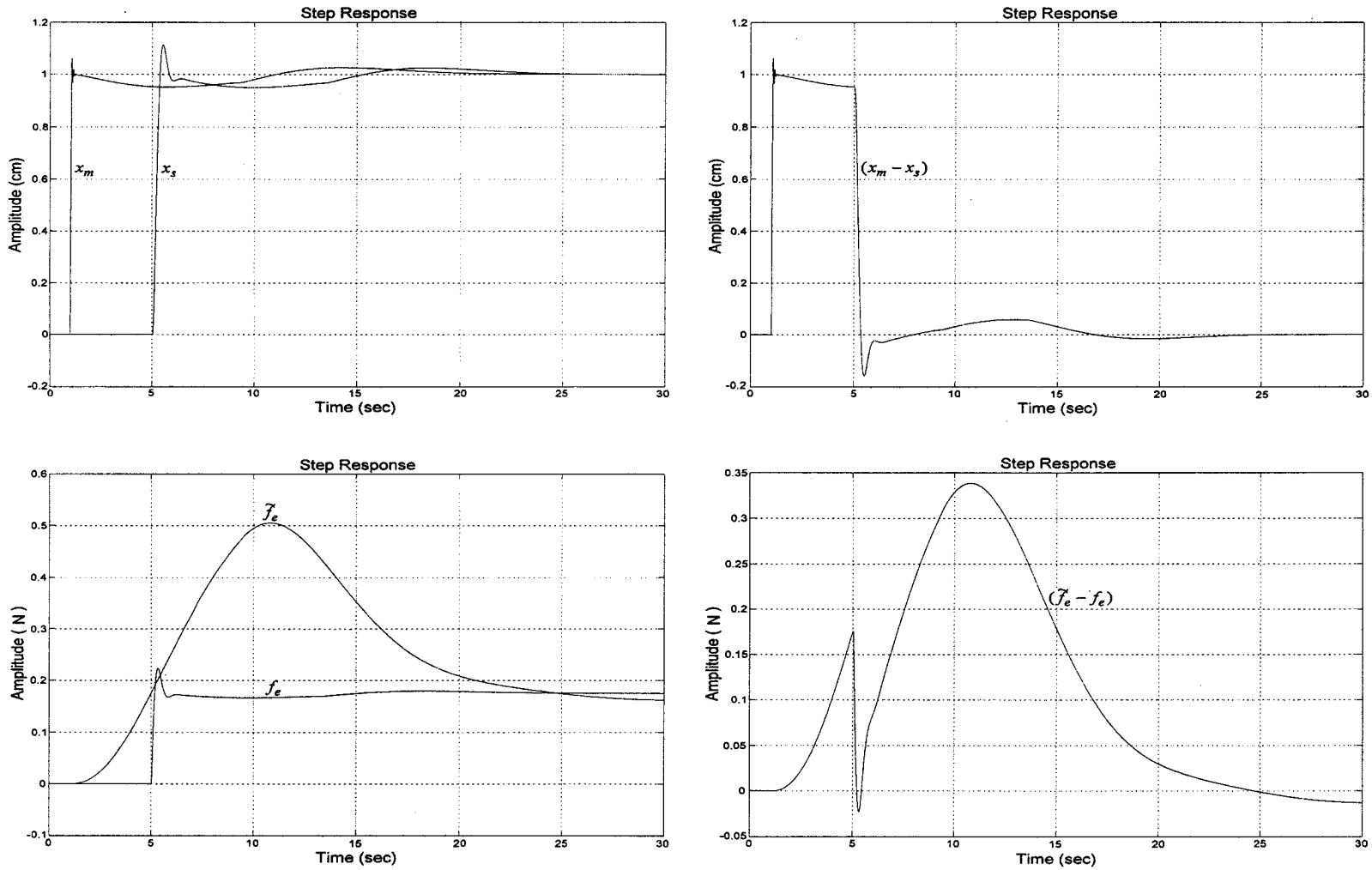


Fig. B.11 Case 11: $G_m = G_0(1-W_{mul})$, $G_s = G_0(1+W_{mul})$ **and** $G_e = B_0(1-db)s + K_0(1-dk)$

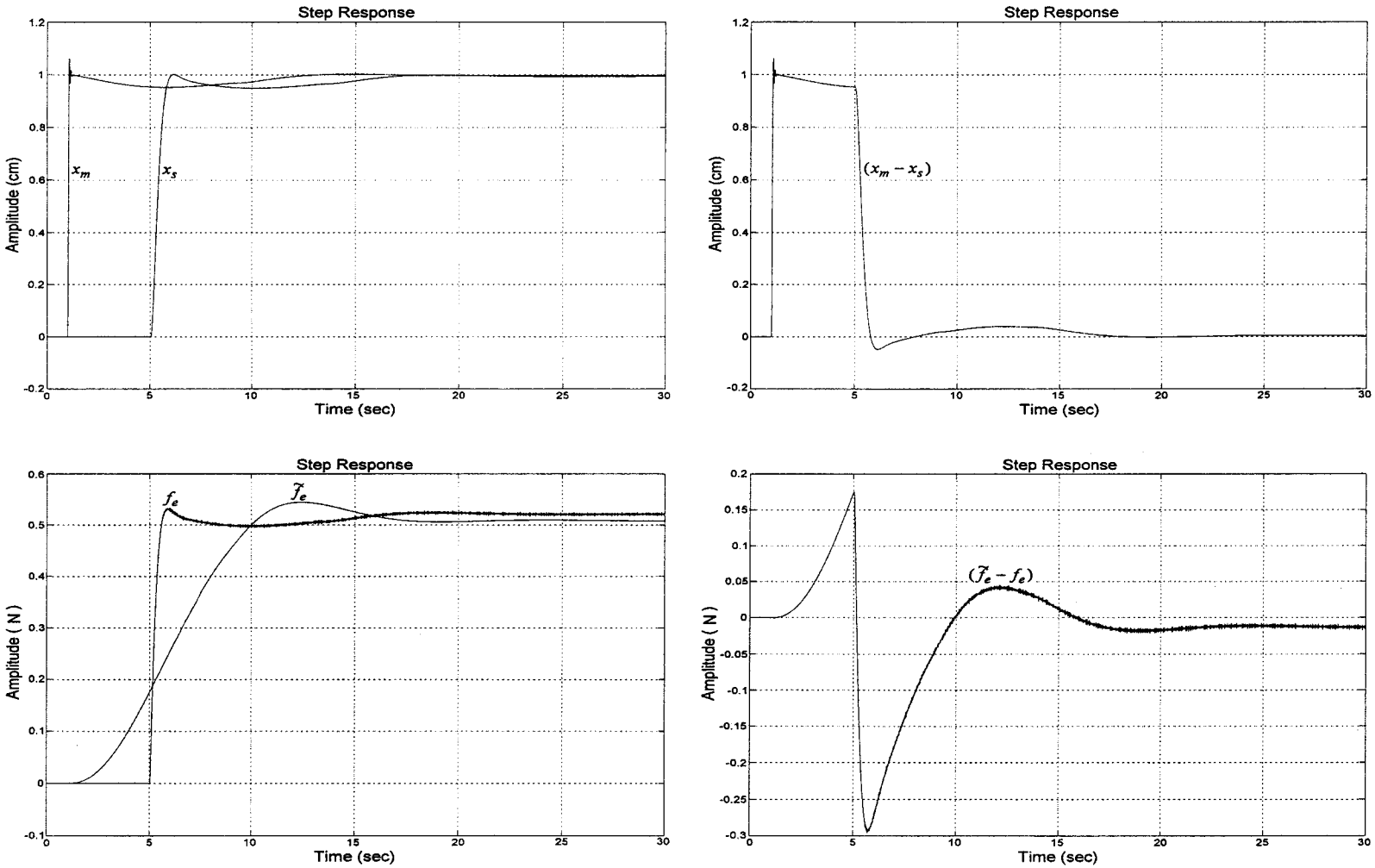


Fig. B.12 Case 12: $G_m = G_0(1-W_{mul})$, $G_s = G_0(1-W_{mul})$ **and** $G_e = B_0(1+db)s + K_0(1+dk)$

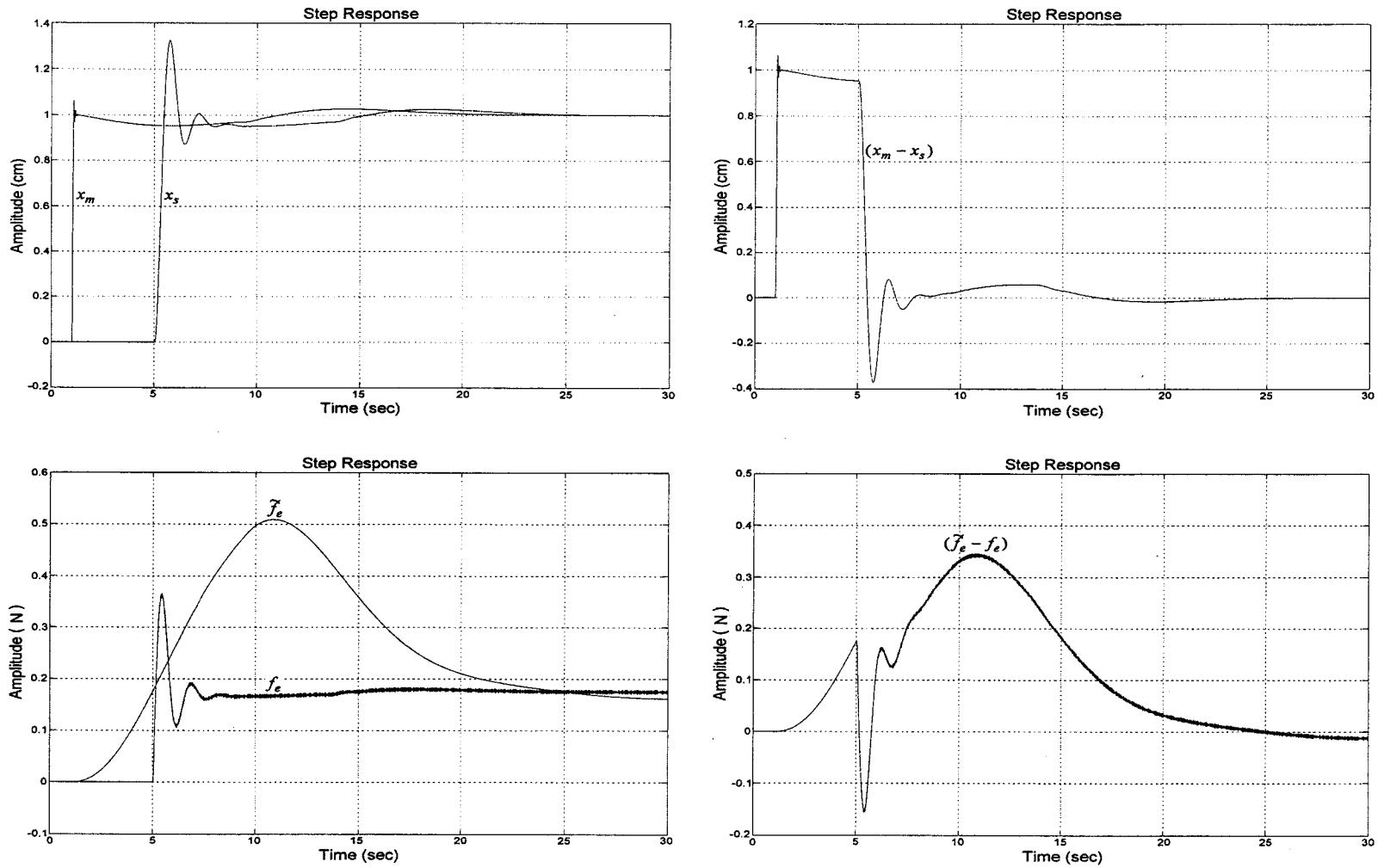


Fig. B.13 Case 13: $G_m = G_0(1-W_{mul})$, $G_s = G_0(1-W_{mul})$ **and** $G_e = B_0(1+db)s + K_0(1-dk)$

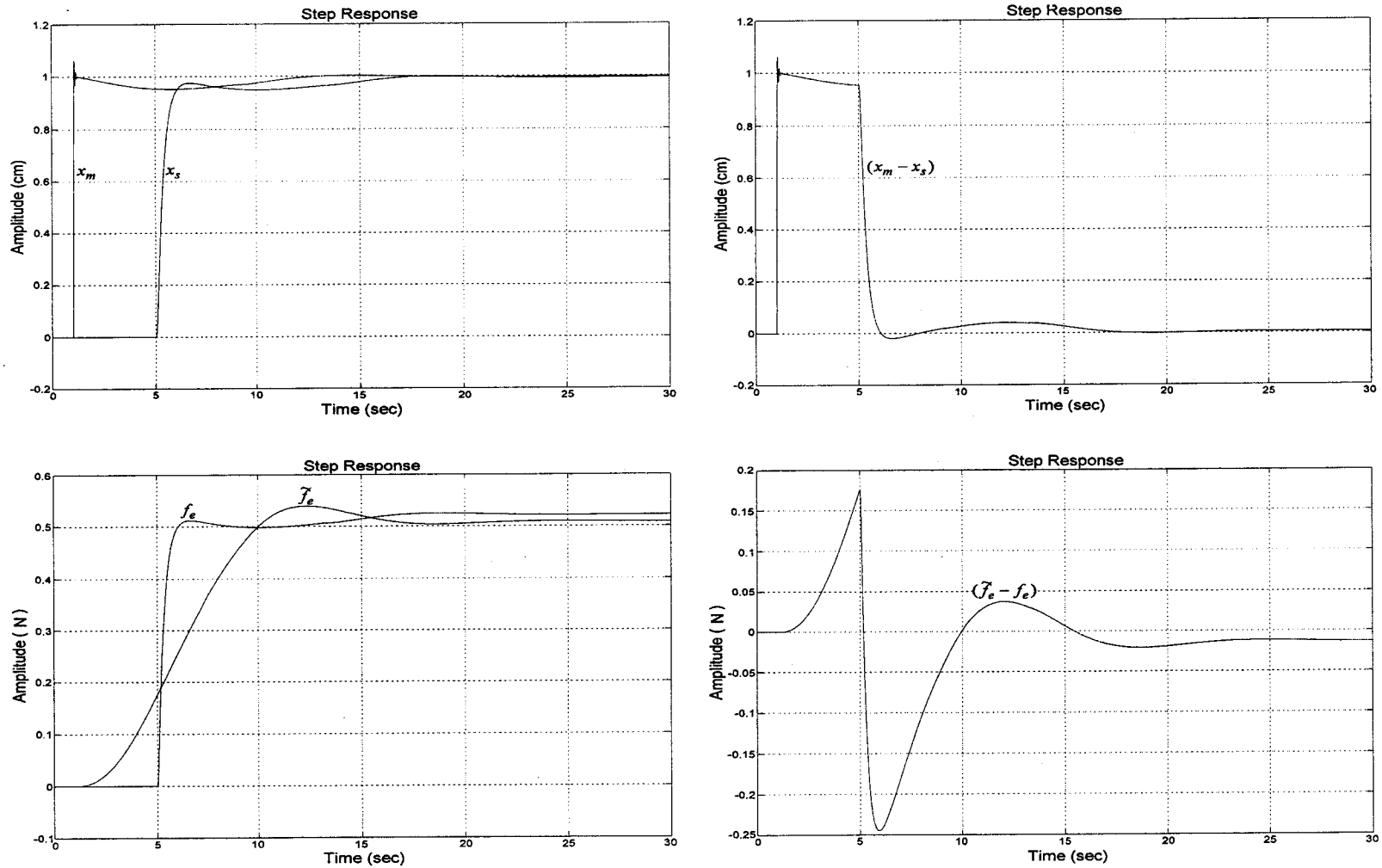


Fig. B.14 Case 14: $G_m = G_0(1-W_{mul})$, $G_s = G_0(1-W_{mul})$ **and** $G_e = B_0(1-db)s + K_0(1+dk)$