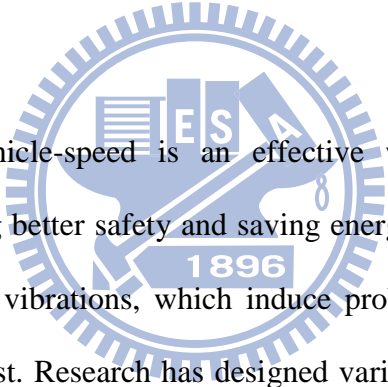# Chapter 1

INTRODUCTION

This dissertation is organized as follows: Chapter 1 introduction; Chapter 2 describes transformation of time domain specification into frequency domain bounds and Chapter 3 presents a QFT/H_∞ technique. Chapter 4 shows controller design of an MIMO system. Chapter 5 shows some simulation results. Finally, Chapter 6 conclusions and future works.

1.1   Motivation

Increasing motor vehicle-speed is an effective way to make the vehicle more competitive, while providing better safety and saving energy. However, a high-speed vehicle causes significant car body vibrations, which induce problems, such as ride stability, ride quality, and maintenance cost. Research has designed various vehicle suspension linking the bogies and the car body to cushion riders from vibrations, categorized as passive, active, and semi-active types. A passive vehicle suspension employing springs and pneumatic or oil dampers can only store energy in the spring and dissipate energy through the damper, fixing both components at the design stage. Replace the damper with a force actuator results in a fully active suspension. The idea behind fully active suspensions is that the force actuator is able to apply a force to the suspension in either bounce or rebound. The sophisticated control scheme employed in the suspension, actively governs this force, requiring high power. The current development of electronics and microprocessors has made commercial vehicles with active suspensions become available. Active suspension systems provide high control

performance over a wide frequency range of excitations induced by roadway irregularities beyond that control of passive suspensions. However, complexity, cost, and power consumption make these systems unacceptable for conventional use. Therefore, passive suspension systems remain dominant in the marketplace, because they are simple, reliable, and inexpensive.

The typical passive suspension system is similar to a spring with a damper placed at each corner of the vehicle. The spring is chosen based solely on the weight of the vehicle, while the damper is the component that defines the suspension's placement on the compromise curve. Depending on the realistic condition of the vehicle, a damper is chosen to make the vehicle perform best in its application. Ideally, the damper should isolate passengers from low-frequency road disturbances and absorb high-frequency road disturbances. Passengers are best isolated from low-frequency disturbances when the damping is high. However, high damping provides poor high frequency absorption. Conversely, when the damping is low, the damper offers sufficient high-frequency absorption, at the expense of low-frequency isolation.

Suspension is the term given to the system of springs, shock absorbers, and linkages that connect a vehicle to its wheels. Suspension systems serve a dual purpose contributing to the car's road holding/handling, braking for good active safety and driving pleasure, and keeping vehicle passenger comfortable and reasonably well isolated from road noise, bumps, and vibrations. These goals are generally at odds, so suspension tuning involves finding the right compromise. The suspension needs to keep the road wheel in contact with the road surface as much as possible, because all the forces acting on the vehicle do so through contact patches on the tires. The suspension protects the vehicle and any cargo or luggage from damage and wear. The design of the front and rear suspension of a car may be different.

Stability for the vehicle suspension system is a very important role. Vehicle systems may use a 4 or 5 link type suspension, multi link suspension, double wishbone type

suspension, rigid axle suspension, or parallel leaf spring type suspension. These suspension systems can effectively improve traffic safety and comfort. Generally, these two requirements conflict each other, therefore the spring and the damper must have a good trade-off design. Some have suggested that conventional passive suspension systems have reached the limits of their performance.

The passive suspension system consists of an energy-dissipating element, including the damper, and an energy-storing element, which is the spring. Since these two elements cannot add energy to the system, this kind of suspension system is passive. This restricts physically realizing the class of passive mechanical impedances. A further problem is that the suspension strut needs a small mass compared to that of the vehicle body and wheel hub, imposing further restrictions on practically realizing the class of mechanical impedances using the classical spring-mass-damper analogue.

Isolating the passenger and cargo from terrain induced shock and vibration is the important task of any ground vehicle suspension. Most suspensions have passive springs and dampers with limited vibration isolation performance for linear and nonlinear restoring or damping characteristics. Their transmissibility factors show that low damping gives good isolation at high frequency but poor resonance characteristics, whilst higher damping results in good resonance isolation at the expense of high frequency performance.

1.2   Classification of Vehicle Suspension System

Recent research efforts have focused on designing and developing an active suspension system to improve the trade-off between ride comfort and road handling. However, high cost has prevented its wide use, requiring a better cost effective suspension system. The semi-active suspension system has the potential to provide a better trade-off between cost and performance. The objective of this project is to develop an individually controlled semi-active

suspension system, which provides simultaneous real time optimum control of all the parameters including roll, pitch, and height control. An active suspension system provides a better trade-off between ride comfort and road handling.

Developing and correcting safe vehicle operation is one of the main trends in modern vehicle production. Manufacturers are increasingly installing advanced electro-mechanical and electronic systems to control vehicle dynamic performance, such as anti-lock braking systems (ABS), electronic break force distribution (EBD), and electronic stability program (ESP).

1.2.1 Passive suspension system

This is an ordinary suspension system used to control vertical motion dynamics of a vehicle. The suspension element does not supply energy to the system. However, it controls the relative motion of the body to the wheel by using different types of damping or energy dissipating elements. The shock absorber is the trade-off between minimizing body vertical acceleration and tire deflection. The typical damping rate for shock absorbers is 24 inch/sec for suspension compression and 5 inch/sec for suspension extension.

1.2.2 Semi-active suspension system

A semi-active suspension [1, 2, 3, 4, 5] provides a rapid change in the rate of spring damping coefficients, and does not provide any energy into the suspension system. This system consists of sensors and actuators to detect the road profile for control input. Hydraulic actuators located in shock absorbers control the damping force rate. The shock absorber controls vehicle altitude changes when steering and accelerating, and the damping force transmitted from an uneven road.

1.2.3 Active suspension system

The active suspension system [6, 7, 8] responds to vertical changes in the road input. This system supplies the energy to system elements, causing relative motion between the vehicle body and wheels. The active suspension system incorporates sensors to measure vehicle parameters, such as vertical vehicle body acceleration and/or vertical wheel acceleration, and uses a force generator in place of the damper, and in some cases, in place of the spring.

A vehicle suspension system consists of a spring and a damper, to improve ride quality and road holding properties. These two requirements conflict with each other. The spring and the damper design must allow for good system performance. To improve suspension properties, many studies have dealt with active suspension systems. Investigations into active suspensions for ground vehicles have increased in recent years because using the active suspension system improves ride comfort.
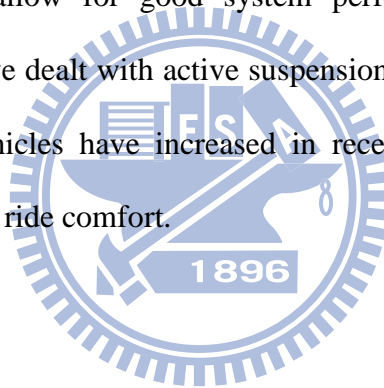
Chapter 2

TRANSLATION OF TIME-DOMAIN BOUNDS TO FREQUENCY-DOMAIN BOUNDS IN QFT DESIGN

2.1 Introduction of QFT

The Quantitative Feedback Theory [9, 10, 11] stems from Horowitz [12]. It is an efficient frequency based robust controller design methodology that maintains system response within pre-specified tolerances despite uncertainties and disturbances.

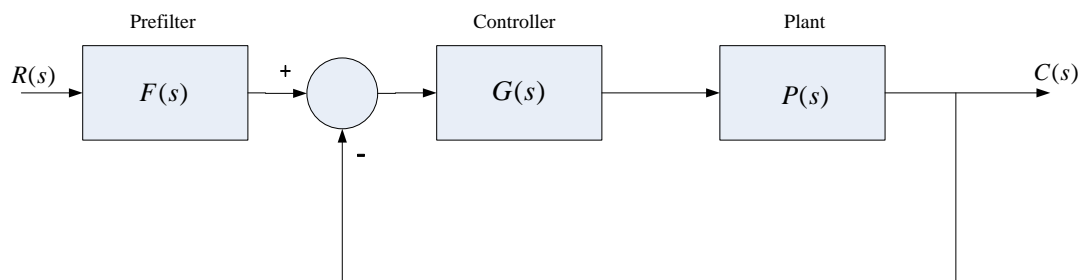A two- degree-of-freedom (TDOF) control system structure is typically assumed for the QFT technique in Figure 2-1.

Figure 2-1: Two-degree-of-freedom feedback structure

There are five principle steps in QFT design procedure:

(1) Translation of time-domain bounds on $C(t)$ into bounds on $|T(j\omega)| \equiv |C(j\omega)/R(j\omega)|$ of Figure 2-2.

(2) Derivation of bounds on $L(j\omega)$ from the bounds on $|T(j\omega)|$ and on $P(j\omega)$.

(3) Formulation of the optimum $L(s)$ from the results of Step 2.

(4) Derivation of the pre-filter $F(s)$ of Figure 2-1.

(5) Modification, if necessary, of $L(s)$ and $F(s)$.

2.2 Statement of Specifications

In a minimum-phase system the magnitude of frequency response $|T(j\omega)|$ completely specifies the transfer function $T(s)$, which in turn uniquely determines the system step response $C(t)$. But the rigorous translation of time-domain bounds into bounds on $|T(j\omega)|$ is, as yet, an unsolved problem. In practice, one may begin, for example, by assuming a simple second-order or third-order system model for $T(s)$[12]. In this dissertation we focus on translation of time-domain tolerances into frequency-domain bounds by (3,0) model.

Since QFT design is based on frequency-domain, a set of time-domain design specifications will be transferred to frequency-domain specifications first.

We assume the system is two-degree-of-freedom (TDOF) with minimum phase, giving time-domain design specifications as shown in Figure 2-2 we will transfer it to the frequency-domain as shown in Figure 2-3.

Time-domain specifications [13] will be defined as follows:
1. 10% of rise time

   The time of output reaches 10% of steady state value. Upper bound specification is denoted as $t_{r10}(1)$; lower bound specification is denoted as $t_{r10}(2)$.

2. 90% of rise time

   The time of output reaches 90% of steady state value. Upper bound specification is

denoted as $t_{r90}(1)$; lower bound specification is denoted as $t_{r90}(2)$.

3. 5% of settling time

The settling time is the time required for the response curve to reach and stay within a range about the final value of size specified by absolute percentage of the final value. In this paper choose 5%. i.e., when $t > t_{s5}$, $|c(t)-1| \leq 0.05$ steady state value.

4. Maximum Overshoot, M_P

The maximum overshoot is the maximum peak value of the response curve measured from unity. It is defined as

$$M_P = \frac{c(t_p)-c(\infty)}{c(\infty)} \times 100\% \ ,$$

where t_p is the time required for the response to reach the first peak of the overshoot.
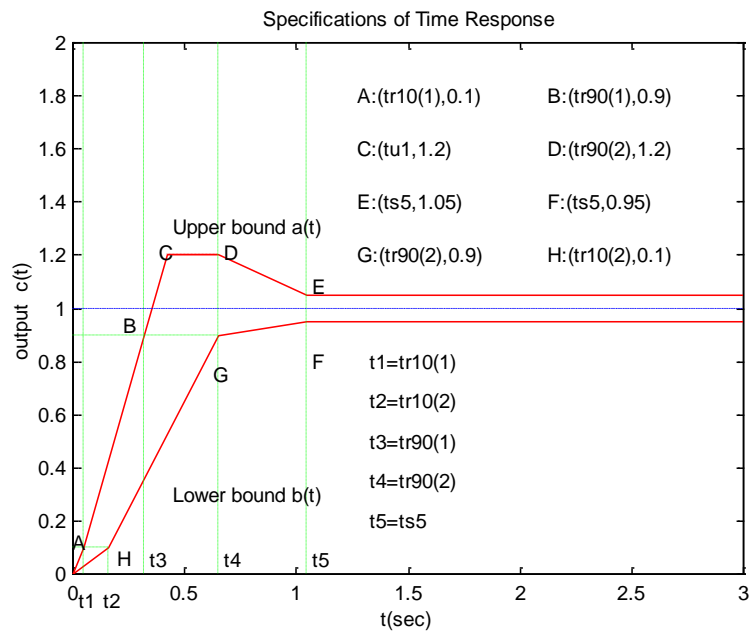
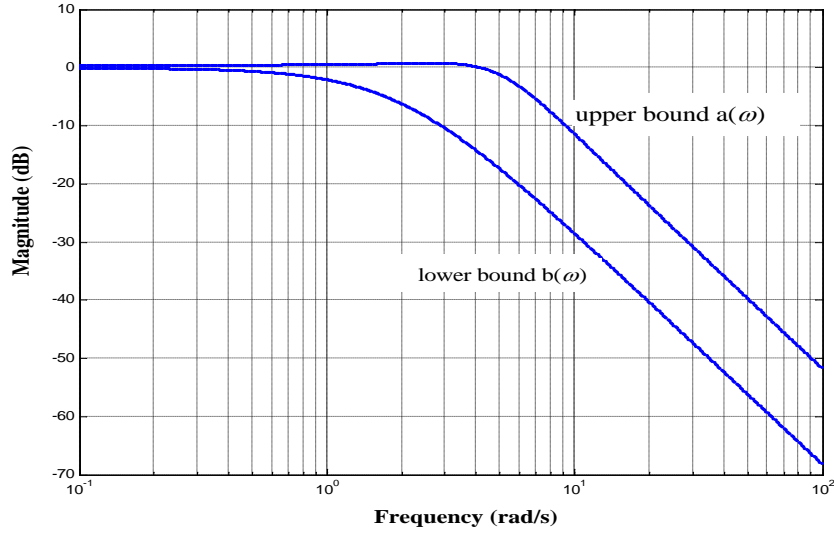**Figure 2-2** Specifications of time response

Figure 2-3 Boundaries of frequency response

2.3 Mathematical Model of the Proposed Transformation

In the third-order all poles (3,0) system, the system transfer function is $T(S)$

$$T(s) = C(s)/R(s) = p\omega_n^2 /[(s+p)(s^2 + 2\xi\omega_n s + \omega_n^2)].$$ (2.1)

If $R(s) = 1/s$, then

$$C(s) = T(s)R(s) = p\omega_n^2 /[s(s+p)(s^2 + 2\xi\omega_n s + \omega_n^2)].$$ (2.2)

In Eq. (2.1) different values of pole (p) and damping ratio (ξ) will be discussed in the following:

1. Under damping system $0 < \xi < 1$

   Taking inverse Laplace transformation of Eq. (2.2), we have

   $$
   \begin{aligned}
   c(t) = {} & 1 - \omega_n^2 e^{-pt} /(p^2 - 2\xi\omega_n p + \omega_n^2) \\
   & + p e^{-\xi\omega_n t}(\sqrt{1-\xi^2}\sqrt{p^2 - 2\xi\omega_n p + \omega_n^2}) \\
   & \sin(\omega_n\sqrt{1-\xi^2}\, t - \phi), \qquad \text{where}
   \end{aligned}
   $$ (2.3)

   $$\phi = \tan^{-1}(\sqrt{1-\xi^2}/-\xi) + \tan^{-1}[\omega_n\sqrt{1-\xi^2}/(p-\xi\omega_n)].$$ (2.4)

9

2. Critically damping system，$\xi=1$

Rewriting Eq. (2.2), we have

$$
\begin{aligned}
C(s) &= \frac{p\omega_n^2}{s(s+p)(s+\omega_n)^2} \\
&= \frac{1}{s} - \frac{\omega_n^2/(\omega_n-p)^2}{s+p} - \frac{p(p-2\omega_n)/(\omega_n-p)^2}{s+\omega_n} \\
&\quad - \frac{p\omega_n/(p-\omega_n)}{(s+\omega_n)^2}.
\end{aligned}
\tag{2.5}
$$

Taking inverse Laplace transformation of Eq. (2.5), yields

$$
\begin{aligned}
c(t) &= 1 - \omega_n^2 e^{-pt}/(\omega_n-p)^2 \\
&\quad - p(p-2\omega_n)e^{-\omega_n t}/(\omega_n-p)^2 \\
&\quad - p\omega_n t e^{-\omega_n t}/(p-\omega_n).
\end{aligned}
\tag{2.6}
$$

3. Over damping system, $\xi>1$

In this case, Eq. (2.2) will be rewritten in the form

$$
\begin{aligned}
C(s) &= \frac{p\omega_n^2}{s(s+p)(s+A)(s+B)} \\
&= \frac{1}{s} - \frac{\omega_n^2}{(p-A)(p-B)(s+p)} \\
&\quad - \frac{p\omega_n^2}{A(A-p)(A-B)(s+A)} \\
&\quad - \frac{p\omega_n^2}{B(B-p)(B-A)(s+B)}.
\end{aligned}
\tag{2.7}
$$

Taking inverse Laplace transformation of Eq. (2.7), we have

$$
c(t) = 1 - k_1 e^{-pt} - k_2 e^{-At} - k_3 e^{-Bt} \quad ,
\tag{2.8}
$$

where

$$
\begin{aligned}
A &= \omega_n(\xi - \sqrt{\xi^2-1}) \\
B &= \omega_n(\xi + \sqrt{\xi^2-1}) \\
k_1 &= \omega_n^2/[(p-A)(p-B)] \\
k_2 &= (p/A)\omega_n^2/[(A-p)(A-B)] \\
k_3 &= (p/B)\omega_n^2/[(B-p)(B-A)].
\end{aligned}
$$

2.4 Flow Charts and Computing Procedures

In this section we will discuss how to transfer the time-domain tolerances into the equivalent frequency response tolerances.

Given $t_{r10}(1), t_{r90}(1), t_{r10}(2), t_{r90}(2), t_{s5}$, five time- domain specifications the corresponding undamped natural frequencies are denoted by $\omega_n(t_{r10}(1)), \omega_n(t_{r90}(1)), \omega_n(t_{r10}(2)),$ $\omega_n(t_{r90}(2)), \omega_n(t_{s5})$.

Adding two constrains $c(t) < m_p$ and $c(t_{s5}) < 1.05$, we have seven time specifications for the transformation. The flow charts are shown in Figure 2-4(a) to Figure 2-4(c).

The upper bound of undamped natural frequency $\boldsymbol{\omega_n}$ ($\omega_{\max}$) can be found by time specifications $t_{r10}(1)$ and $t_{r90}(1)$, which will be denoted by

$$\omega_{\max} = Min(\omega_n(t_{r10}(1)), \omega_n(t_{r90}(1))). \tag{2.9}$$

The lower bound of undamped natural frequency ω_n ($\omega_{\min}$) can be found by time specifications $t_{r10}(2)$, $t_{r90}(2)$, $and\, t_{s5}$, which will be denoted by

$$\omega_{\min} = Max(\omega_n(t_{r10}(2)), \omega_n(t_{r90}(2)), \omega_n(t_{s5})). \tag{2.10}$$

If $\omega_{\max} > \omega_{\min}$ then the region of undamped natural frequency of the given $t_{r10}(1), t_{r90}(1), t_{r10}(2), t_{r90}(2), t_{s5}$ five time-domain specifications can be transferred to the frequency specifications. Otherwise, it cannot be transferred. Furthermore, adding two constrains $c(t) < m_p$ and $c(t_{s5}) < 1.05$, we can determine the overall transformation region.

2.5 Illustrating Examples

In the following, we will give two examples to show how to transfer time-domain specifications to frequency -domain specifications [14].

Example 1：

Time-domain specifications are

$t_{r10}(1) = 0.05,\ t_{r10}(2) = 0.16,\ t_{r90}(1) = 0.219,$
$t_{r90}(2) = 0.65, t_{s5} = 1.047$ and $m_p = 1.2.$

The transformation results are shown in Figure 2-5 to Figure 2-10.

Example2:

Time-domain specifications are

$t_{r10}(1) = 0.079, t_{r10}(2) = 0.171,\ t_{r90}(1) = 0.341,$
$t_{r90}(2) = 1.158, t_{s5} = 1.513$ and $m_p = 1.2.$

The transformation results are shown in Figure 2-11 to Figure 2-16.

In the third-order model with all poles (3, 0) system, example 1, when we choose real pole much larger than the undamped natural frequency, the result is equivalent to the second-order model with all poles (2, 0) system.

Comparing Figure 2-8 with Figure 2-10, we find that the frequency-domain tolerances of third-order model with all poles (3, 0) system are wider than the second-order with all poles (2, 0) system, which will be easier to design robust controllers.
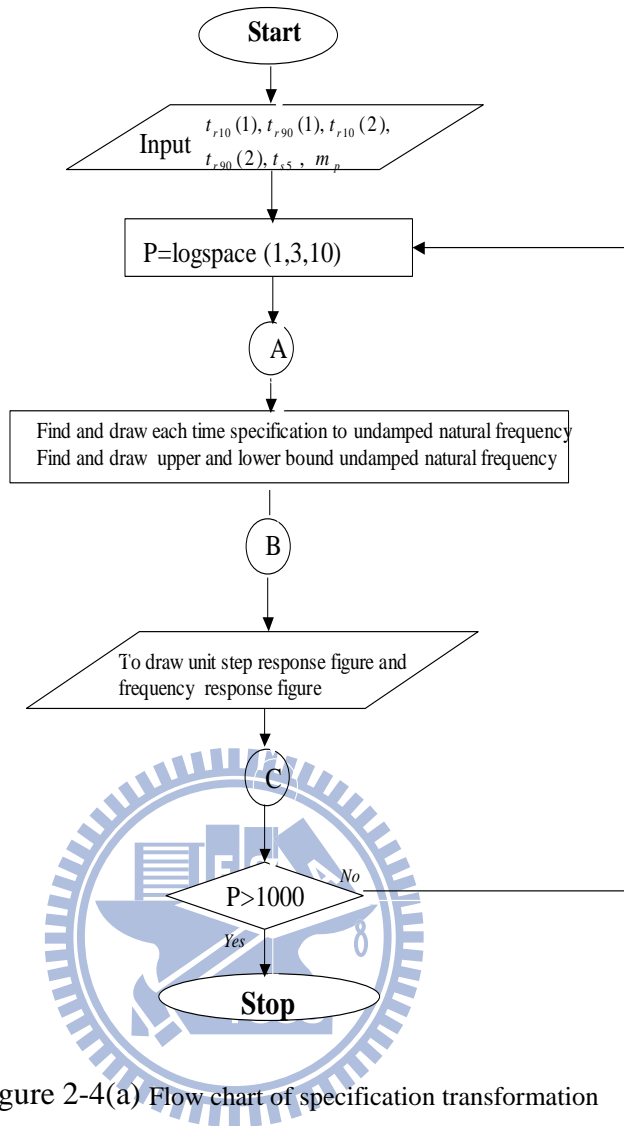
Start

Input $t_{r10}(1)$, $t_{r90}(1)$, $t_{r10}(2)$, $t_{r90}(2)$, t_{s5}, m_p

P=logspace (1,3,10)

A

Find and draw each time specification to undamped natural frequency
Find and draw upper and lower bound undamped natural frequency

B

To draw unit step response figure and frequency response figure

C

P>1000

No

Yes

Stop

Figure 2-4(a) Flow chart of specification transformation

A

$$\xi = 0.05 : 0.05 : 3.00$$

$Find \ \& \ Save \ \omega_n(t_{r10}(1)), \omega_n(t_{r90}(1)),$
$\omega_n(t_{r10}(2)), \omega_n(t_{r90}(2), \omega_n(t_{s5}),$

$\xi \geq 1$ — No

Yes

$\xi = 1, \omega_n < p$ or
$\xi > 1, \omega_n < p/(\xi + \sqrt{\xi^2 - 1})$

No

Yes

$\xi > 3$

Yes

$Search \quad \omega_n(t_{r10}(1)), \omega_n(t_{r90}(1))$
$\omega_n(t_{r10}(2)), \omega_n(t_{r90}(2)), \omega_n(t_{s5})$

To draw damping ratio vs
undamped nature frequency figure

$Find \ \xi_a = Min[length(\omega_n(t_{r10}(1)) \quad \omega_n(t_{r90}(1)))]$
$\xi_b = Min[length(\omega_n(t_{r10}(2)) \quad \omega_n(t_{r90}(2)) \quad \omega_n(t_{s5}))]$

$\omega_{max} = Min(\omega_n(t_{r10}(1), \omega_n(t_{r90}(1)))$
$\omega_{min} = Max(\omega_n(t_{r10}(2), \omega_n(t_{r90}(2)), \omega_n(t_{s5}))$

To draw upper and lower bound figure of
undamped nature frequency

B
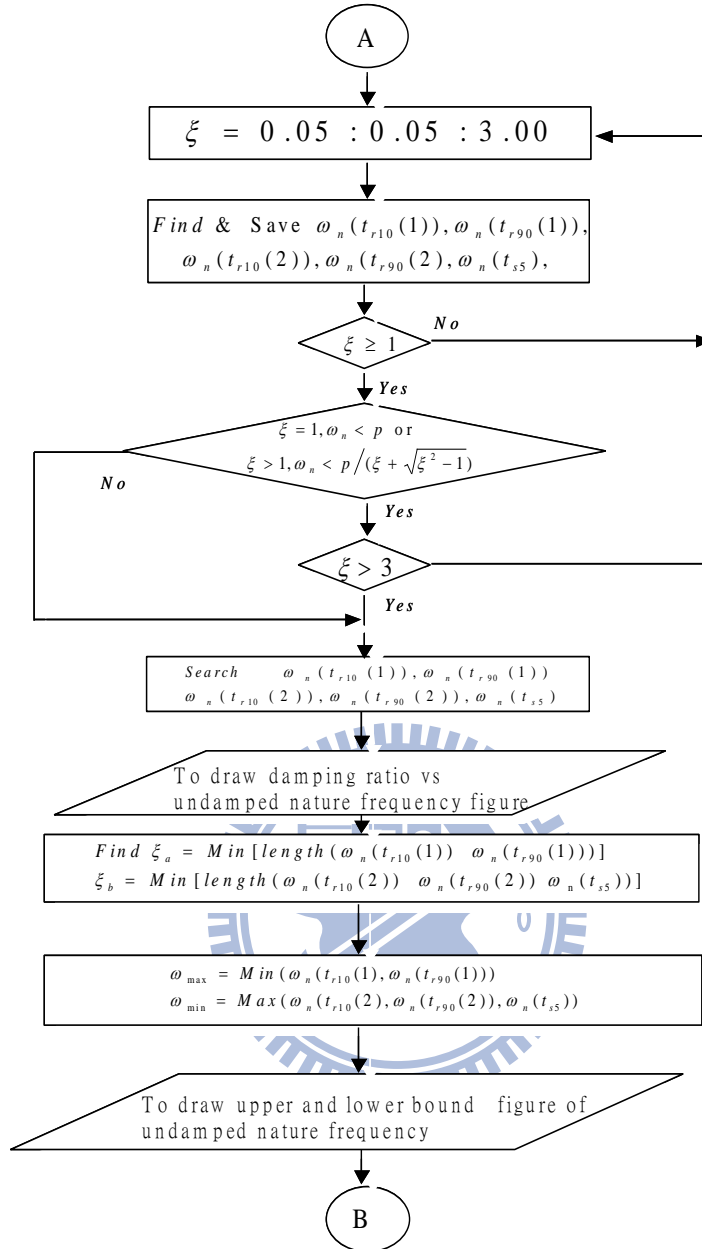
Figure 2-4(b) Flow chart of specification transformation
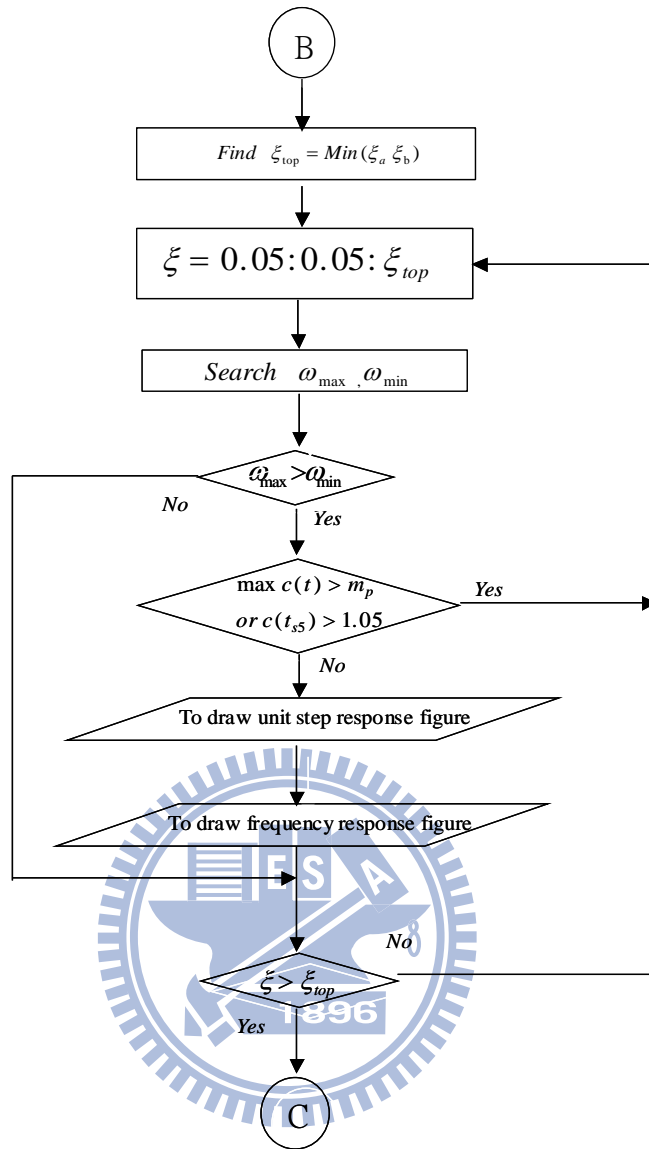
Figure 2-4(c) Flow chart of specification transformation
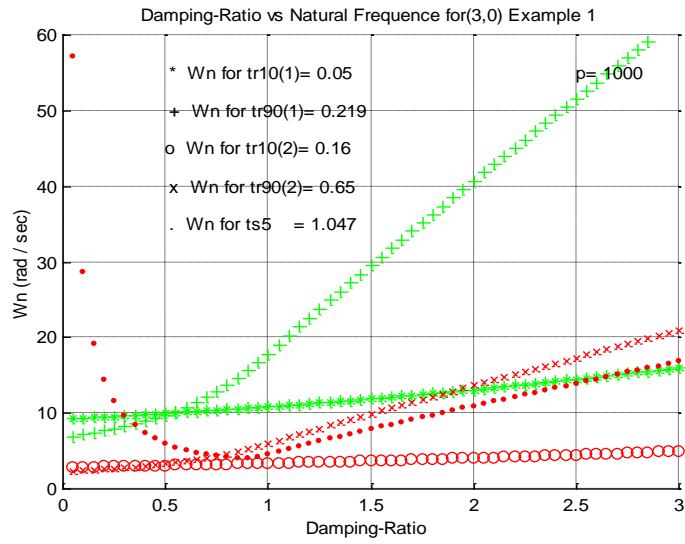
Figure 2-5: Boundary of undamped natural frequency
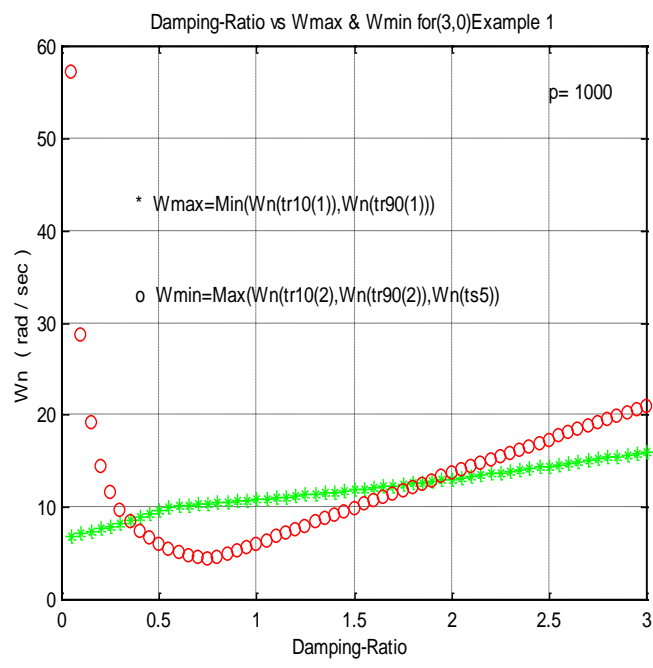
Figure 2-6: Upper bound and lower bound of undamped natural frequency
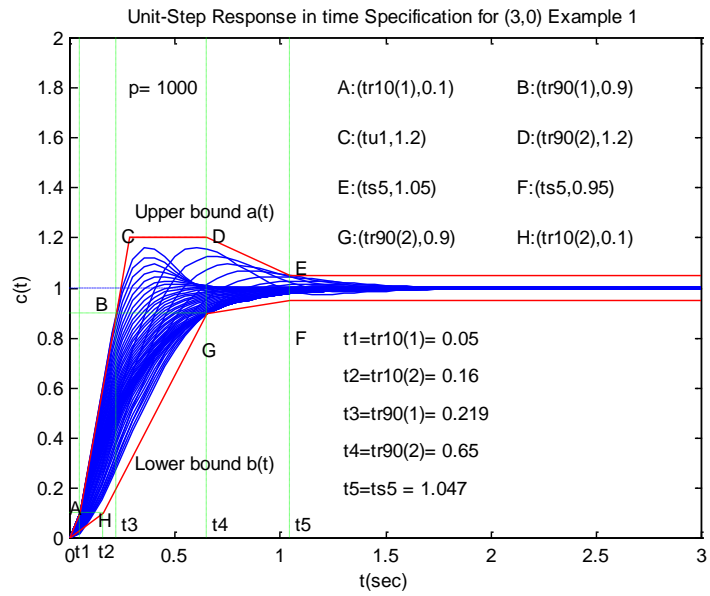
Figure 2-7: Time-domain response of unit step input
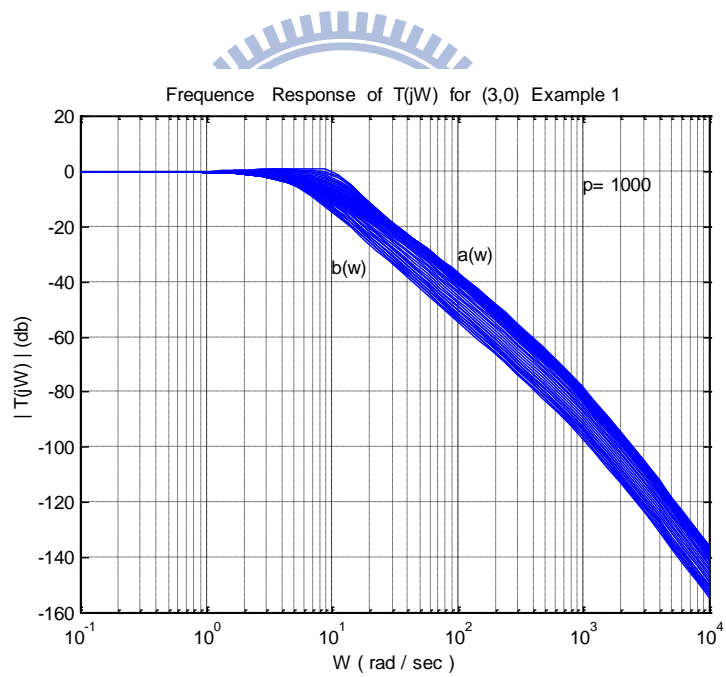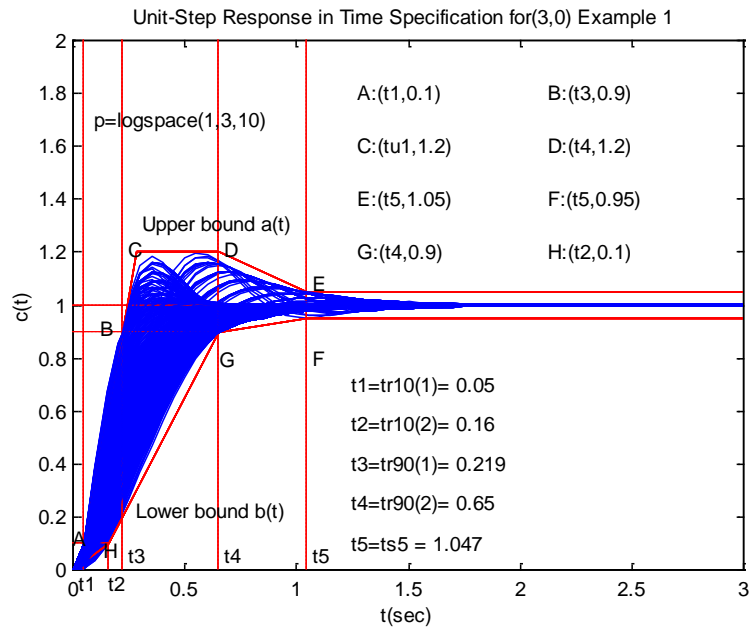

Figure 2-8: Frequency-domain bounds
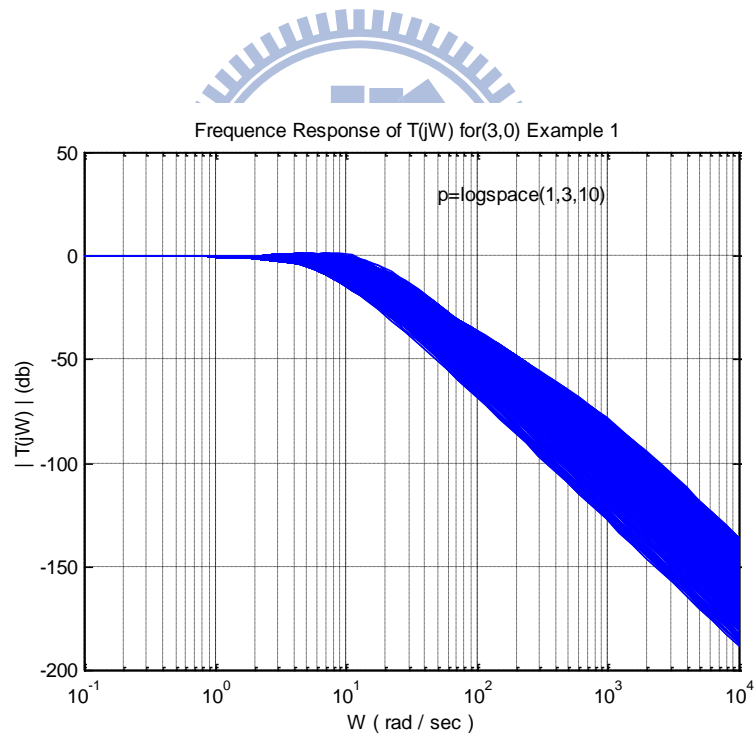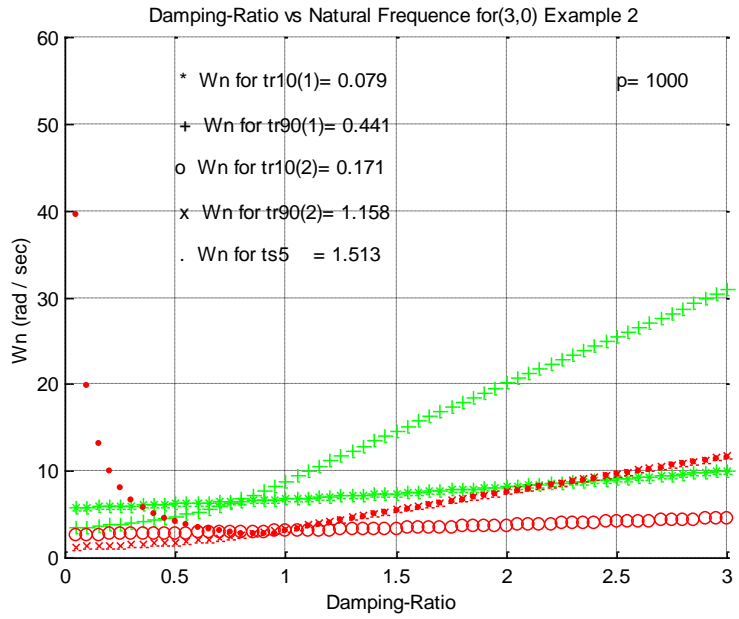
Figure 2-9: Time-domain response of unit step input

Figure 2-10: Frequency-domain bounds
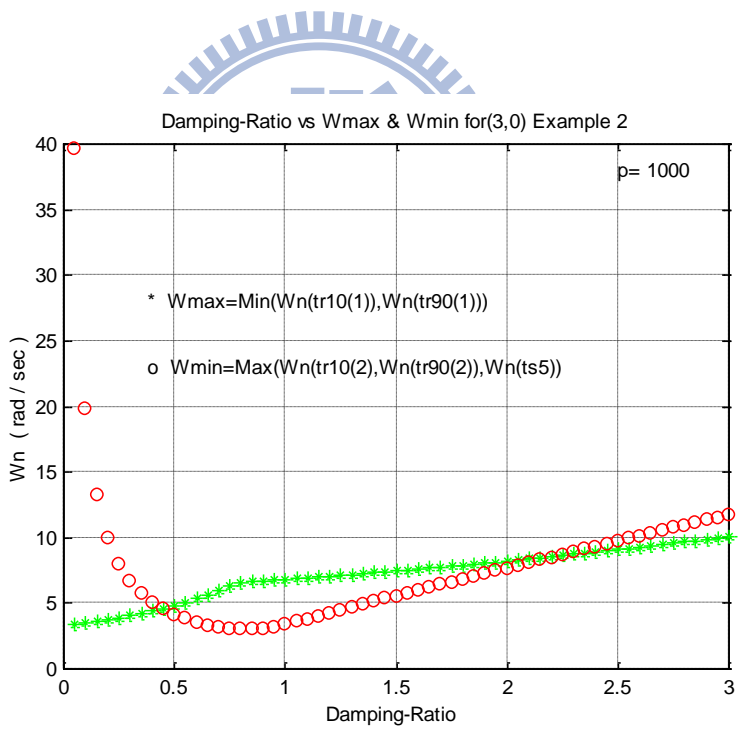
Figure 2-11: Boundary of undamped natural frequency

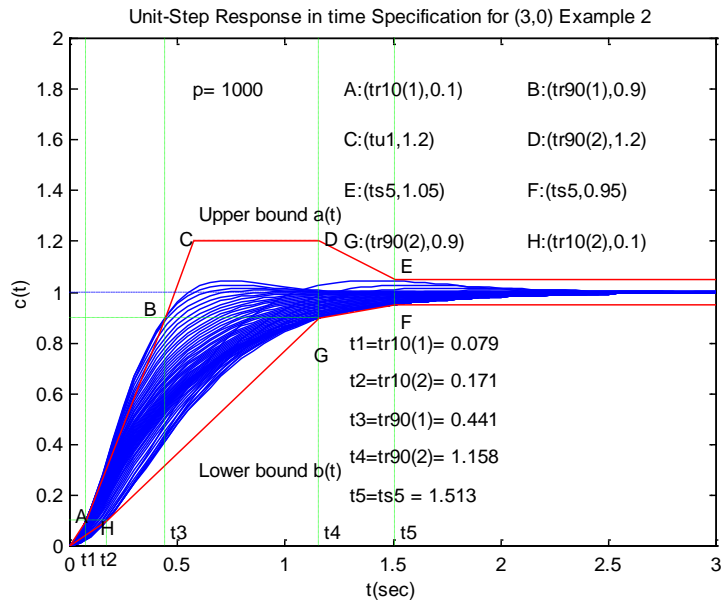**Figure 2-12:** Upper bound and lower bound of undamped natural frequency

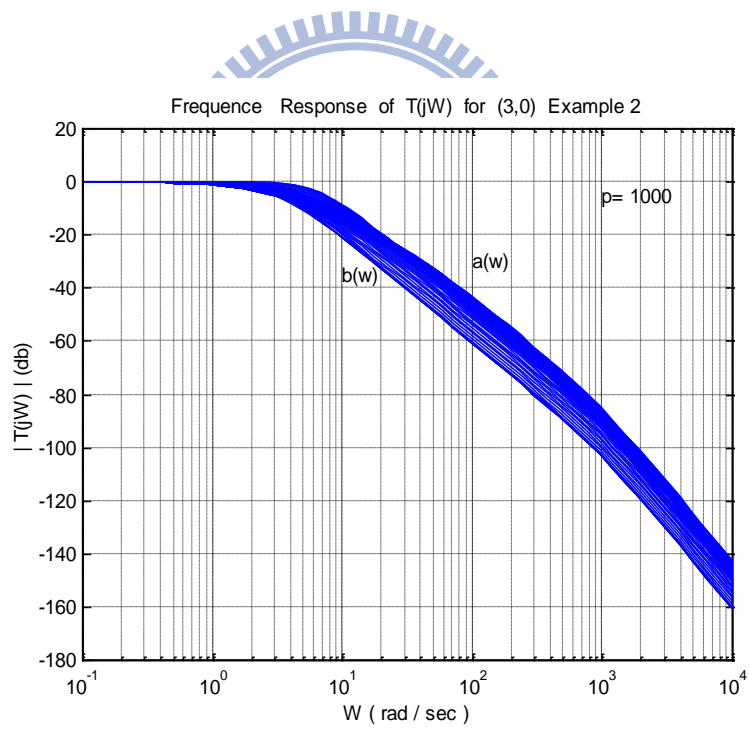**Figure 2-13:** Time-domain response of unit step input
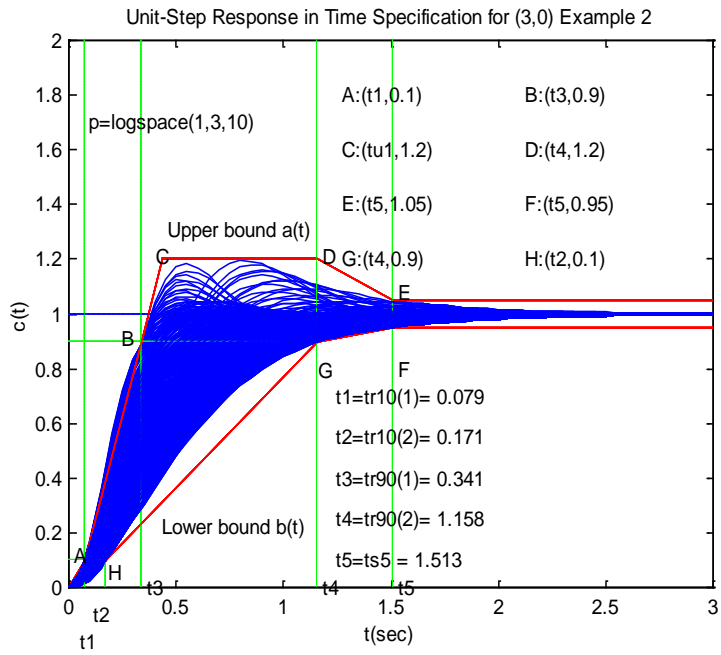

Figure 2-14: Frequency-domain bounds

20

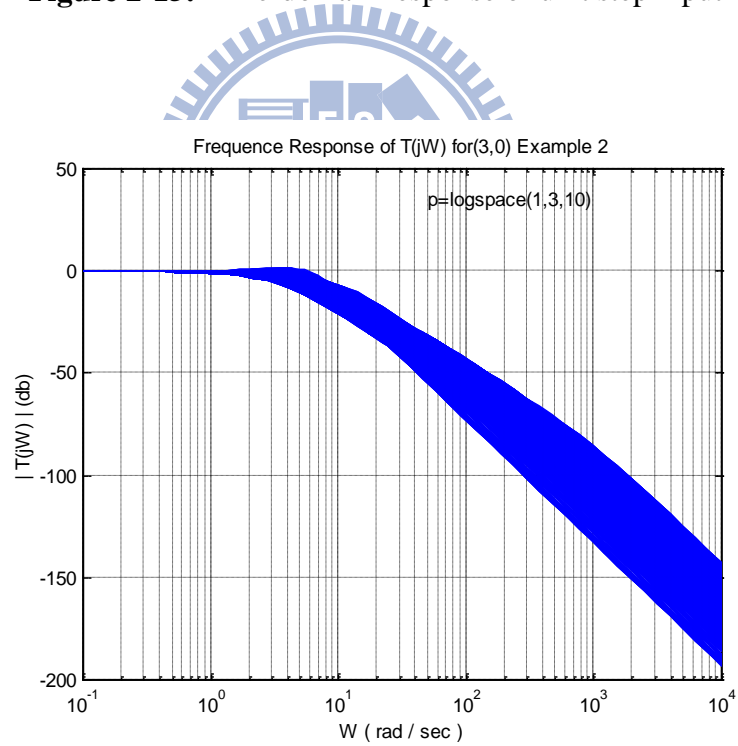**Figure 2-15:** Time-domain response of unit step input

Figure 2-16: Frequency-domain bounds

21

Chapter 3

A QFT/H_∞ DESIGN TECHNIQUE

3.1 Introduction of QFT/H_∞

H_∞ Control theory has been intensively developed for twenty years. Especially, the 2-Riccati-equation method given by Glover and Doyle (1988) opened a new way for state-space theory in H_∞ control. Only in the past ten years have researchers in control theory began to realize that the QFT and H_∞ design philosophies are related. Both QFT and H_∞ design techniques preoccupied the control community for a long time, since the beginning of the 1970s.

The LQG theory was first introduced in the optimal control system by Kalman (1960); Zames (1981) proposed the H_∞ norm concept to give a control system performance criterion, which beginning the H_∞ optimal control research. In 1988 Doyle and Glover [14] has derived controller formula for standard system structure, such that H_∞ optimal control reaches the well stage [16, 17]. The H_∞ optimal control is focus on plant with uncertainty and external disturbance to design a stabilizing control, such that the H_∞ norm from external disturbance to error signal is minimized [16].

Computer program has developed quickly in recent years, although we can design controller by MATLAB QFT Toolbox directly, but in the frequency-domain design procedure, we also need loop shaping in Nichols chart. It is inconvenience to designer. For avoiding the tedious loop-shaping procedure, Sidi [18] proposed a method that combines in an efficient

way the QFT and H_∞ paradigms in the design of two-degree-of-freedom (TDOF) systems [5]. However, how to choose the weighting function has no detail explanation and analysis. Furthermore, many constrains in H_∞ calculation, cannot directly find the solution. This paper proposes the less constrains LMI based method to design H_∞ controller by using MATLAB, which is quickly and simply.

3.2 Statement of QFT Design Problem

A two-degree-of-freedom (TDOF) feedback structure is typically assumed for the QFT technique. In Figure 3-1, wherein the command input (r), plant (P), controller (K), pre-filter (F), system output (y), disturbance (d) and sensor noise (n) are depicted. Where K is the 1st –design-degree-of-freedom to reduce the system sensitivity, and F is the 2nd-design-degree-of-freedom to meet the system performance [10, 19]. Assume the plant with parameter variations; we can denote the transfer function as:

$$P(s) = \frac{b_m s^m + b_{m-1} s^{m-1} + \cdots + b_1 s + b_0}{a_n s^n + a_{n-1} s^{n-1} + \cdots + a_1 s + a_0}, \ (m<n) \tag{3.1}$$

where $a_i \in [a_{i\min}, a_{i\max}]$, $i = 0, 1, \cdots, n$ and $b_j \in [b_{j\min}, b_{j\max}]$, $j = 0, 1, \cdots, m$.

Since plant (P) is with uncertainties, will denote the set of $\wp = \{P(s)\}$ contains all possible transfer function of $P(s)$.
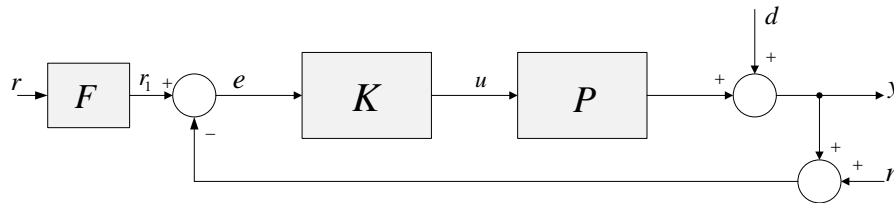

Figure 3-1: Two-degree-of freedom feedback structure

For a plant with parameter variations, if we want to satisfy the time domain specifications: rise time, delay time, settling time and maximum overshoot, then the unit step

23

response of the system must lie within the upper bound $T_u(t)$ and the lower bound $T_l(t)$. In other words, $\forall P(s) \in \wp$, when given a unit step command input $r(t)$, the output $y(t)$ must satisfy：

$$T_l(t) \le y(t) \le T_u(t) \tag{3.2}$$

If we transfer Eq. (3.2) to frequency domain representation, the frequency response of the system must lie within upper bound $T_u(\omega)$ and lower bound $T_l(\omega)$. In other words,

$$T_l(\omega) \le |T(j\omega)|_{dB} \le T_u(\omega) \tag{3.3}$$

where

$$T(s) = \frac{Y(s)}{R(s)} = F(s)T_1(s), \quad T_1(s) = \frac{P(s)K(s)}{1+P(s)K(s)} \equiv \frac{L(s)}{1+L(s)}.$$

The plant uncertainty or output disturbance and sensor noise, the overall system response variation range must satisfy the frequency domain specification:

$$\Delta|T(j\omega)|_{dB} \le T_u(\omega) - T_l(\omega) \tag{3.4}$$

The QFT design steps are as follows: First, establish a template of plant with parameter variations. Second, choose the nominal plant as the reference point of the design. Third, transfer time-domain specifications to frequency-domain specifications. Fourth, set up satisfying frequency specifications boundary in Nichols chart. Finally, choose some frequencies and design loop-gain, which all above the tracking bounds and near U-contour, this step is called loop shaping.
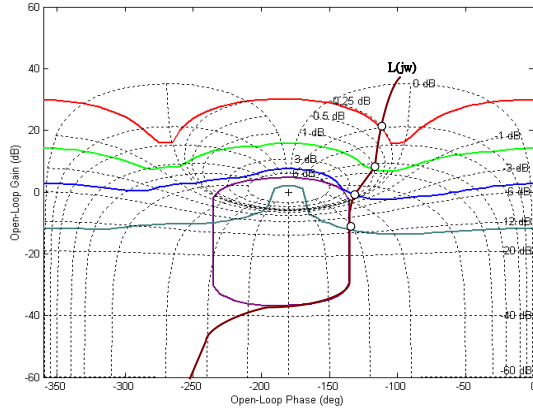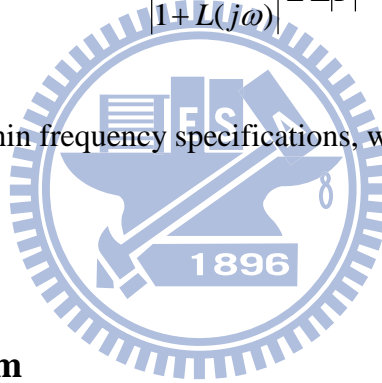
Figure 3-2: Loop shaping

After loop shaping, Eq. (3.4) hold, such that

$$\left| \frac{L(j\omega)}{1 + L(j\omega)} \right| \le \Delta |T|^* \quad . \tag{3.5}$$

For overall $|T(j\omega)|$ within frequency specifications, we should add prefilter to appropriate modification.

3.3 H_∞ **design problem**

A Standard H_∞ control structure is described in Figure 3-3, wherein y is the measurement value of controller, u is the output of controller, ω is external disturbance, z is error signal, and G is transfer function matrix, which contains weighting function[17].
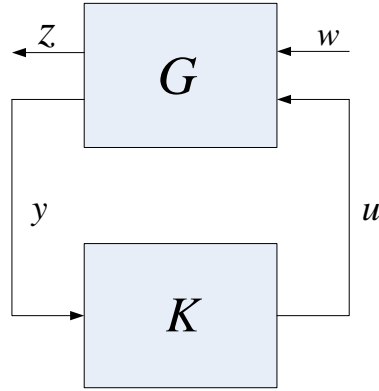
Figure 3-3: A Standard H_∞ control structure

The goal of optimal control is to find a stabilizing controller K, such that the norm of T_{zw} is minimum, where H_∞ norm of T_{zw} is defined as:

$$\|T_{zw}\|_\infty = \sup_\omega \bar{\sigma}(T_{zw}(j\omega)) \tag{3.6}$$

In Figure 3-3 the block diagram T_{zw} is described by a linear fractional transformation (LFT) matrix function, which is convenient for dealing with control problem.

Consider a standard feedback structure in Figure 3-4, we define the loop transfer function as:

$$L = PK \quad . \tag{3.7}$$

Furthermore, we define sensitivity function from d to y as:

$$S = \frac{1}{1+L} . \tag{3.8}$$

Also complementary sensitivity function as:

$$T = 1 - S = \frac{L}{1+L} \quad . \tag{3.9}$$

The transfer function from n to u is

$$T_{un} = \frac{u}{n} = \frac{-K}{1+KP_0} . \tag{3.10}$$

A well-defined feedback control system must satisfy the disturbance rejection in low

frequency range; in high frequency range the effect of sensor noise must be reduced. In the design of feedback controller, we ought to consider the above factors, such that the loop gain lies within a suitable location [17].
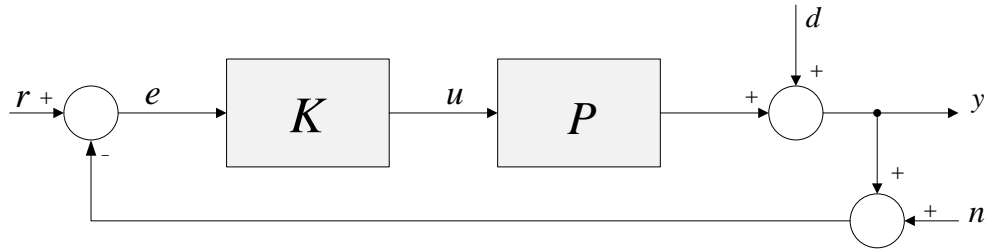

Figure 3-4: A standard feedback structure

For nominal performance and robust stability, we want

$$\|W_S S\|_\infty \leq 1 \tag{3.11}$$

$$\|W_{un} T_{un}\|_\infty \leq 1 \tag{3.12}$$

$$\|W_T T\|_\infty \leq 1 \tag{3.13}$$

where W_S, W_{un}, and W_T stand for the weighting functions of the sensitivity function (S), control sensitivity function (T_{un}), and complementary sensitivity function (T) respectively. This kind of design methodology is called H_∞ mixed sensitivity problem.
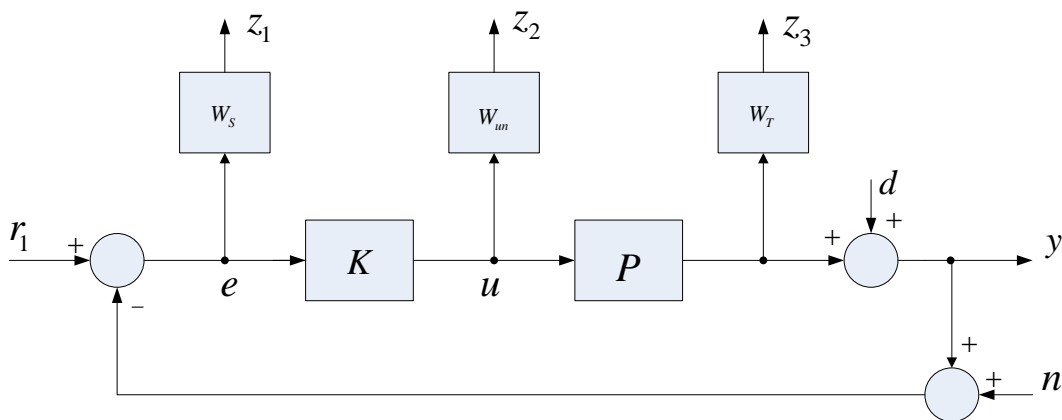

Figure 3-5: Feedback system structure with weighting

Transferring Figure 3-5, feedback structure with weighting to Figure 3-3, LFT structure, and then we can find the controller by H_∞ calculation. There are many design methods in H_∞. Here we use the LMI based H_∞ controller design method, which has less constrain, and is proposed by Gahine and pkarian [20].

Assume the state-space representation of the transfer function matrix (G) be

$$\begin{cases} \dot{x}(t) = Ax(t) + B_1 w(t) + B_2 u(t) \\ z(t) = C_1 x(t) + D_{11} w(t) + D_{12} u(t) \\ y(t) = C_2 x(t) + D_{21} w(t) + D_{22} u(t) \end{cases} \tag{3.14}$$

where $A \in \Re^{n \times n}$, $D_{11} \in \Re^{p_1 \times m_1}$, $D_{22} \in \Re^{p_2 \times m_2}$. We will make the following assumptions:

(A1) (A, B_2) is stabilizable and (A, C_2) detectable. (This is required for the existence of a stabilizing controller K).

(A2) $D_{22} = 0$.

There exists a γ-sub-optimal controller such that $\|T_{zw}\|_\infty = \|F_l(G, K)\|_\infty < \gamma$, iff, there exist two symmetry matrix R and S such that the following inequalities hold：

$$\begin{bmatrix} N_R & 0 \\ 0 & I \end{bmatrix}^T \begin{bmatrix} AR + RA^T & RC_1^T & B_1 \\ C_1 R & -\gamma I & D_{11} \\ B_1^T & D_{11}^T & -\gamma I \end{bmatrix} \begin{bmatrix} N_R & 0 \\ 0 & I \end{bmatrix} < 0 \tag{3.15}$$

$$\begin{bmatrix} N_S & 0 \\ 0 & I \end{bmatrix}^T \begin{bmatrix} A^T S + SA & SB_1 & C_1^T \\ B_1^T S & -\gamma I & D_{11}^T \\ C_1 & D_{11} & -\gamma I \end{bmatrix} \begin{bmatrix} N_S & 0 \\ 0 & I \end{bmatrix} < 0 \tag{3.16}$$

$$\begin{bmatrix} R & I \\ I & S \end{bmatrix} \geq 0 \tag{3.17}$$

where N_R and N_S stand for the null space of (B_2^T, D_{12}^T) and (C_2, D_{21}), respectively.

The solution of the controller is very hard. Here we use the result of γ-sub-optimal controller transfer function

$$K(s) = D_K + C_K(sI - A_K)^{-1}B_K \qquad (3.18)$$

which satisfies the following relationship：

$$T_{zw}(s) = F_l(G, K)(s) = D_{cl} + C_{cl}(sI - A_{cl})^{-1}B_{cl} \qquad (3.19)$$

where

$$A_{cl} = \begin{bmatrix} A + B_2 D_K C_2 & B_2 C_K \\ B_K C_2 & A_K \end{bmatrix}, \quad B_{cl} = \begin{bmatrix} B_1 + B_2 D_K D_{21} \\ B_K D_{21} \end{bmatrix}$$

$$C_{cl} = \begin{bmatrix} C_1 + D_{12} D_K C_2 & D_{12} C_K \end{bmatrix}, \quad D_{cl} = D_{11} + D_{12} D_K D_{21}. \qquad (3.20)$$

Using the hinflmi instruction in MATLAB, we can rapidly find the

γ -sub-optimal H_∞ control solution based on LMI.

3.4   A Combined QFT/H_∞ design method

A combined QFT/H_∞ design method no longer use Nichols chart to find controller. It is

using H_∞ optimal control method to find the controller K . Finally we design a prefilter such

that the output response meets the desired performance.

Reconsider Figure 3-1, the design of two-degree-of-freedom uncertain feedback system,

the design steps are as follows:

3.4.1 Design controller K：

Firstly, without considering the prefilter F , transfer the desired performance

specifications to proper weighting function. Then, use H_∞ method to calculate controller K.

3.4.2 Design prefilter F：

After the controller is designed, we add the prefilter to modify the frequency response

Overall design steps are as shown in Figure 3-6. In the design processes the most important key point is how to choosing weighting function.
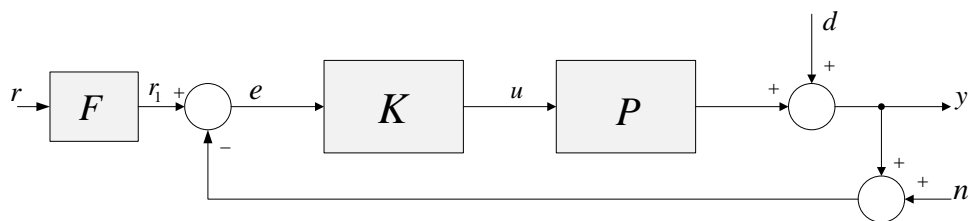
<Step 1>

<Step 2>
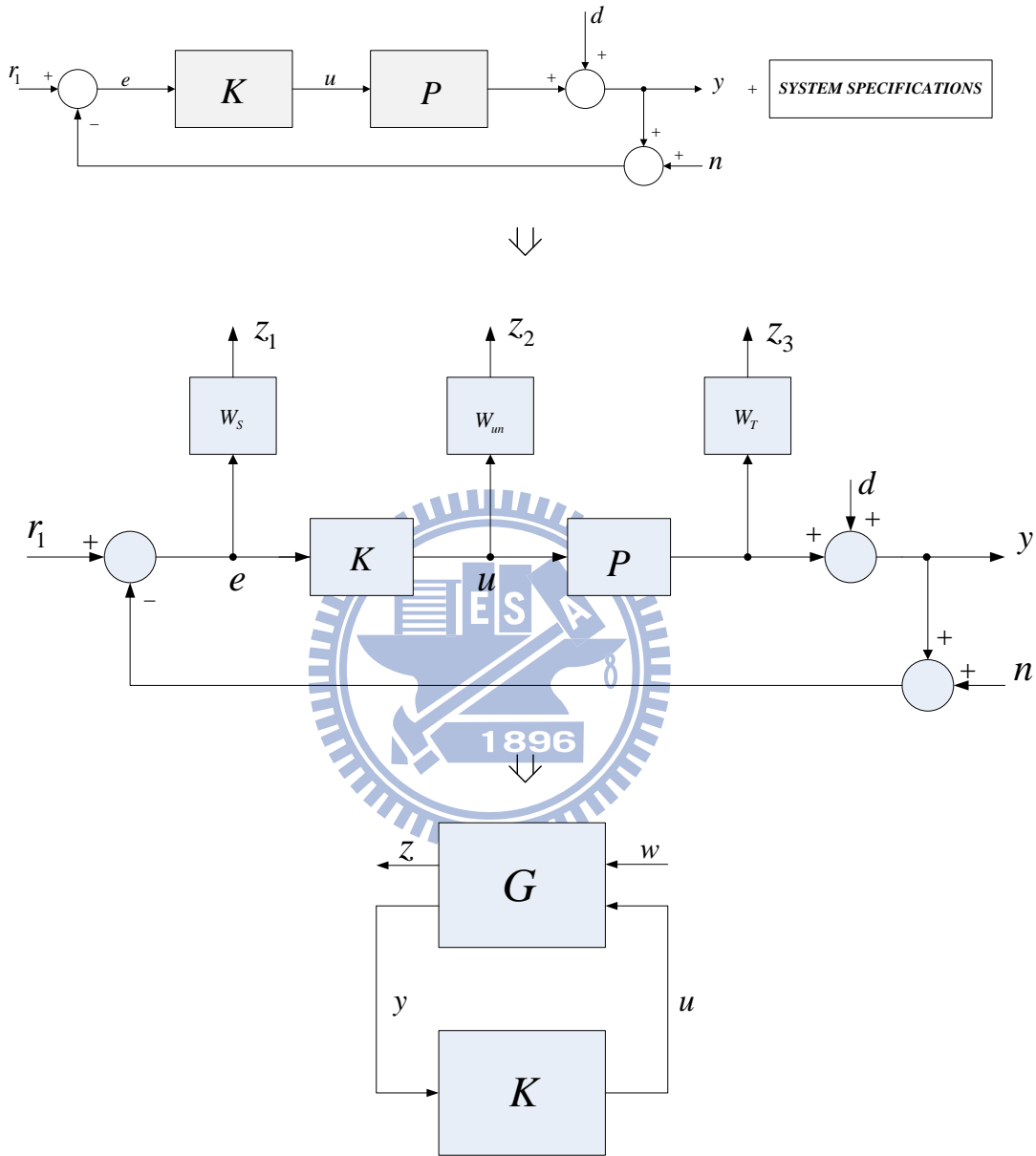
Figure 3-6: A combined QFT / H_∞ design steps

3-5 Multiplicative uncertain model of plant and weighting function W_T.

Assume $P \in \wp$ is a plant with parameter variations, which use the multiplicative uncertainty model, as shown in Figure 3-7. Limit P in the neighbor of the normalized nominal plant P_0, i.e.,

$$P = P_0(I + \Delta P) \tag{3.21}$$

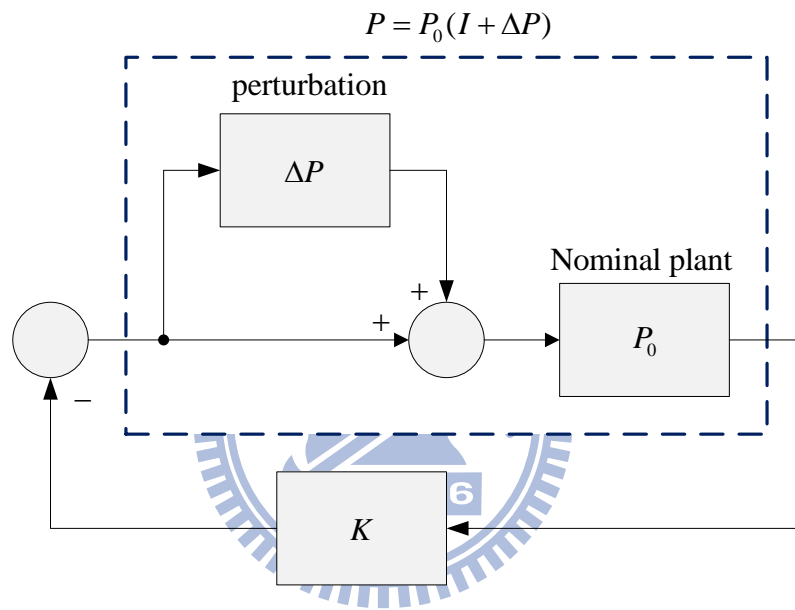where ΔP is the error of multiplicative model.

Figure 3-7: The multiplicative uncertainty model

The term of stability [20, 21] is：

$$\left\| \frac{KP_0}{1 + KP_0} \right\|_\infty < \frac{1}{\|\Delta P\|_\infty} \tag{3.22}$$

or

$$\left\| \Delta P \cdot \frac{KP_0}{1 + KP_0} \right\|_\infty \equiv \|W_T \cdot T_0\|_\infty < 1 \tag{3.23}$$

where $T_0 = \left. \dfrac{y}{r_1} \right|_{P_0} = \dfrac{KP_0}{1 + KP_0}$.

If the system is SISO, then the weighting function can be finding by Eq. (3.24) as

$$\left| \frac{P(j\omega)}{P_0(j\omega)} - 1 \right| \leq |W_T(j\omega)| .$$ (3.24)

3.6 Sensitivity function and weighting function W_S

One of the most important indexes of control system performance specification is sensitivity function, which reflect the degree of overall performance affected by parameter variations in the system. The sensitivity function is defined by

$$\left| S_P^T(j\omega) \right| \equiv \frac{\Delta |T(j\omega)|_{\max}(dB)}{\Delta |P(j\omega)|_{\max}(dB)}$$ (3.25)

where $\Delta |T(j\omega)|_{\max}(dB)$ stands for the largest allowed variation range of $|T(j\omega)|$ in Bode-plot; $\Delta |P(j\omega)|_{\max}(dB)$ stands for the largest allowed variation range of $|P(j\omega)|$ in Bode-plot.

In QFT design concept, we generally choose some frequencies less than 10 rad/sec. Hence when we find sensitivity function we also choose the above frequencies. After choosing nominal plant, using Eq. (3.25), we can find the magnitude response of the nominal sensitivity function $|S_0(j\omega)|$ in Bode-plot. Next by using curve-fitting we can get the nominal sensitivity function $S_0(s)$.

If the desired sensitivity function satisfies $|S(j\omega)| \leq D^*(\omega)$, then the system desired performance could be written as:

$$|W_S(j\omega)S_0(j\omega)| \leq 1$$ (3.26)

or

$$|W_S(j\omega)| \leq \frac{D^*(\omega)}{|S_0(j\omega)|} .$$ (3.27)

3.7 High frequency noise rejection and weighting function W_{un}

After giving nominal plan P_0 , the transfer function from *n* to *u* is

$$T_{un} = \frac{u}{n} = \frac{-K}{1 + KP_0} \tag{3.28}$$

Eq. (3.28) stands for the amplifier effect of sensor noise. If we can reduce the amplifier effect

of the sensor noise, then the cost of feedback design is also reduced.   Adding the weighting

function W_{un} into the transfer function T_{un}, we have

$$\left| W_{un}(j\omega)T_{un}(j\omega) \right| \leq 1. \tag{3.29}$$

A good control system should have proper loop-gain, which reduces the sensor noise

disturbance at high frequency, and reduces sensitivity at low frequency. This property is

considered in choosing W_{un} and W_s .

After transfer the system desired performance specifications to proper weighting

function, we can use H_∞ method to find the controller K , and then use the same method in

QFT, we add prefilter to make suitable modification. Finally check the overall step response

in Eq. (3.2) to meet the desired. The flow chart of overall combined QFT / H_∞ design method
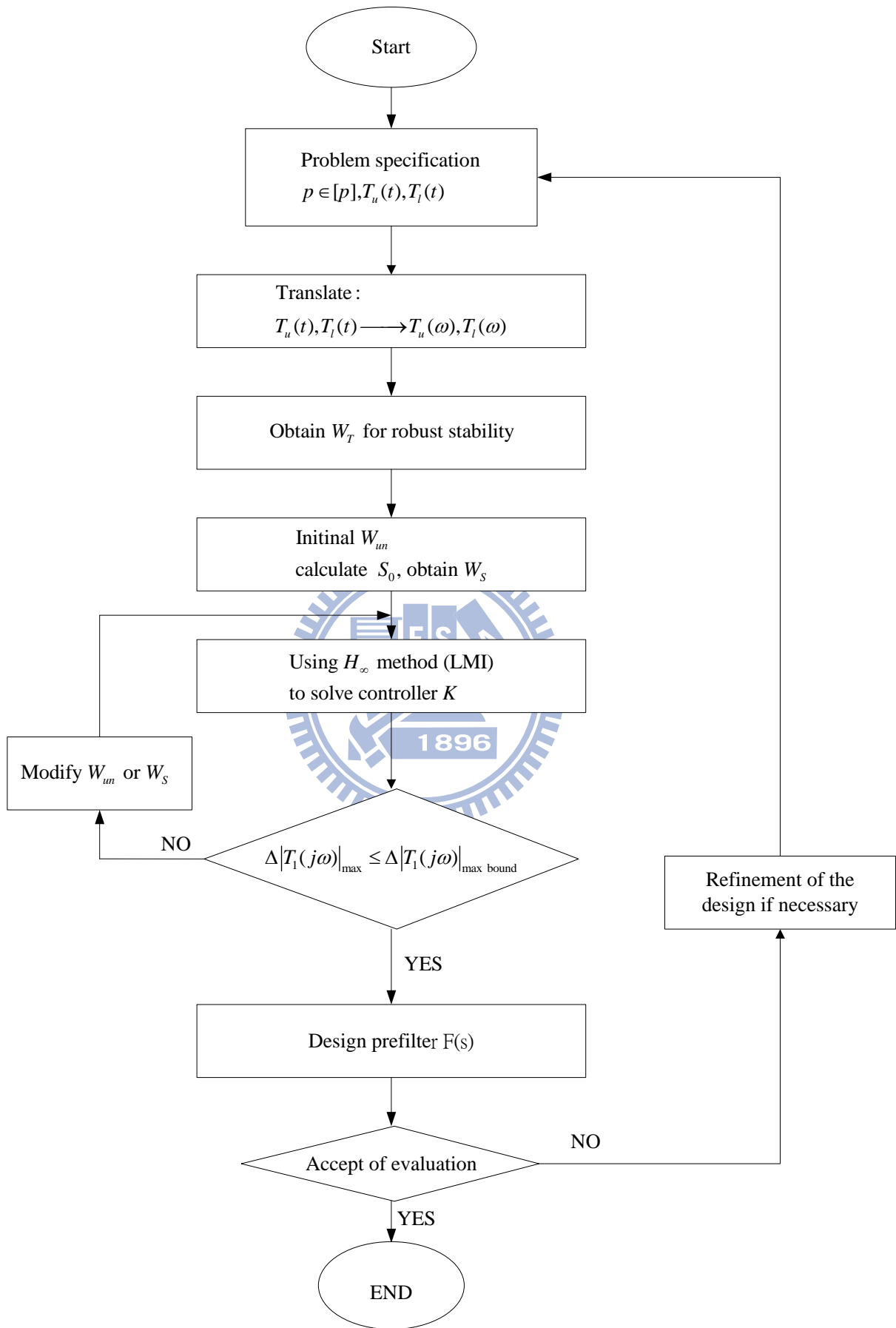
is shown in Figure 3-8.

Figure 3-8: A flow chart of combined QFT/H_∞ design

Start

Problem specification
$p \in [p], T_u(t), T_l(t)$

Translate :
$T_u(t), T_l(t) \longrightarrow T_u(\omega), T_l(\omega)$

Obtain W_T for robust stability

Initinal W_{un}
calculate S_0, obtain W_S

Using H_∞ method (LMI)
to solve controller K

Modify W_{un} or W_S

NO

$\Delta |T_1(j\omega)|_{max} \leq \Delta |T_1(j\omega)|_{max\ bound}$

Refinement of the
design if necessary

YES

Design prefilter F(s)

NO

Accept of evaluation

YES

END

34

3.8 Design Example of SISO System

3.8.1 Problem description

Assume plant with large parameter variations as following,

$$\wp = \left\{ P(s) = \frac{k}{(s+a)(s+b)}, \ k \in [1, 20], \ a \in [1, 5], \ b \in [1.5, 2.5] \right\}$$

design controller K and prefilter F such that the system satisfies: (1) settling time: $t_s \leq 3$ sec, (2) maximum overshoot: $M_p \leq 10\%$, (3) disturbance attenuation:

$$\left| T_d(j\omega) \right|_{dB} \leq 3 \ dB [21].$$

3.8.2 Specification transformation

(1) Transfer time-domain specifications to the transfer function of upper and lower bound,

$$T_u(s) = \frac{0.338s + 6.76}{s^2 + 3.12s + 6.76}, \quad T_l(s) = \frac{29.4}{(s+15)(s^2 + 3.08s + 1.96)}$$

Drawing the boundary of frequency for the system in Figure 3-9.

(2) Disturbance attenuation $\left| T_d(j\omega) \right|_{dB} \leq 3 \ dB$, i.e. sensitivity function should satisfy.

3.8.3 Choosing the weighing function W_{T}

(1) Let the nominal plant be $P_0(s) = \frac{1}{(s+1)(s+1.5)}$.

(2) Using Equation (3.24) to find weighting function W_T. Computer $\left| \frac{P(j\omega)}{P_0(j\omega)} - 1 \right|$. Since

$\max \left| \frac{P(j\omega)}{P_0(j\omega)} - 1 \right| = 19$, we can choose $W_T = 20$, as shown in Figure 3-9.

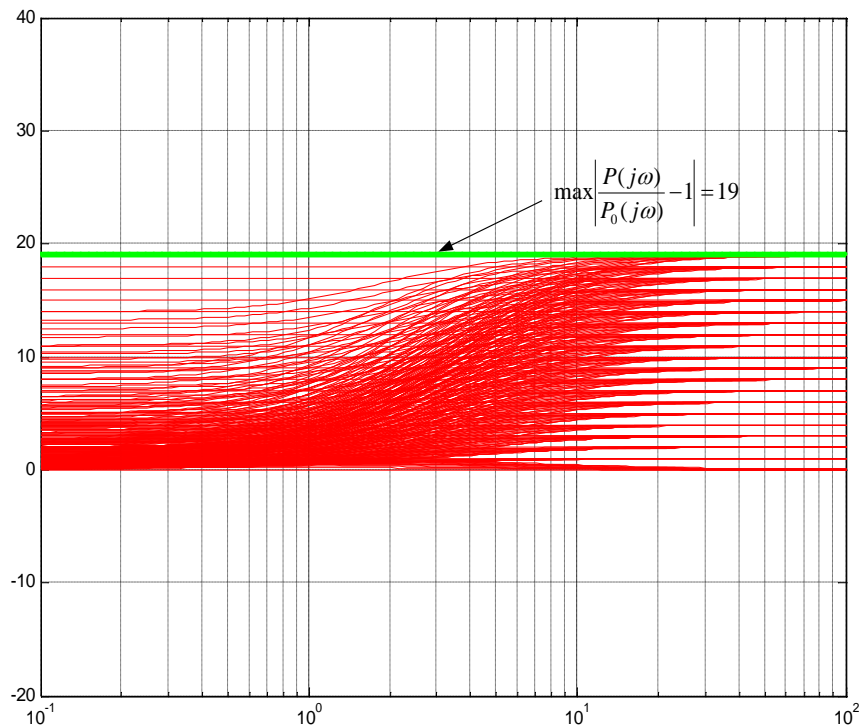$$\max\left|\frac{P(j\omega)}{P_0(j\omega)}-1\right|=19$$

Figure 3-9: Selection of complementary weighting function

3.8.4   Choosing weighting function W_{un} and W_s

(1) First let initial value of $W_{un}=1$, and then if necessary we can adjust it.

(2) Using Eq. (3.25) and Eq. (3.26) to find the weighting function W_s.

We choose only some special frequencies, $\omega = 0.1,\ 0.2,\ 0.4,\ 0.8,\ 1,\ 2,\ 4,\ 8,\ 10$ to design.

Sensitivity function can be found by following table directly.

Table 1: Table of sensitivity function

$\omega(rad/s)$	$\Delta|T(j\omega)|_{max}(dB)$	$\Delta|P(j\omega)|_{max}(dB)$	$|S(j\omega)|$
0.1	0.0665	44.3832	0.0015
0.2	0.2619	44.2248	0.0059
0.4	0.9893	43.6315	0.0227
0.8	3.2984	41.7346	0.079
1	4.6318	40.6446	0.114
2	10.4374	35.8033	0.2915
4	12.8271	30.7046	0.4178
8	13.0651	27.64	0.4727
10	13.7038	27.1131	0.5054

When we have the data of ω and $|S(j\omega)|$, and then we can use mrfit instruction of MATLAB[22, 23] to find approximately the sensitivity function as：

$$S(s) = \frac{0.50524\,(s+0.001)^2}{s^2 + 3.5616\,s + 3.336}$$

For guaranteeing $|S(j\omega)| \le 1.414 \equiv D^*(\omega)$, and $|W_S(j\omega)S(j\omega)| \le 1$, we choose

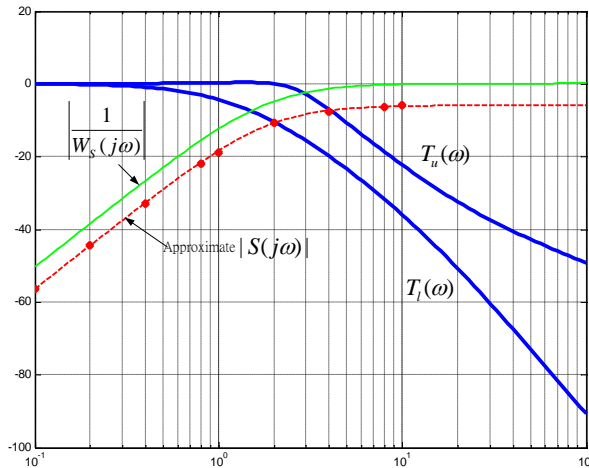$$W_S(s) = \frac{0.5}{S(s)} = \frac{0.9896\,(s^2 + 3.5616\,s + 3.336)}{(s+0.001)^2}$$

Figure 3-10: Boundary of frequency and sensitivity function

3.8.5 Using H_∞ algorithm to find controller $K(s)$

Using the hinflmi instruction of MATLAB to calculate, we can get the controller as:

$$K(s) = \frac{108607.3671(s+1.5)(s+1)(s+0.1574)}{(s+5147)(s+6.759)(s+0.001003)(s+0.0009754)}$$

$$\approx \frac{21.1001(s+1.5)(s+1)(s+0.1574)}{(s+6.759)(s+0.001003)(s+0.0009754)}$$

3.8.6 Checking the frequency response

When we got the controller, and then plot the frequency response $|T_1(j\omega)|$ without prefilter. From Figure 3-11, we can find that the controller does not meet the desired performance in the low frequency, hence we must adjust the weighting function. The method of adjustment may reduce the value of W_s or W_{un}, and then find the controller again.
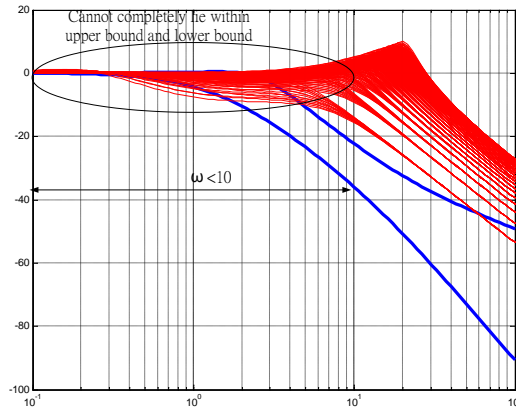
38

Figure 3-11: Frequency response without prefilter ($W_{un} = 1$)

If we let weighting function be $W_{un} = 0.1$, the controller is

$$K(s) = \frac{1367680.4466(s+1.5)(s+1)(s+0.1745)}{(s+6782)(s+20.34)(s+0.0009997)(s+0.0009639)}$$

$$\approx \frac{201.663(s+1.5)(s+1)(s+0.1745)}{(s+20.34)(s+0.0009997)(s+0.0009639)}$$

Similarly, after checking $\Delta|T_1(j\omega)|_{max}$ we see that it does not completely lie within upper boundary and low boundary, as showing in Figure 3-12. Hence we should adjust weighting function again. We choose $W_{un} = 0.05$ to find the controller again.
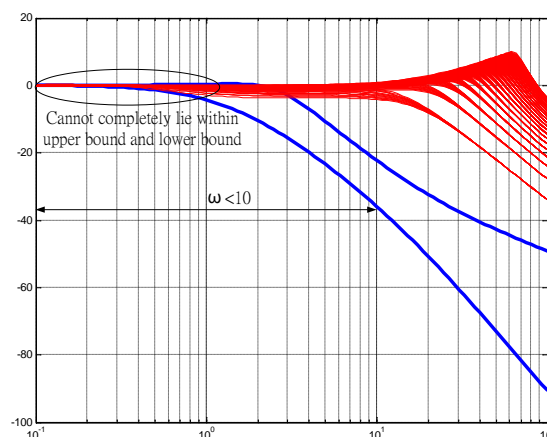
Figure 3-12: Frequency response without prefilter ($W_{un} = 0.1$)

When we choose weighting function $W_{un} = 0.05$, the controller is

$$K(s) = \frac{1112902.8204(s+1.5)(s+1)(s+0.09977)}{(s+2802)(s+28.6)(s+0.001006)(s+0.0009742)}$$

$$\approx \frac{397.1816(s+1.5)(s+1)(s+0.09977)}{(s+28.6)(s+0.001006)(s+0.0009742)}$$

Checking again, we find $\Delta|T_1(j\omega)|_{\max}$ that the response curves are all completely within upper boundary and low boundary, as showing in Figure 3-13, and then the controller K is acceptable.
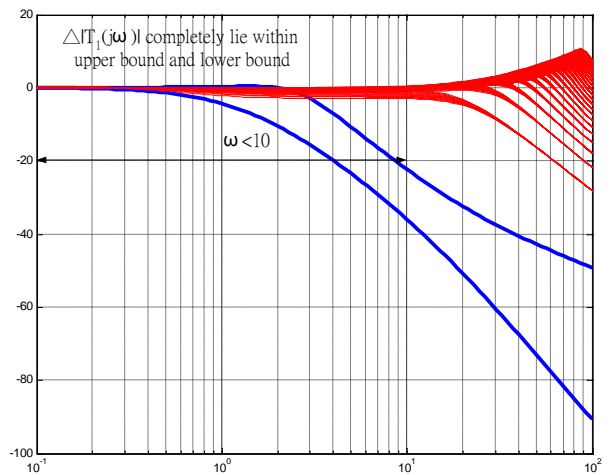

Figure 3-13: Frequency response without prefilter ($W_{un} = 0.05$)

From Figure 3-11 to Figure 3-13, we can adjust the rate of rolling off in high frequency by weighting function W_{un}.

3.8.7 Design prefilter $F(s)$

Although $\Delta|T_1(j\omega)|_{\max}$, completely lie within upper boundary and lower boundary, still must add prefilter such that overall transfer function $|T(j\omega)| = |F(j\omega)T_1(j\omega)|$ meets the desired frequency specification. After proper adjust, then we get the prefilter as:

$$F(s) = \frac{1}{(1+\frac{s}{2})(1+\frac{s}{3})} = \frac{6}{(s+2)(s+3)}$$
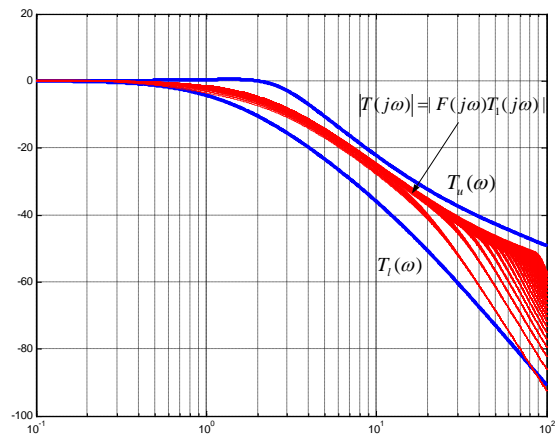

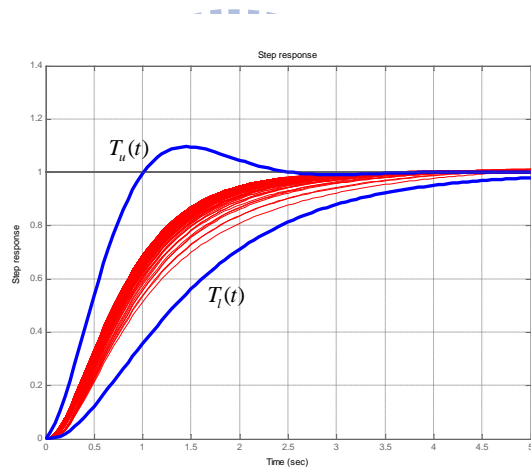**Figure 3-14:** Frequency response with prefilter

Figure 3-15: Step response with prefilter

Finally, we check step response of Figure 3-15, if all step responses meet the desired time-domain specifications, and then overall design procedure is completed.

3.8.8 Comparison QFT/H_∞ and QFT design

Traditional QFT design controller is

$$K(s) \approx \frac{271.6614\,(s+2.788)(s+2.442)(s+0.505)}{(s+96.78)(s+0.3369)(s+0.003733)}$$

prefilter is $F(s) = \dfrac{0.28(s+25)}{(s+2)(s+3.5)}$
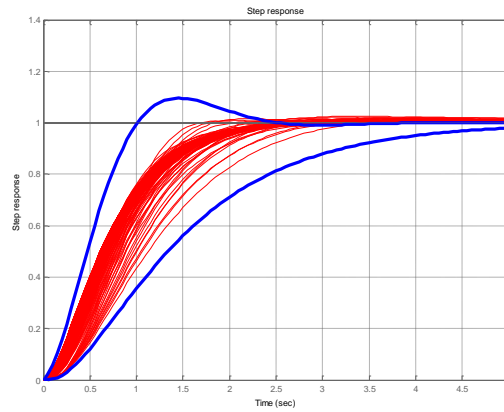

Figure 3-16: Step response of traditional QFT design

From Figure 3-15 and Figure 3-16, we find a combined of QFT/H_∞ design can achieve
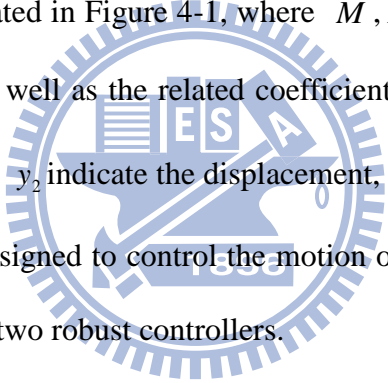
the traditional QFT design.

Chapter 4

CONTROLLER DESIGN OF AN MIMO

SUSPENSION SYSTEM

4.1   Structure Description

A basic quarter vehicle suspension system presented in [23] as a target model for an MIMO design case is illustrated in Figure 4-1, where M, K, and B denote the mass, spring, and damper, respectively, as well as the related coefficients, and where f_1 and f_2 indicate the driving force and y_1 and y_2 indicate the displacement, respectively. The two-mass motion control system set up and designed to control the motion of the suspension, using the control functionality realized by the two robust controllers.
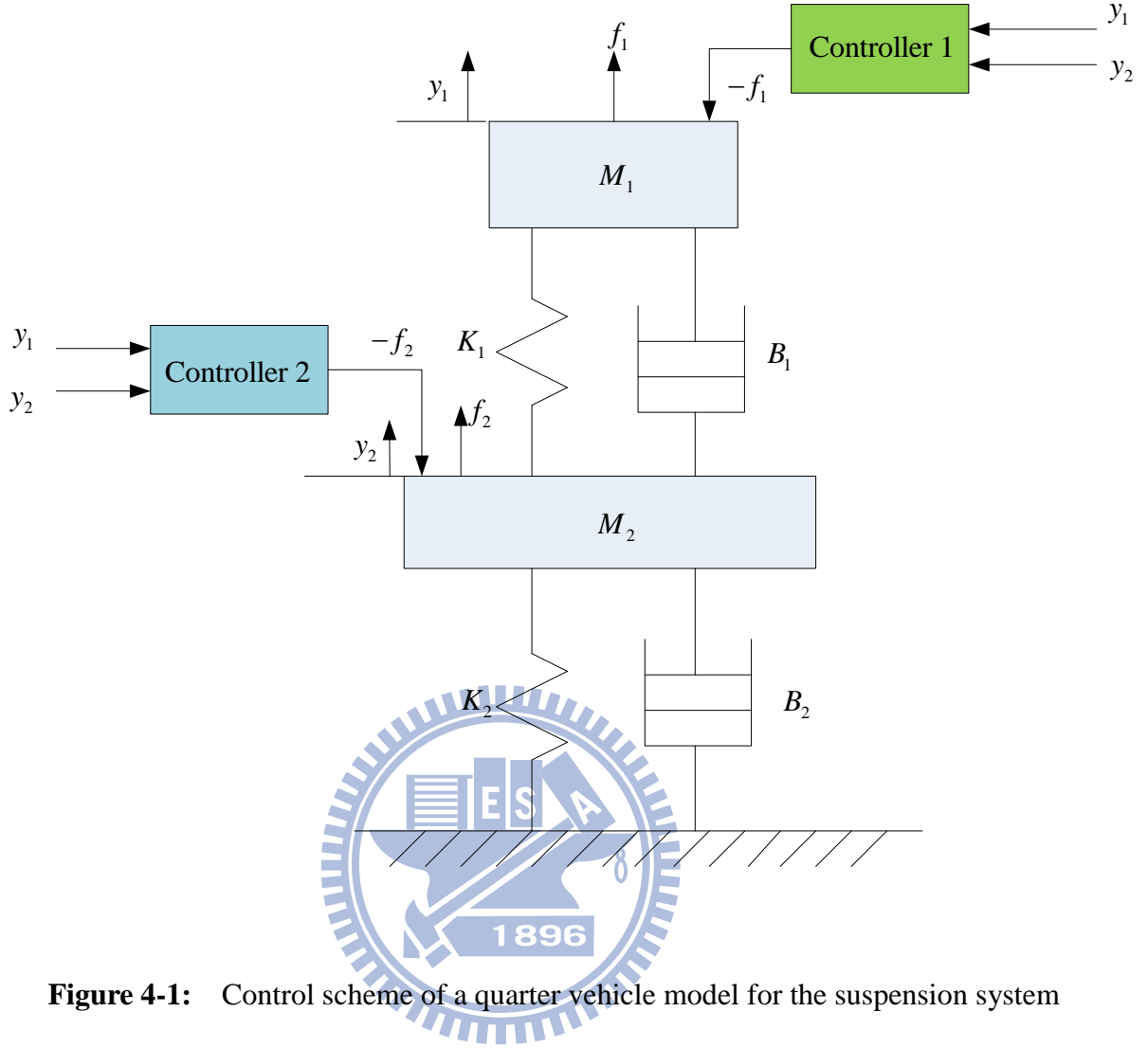
Figure 4-1: Control scheme of a quarter vehicle model for the suspension system

Moreover, the motion equations of the suspension system in Figure 4-1are described as

$$f_1 = M_1\ddot{y}_1 + B_1(\dot{y}_1 - \dot{y}_2) + K_1(y_1 - y_2) \tag{4.1}$$

$$f_2 = M_2\ddot{y}_2 + B_1(\dot{y}_2 - \dot{y}_1) + K_1(y_2 - y_1) + K_2 y_2 + B_2\dot{y}_2 \tag{4.2}$$

Let $y_1 = x_1$, and $y_2 = x_2$, then the first-order differential equations of displacements of the spring mass and unsprung mass can be described as $\dot{y}_1 = \dot{x}_1$, $\dot{y}_2 = \dot{x}_2$. Subsequently, assume $\dot{x}_1 = x_3$ and $\dot{x}_2 = x_4$ then the first-order differentiation expressions of x_3 and x_4 are represented as $\ddot{y}_1 = \dot{x}_3$ and $\ddot{y}_2 = \dot{x}_4$, respectively. Eq. (4.1) can be rewritten as

$$f_1 = M_1 \dot{x}_3 + B_1(x_3 - x_4) + K_1(x_1 - x_2)$$

$$\dot{x}_3 = \frac{-K_1}{M_1} x_1 + \frac{K_1}{M_1} x_2 - \frac{B_1}{M_1} x_3 + \frac{B_1}{M_1} x_4 + \frac{1}{M_1} f_1$$

(4.3)

and also Eq. (4.2) can be rewritten as

$$f_2 = M_2 \dot{x}_4 + B_1(x_4 - x_3) + K_1(x_2 - x_1) + K_2 x_2 + B_2 x_4$$

$$\dot{x}_4 = \frac{K_1}{M_2} x_1 - \frac{(K_1 + K_2)}{M_2} x_2 + \frac{B_1}{M_2} x_3 - \frac{(B_1 + B_2)}{M_2} x_4 + \frac{1}{M_2} f_2.$$

(4.4)

The motion equations of the suspension system model are converted into matrix form as

$$
\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} =
\begin{bmatrix}
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1 \\
-\dfrac{K_1}{M_1} & \dfrac{K_1}{M_1} & -\dfrac{B_1}{M_1} & \dfrac{B_1}{M_1} \\
\dfrac{K_1}{M_2} & -\dfrac{K_1 + K_2}{M_2} & \dfrac{B_1}{M_2} & -\dfrac{B_1 + B_2}{M_2}
\end{bmatrix}
\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} +
\begin{bmatrix}
0 & 0 \\
0 & 0 \\
\dfrac{1}{M_1} & 0 \\
0 & \dfrac{1}{M_2}
\end{bmatrix}
\begin{bmatrix} f_1 \\ f_2 \end{bmatrix}
$$

(4.5)

and the output equations are shown as

$$
\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} =
\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}
\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}.
$$

(4.6)

Furthermore, in frequency domain, the displacement responses of sprung mass and unsprung mass are calculated by using Eq. (4.5) and Eq. (4.6) taking Lapalce transform to obtain

$$
\begin{bmatrix} Y_1(s) \\ Y_2(s) \end{bmatrix} =
\begin{bmatrix}
\dfrac{M_2 s^2 + (B_1 + B_2)s + K_1 + K_2}{\Delta} & \dfrac{B_1 s + K_1}{\Delta} \\
\dfrac{B_1 s + K_1}{\Delta} & \dfrac{M_1 s^2 + B_1 s + K_1}{\Delta}
\end{bmatrix}
\begin{bmatrix} F_1(s) \\ F_2(s) \end{bmatrix},
$$

(4.7)

where $\Delta = (M_1 s^2 + B_1 s + K_1)[M_2 s^2 + (B_1 + B_2)s + K_2] - (B_1 s + K_1)^2$. Moreover, in this case, the numerical values of suspension system depicted in Eq. (4.7) are represented as

$$P_{11} = \frac{0.001028s^2 + 0.01s + 1.297}{s^4 + 10.86s^3 + 1306s^2 + 1004s + 3.894 \times 10^4}$$

$$P_{12} = \frac{0.009872s + 0.3851}{s^4 + 10.86s^3 + 1306s^2 + 1004s + 3.894 \times 10^4}$$

$$P_{21} = \frac{0.009872s + 0.3851}{s^4 + 10.86s^3 + 1306s^2 + 1004s + 3.894 \times 10^4}$$

$$P_{22} = \frac{0.008772s^2 + 0.009872s + 0.3851}{s^4 + 10.86s^3 + 1306s^2 + 1004s + 3.894 \times 10^4}$$

(4.8)

From the control system block diagram, a standard feedback configuration with weights is shown in Figure 3-6. Figure 3-6 can be redrawn in the general linear fractional transformation (LFT) framework as shown in Figure 4-2. Consequently, the H-infinity controller of quarter vehicle suspension system model can be calculated and obtained.
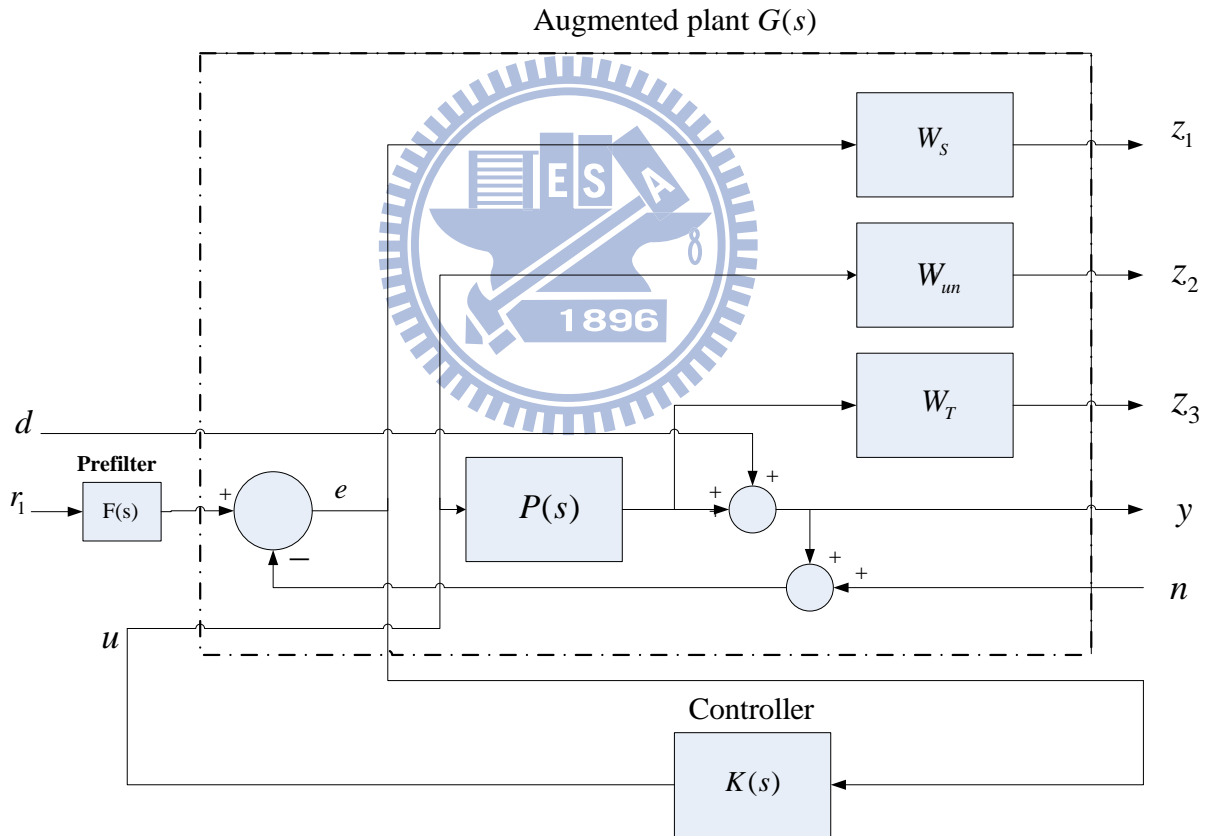

Figure 4-2: A LFT framework for control system structure

4.2 Weighting function selections

The selection of three weighting functions shown in Figure 4-2, is described as follows.

Firstly, the weighting function W_T is obtained from Eq. (3.24), where P_0 denotes the nominal plant and P indicates the uncertainty elements of plant. Next, the nominal model P_0 is considered the diagonal elements and the uncertainty model P is considered the off-diagonal elements [25] in Eq. (4.8) so that the system variations $\left| P(j\omega)/P_0(j\omega)-1 \right|$ will be less or equal than $\left| W_T \right|$ in all frequency. Secondly, in order to obtain the weighting function W_S, Eq. (3.24) is computed and then use curve fitting algorithm to obtain an initial and suitable two-order transfer functions so that $\left| S(j\omega)W_S(j\omega) \right|=1$. Finally, the first design stage for selection of weighting function W_{un} is chosen a low gain $\left| W_{un}(j\omega) \right|$. With this initial condition, the bode plot of the overall $\left| \Delta T_1(j\omega) \right|$ for $\forall P$ and checks if the overall $\Delta \left| T_1(j\omega) \right|$ changes are smaller at $\forall \omega$. In other words, the adjustment value of weighting function W_{un} as a tuning parameter depends upon $\Delta \left| T_1(j\omega) \right|$ variations due to the flowchart design procedure shown in Figure 3-8.

4.3 Controller Design

The controller design of suspension system is described in this section. The design steps of controller are shown in Figure 3-8, where the weighting functions W_{un} and W_S are tuning parameters. The developed H_∞ controllers are rewritten in Table 3. Obviously, the mainly advantage of this paper is that the new flowchart is redrawn and illustrated in Figure 3-8 without noise filter because of the developed H_∞ controller. Namely, the noise filter, proposed by Sidi [18], is substituted for the developed H_∞ controller which can be applied in the variation of each parameter of suspension system.

Chapter 5

SIMULATION RESULTS

The response of the controllers obtained from this QFT / H_∞ approach can be described. However, *QFT* algorithm does not deal with the controller design of MIMO system very well. The *QFT* constraints on the criteria of the MIMO system with uncertainty components can be described. Based on setting the design conditions appropriately, the applicable controllers can be calculated by the QFT / H_∞ approach. Figure 5-1 shows the singular value of uncertainty plant, where the solid line represents the weighting function W_{T1}, $\frac{0.4(0.4s+300)}{(0.55s+30)}$ and the dashed line denotes the uncertainty parameter variations. Then, the weighting function W_T for this case is chosen $W_T = diag[W_{T1} * inv(W_{T1}), W_{T1} * inv(W_{T1})]$ so that the fluctuation of overall system can be reduced to zero. Figure 5-2 shows the magnitude of sensitivity function, W_S. The selecting of control effort weighting function W_{un} is as a tuning parameter depending upon the sensitivity function variation. In this case, the weighting function is constant gain for suspension system, i.e. $W_{un} = 1.2diag(120,120)$. The table 2 represents the specific design items for suspension system and the table 3 provides the design controller of the suspension system with the sub-optimal $\gamma = 24.4162$. Finally, Figure 5-3 illustrates the acceleration response y_1, which denotes input signal, road profile, to mass 1, M_1, and Figure 5-4 illustrates the acceleration response of y_2, which denotes input signal, road profile, to mass 2, M_2, that is, the variations of mass 1 and mass 2 can be reduced to zero when the suspension

system travels along the unit step target path. Figure 5-5 is Bode plot of $P_{11} * P_{22}$ and $P_{12} * P_{21}$, Figures 5-6 to 5-9 are the Bode plots of plant P_{11}, P_{12}, P_{21} and P_{22} , respectively. Figures 5-10 to 5-15 are the Bode plots of plant by B_1, B_2, K_1, K_2, M_1, and M_2 variations, respectively. Figure 5-16 shows y_1 response without prefilter in time domain and Figure 5-17 shows y_1 response with prefilter in time domain. Figure 5-18 shows y_1 response without prefilter in frequency domain, and Figure 5-19 shows y_1 response with prefilter in frequency domain. Figure 5-20 shows y_2 response without prefilter in time domain and Figure 5-21 shows y_2 response with prefilter in time domain. Figure 5-22 shows y_2 response without prefilter in frequency domain, and Figure 5-23 shows y_2 response with prefilter in frequency domain.

Table 2. System parameters

Parameters	Values	Units
M_1	973	(kg)
M_2	114	(Kg)
B_1	1095	(Ns/m)
B_2	14.6	(Ns/m)
K_1	42720	(N/m)
K_2	101115	(N/m)

Table 3. The results of *QFT/H$_\infty$* controller

Controllers	Input (u_1)
y_1	$K_{11} = \dfrac{25230.2692(s+1.076)(s+0.5117)(s^2+0.5534s+30.68)(s^2+10.32s+1269)}{(s+3.055)(s+0.7148)(s^2+0.5414s+30.68)(s^2+10.32s+1269)}$
y_2	$K_{12} = \dfrac{91.4522(s+1.076)(s+0.5117)(s-0.1167)(s^2+9.955s+1285)}{(s+3.055)(s+0.7148)(s^2+0.5414s+30.68)(s^2+10.32s+1269)}$

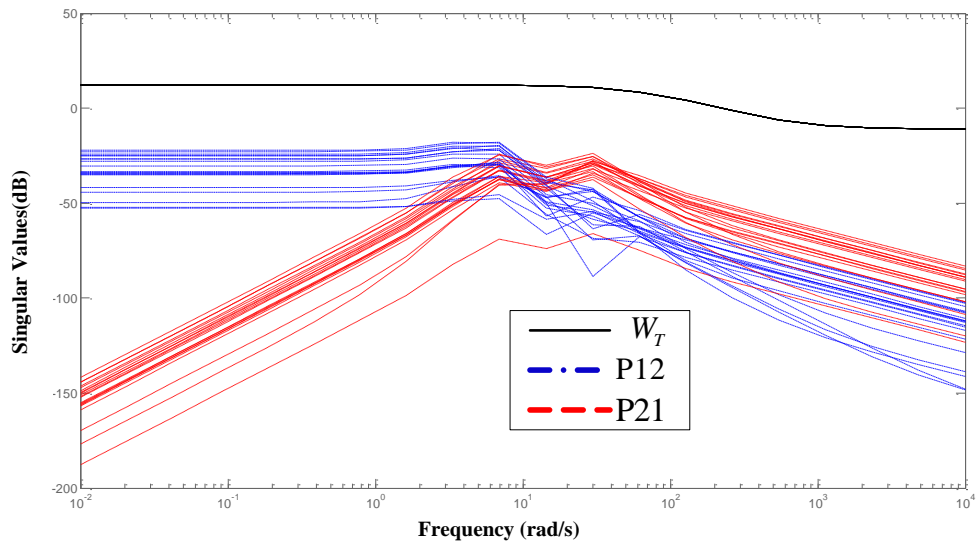| Controllers	Input (u_2)
y_1	$K_{21} = \dfrac{70.9165(s+3.055)(s+1.149)(s+1.124)(s^2+10.73s+1290)}{(s+3.055)(s+0.7148)(s^2+0.5414s+30.68)(s^2+10.32s+1269)}$
y_2	$K_{22} = \dfrac{19564.5581(s+0.5583)(s+1.149)(s^2+10.32s+1270)(s^2+0.5425s+30.68)}{(s+3.055)(s+0.7148)(s^2+0.5414s+30.68)(s^2+10.32s+1269)}$

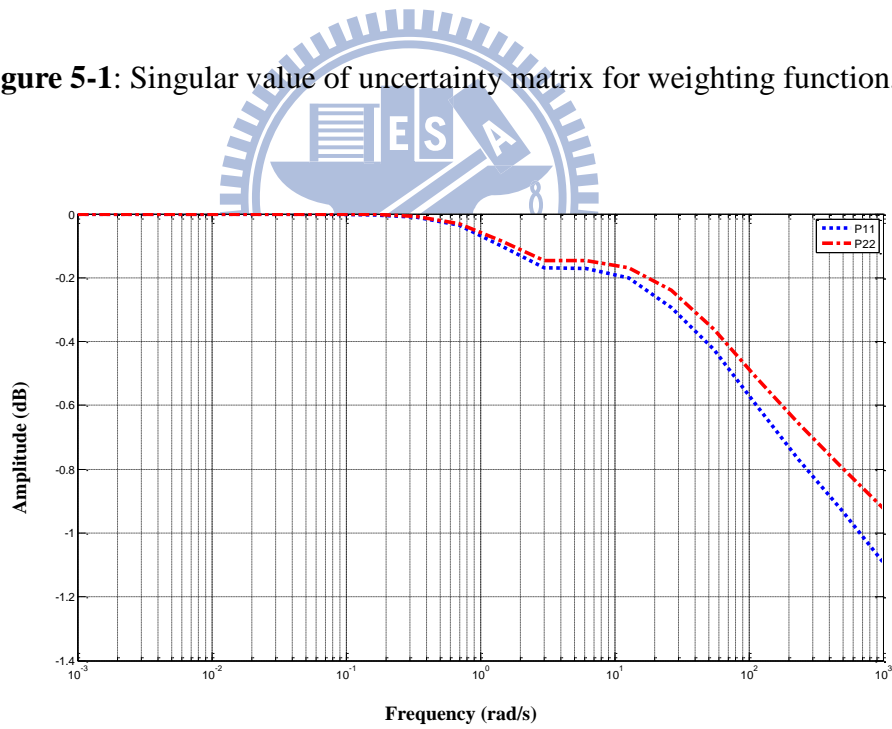**Figure 5-1**: Singular value of uncertainty matrix for weighting function, W_T.
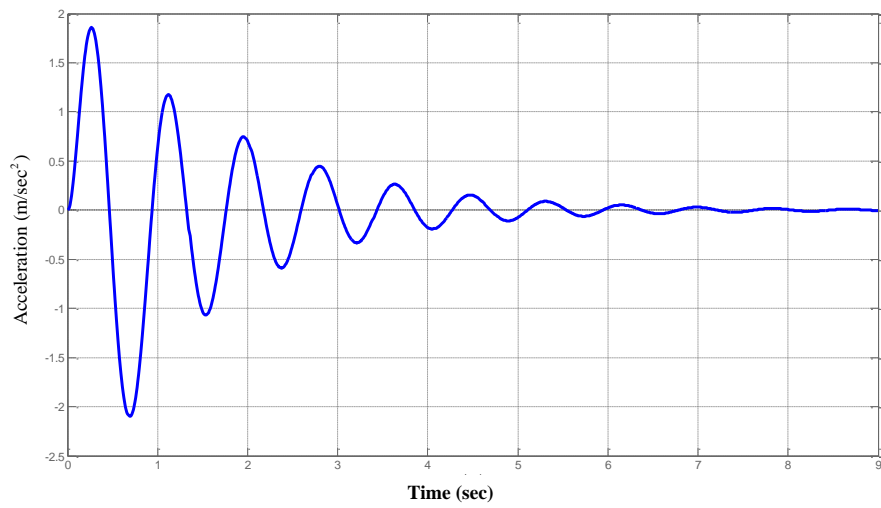

Figure 5-2: Sensitivity function variation

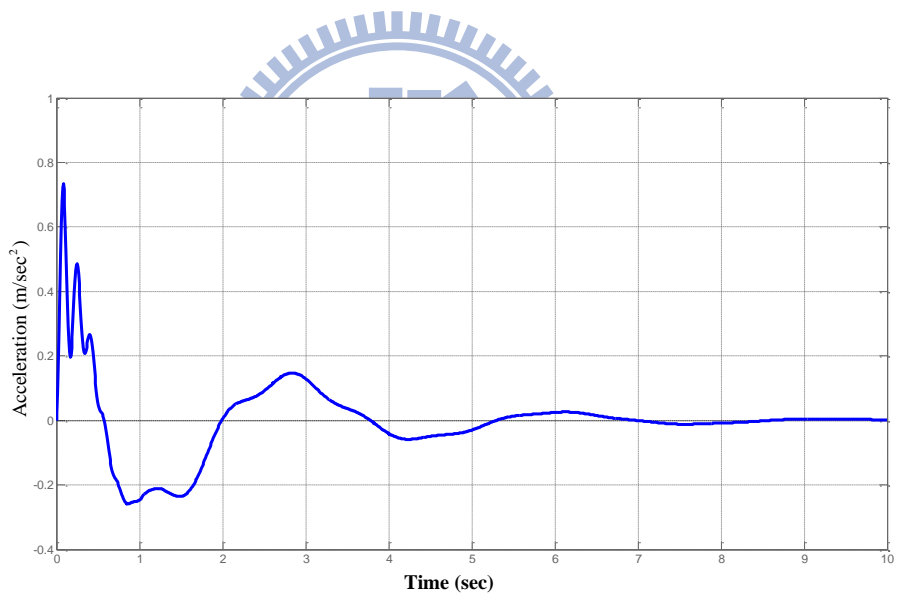**Figure 5-3:** The acceleration response of y_1

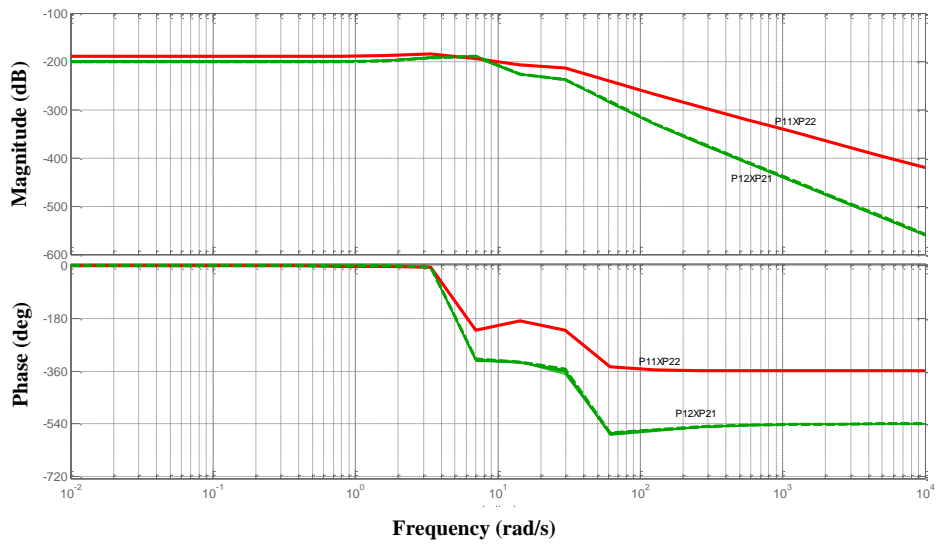**Figure 5-4:** The acceleration response of y_2.

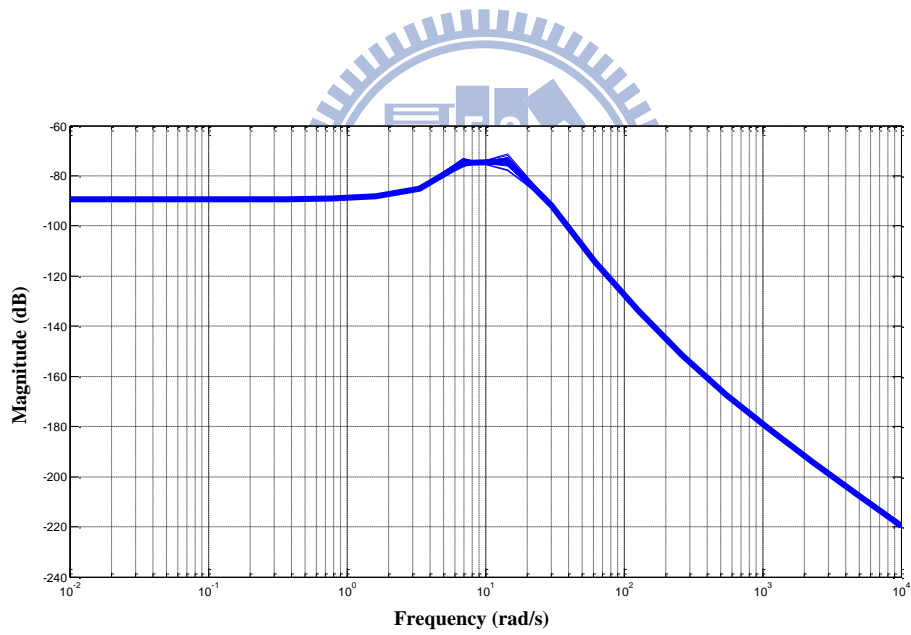**Figure 5-5:** The Bode plot of $P_{11} * P_{22}$ and $P_{12} * P_{21}$
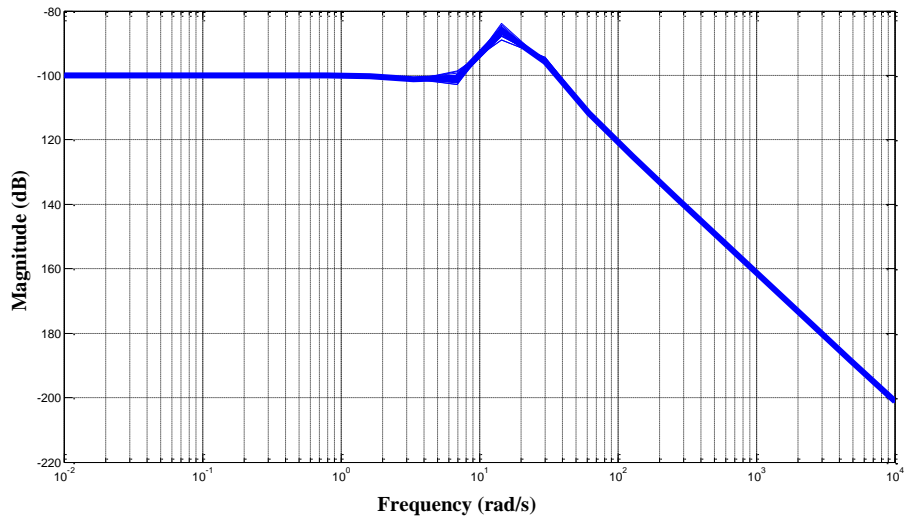

Figure 5-6: The Bode plot of plant P_{11} variation
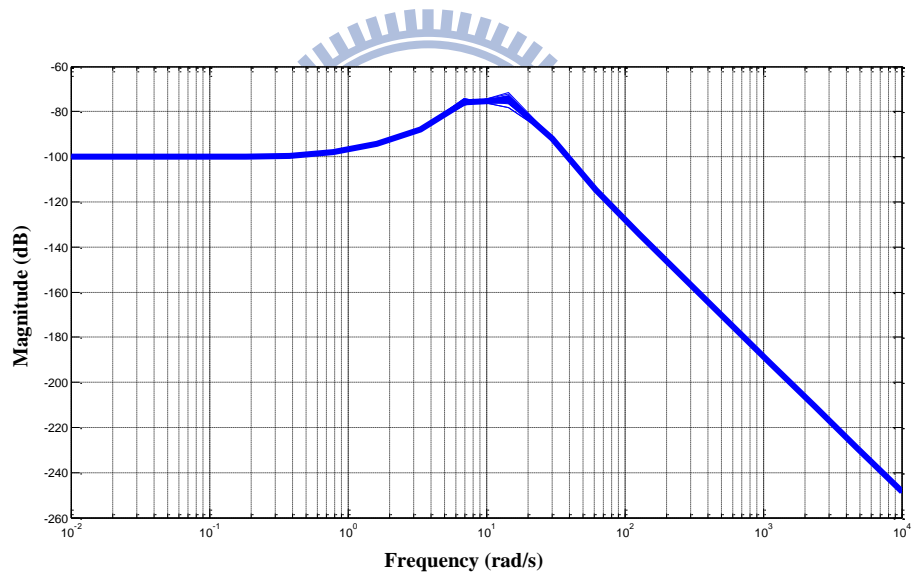
Figure 5-7: The Bode plot of plant P_{12} variation

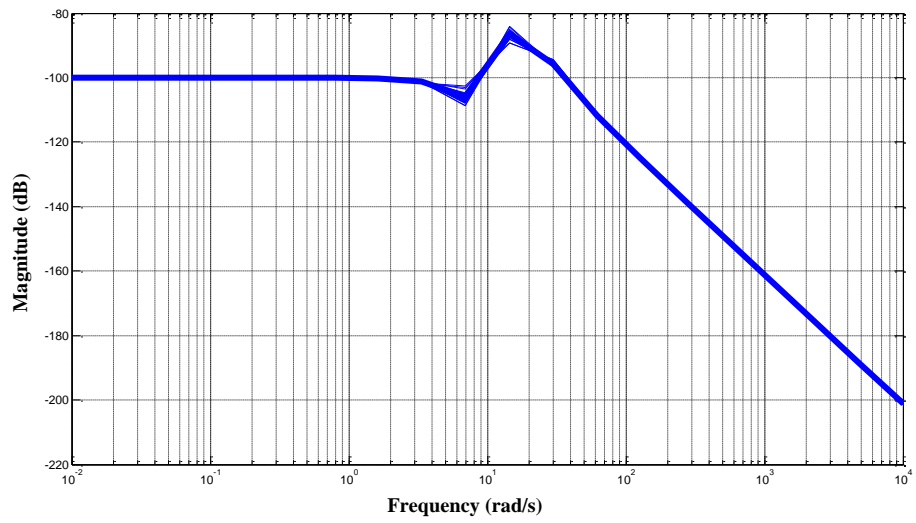**Figure 5-8:** The Bode plot of plant P_{21} variation

Figure 5-9: The Bode plot of plant P_{22} variation

Figure 5-10: The Bode plot of plant by B_1 variations
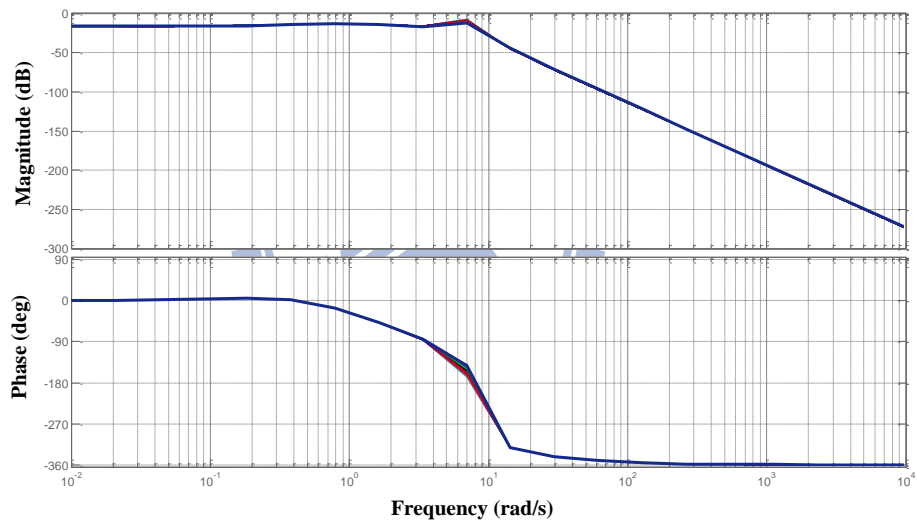
Figure 5-11: The Bode plot of plant by B_2 variations
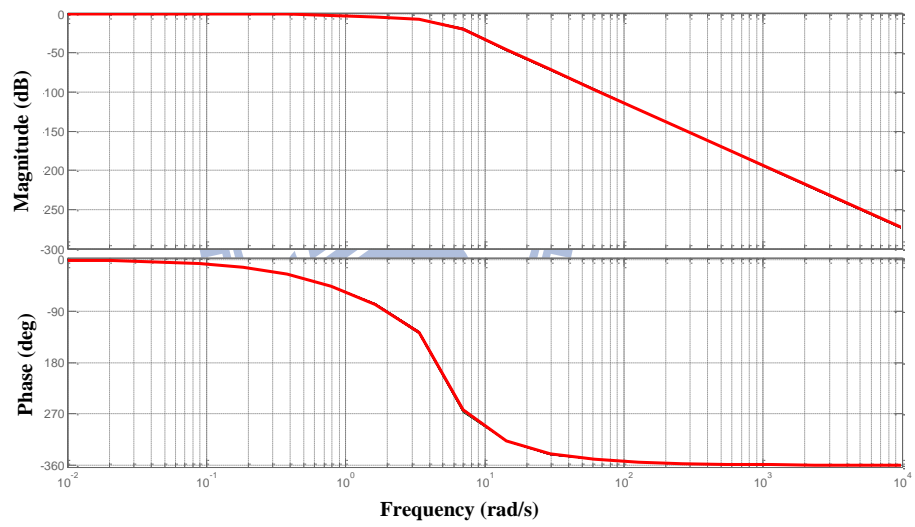
Figure 5-12: The Bode plot of plant by K_1 variations
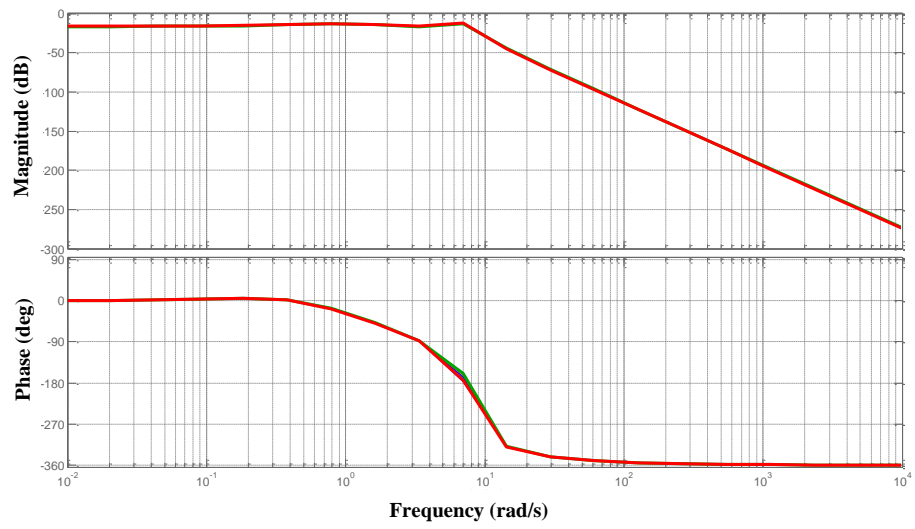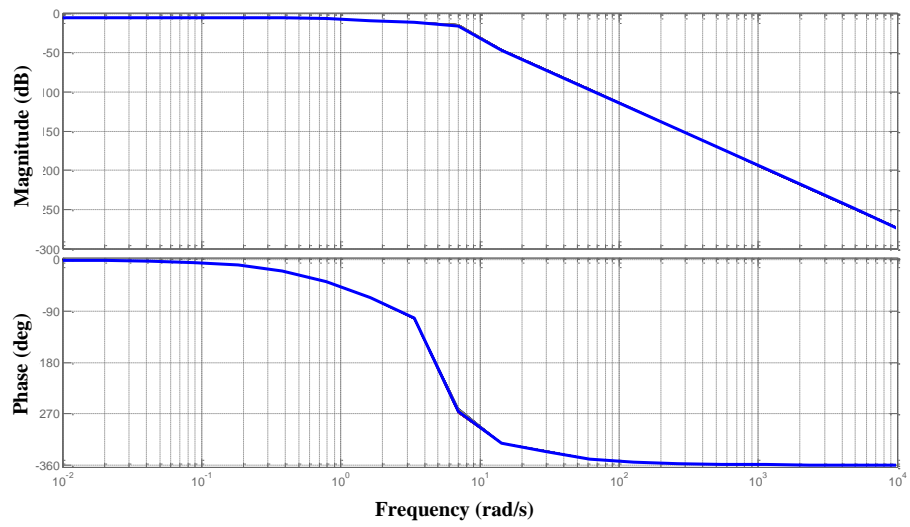
Figure 5-13: The Bode plot of plant by K_2 variations
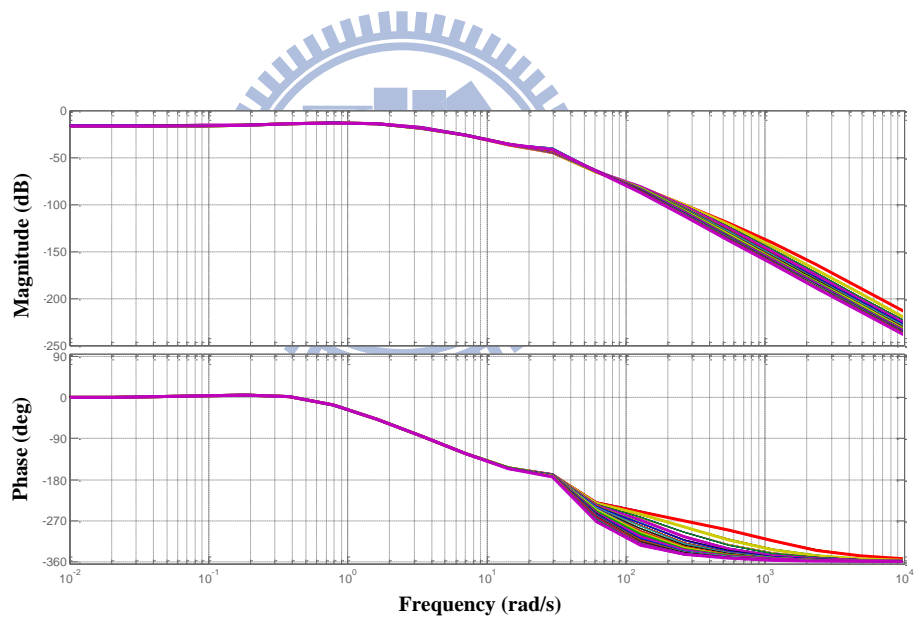
Figure 5-14: The Bode plot of plant by M_1 variations
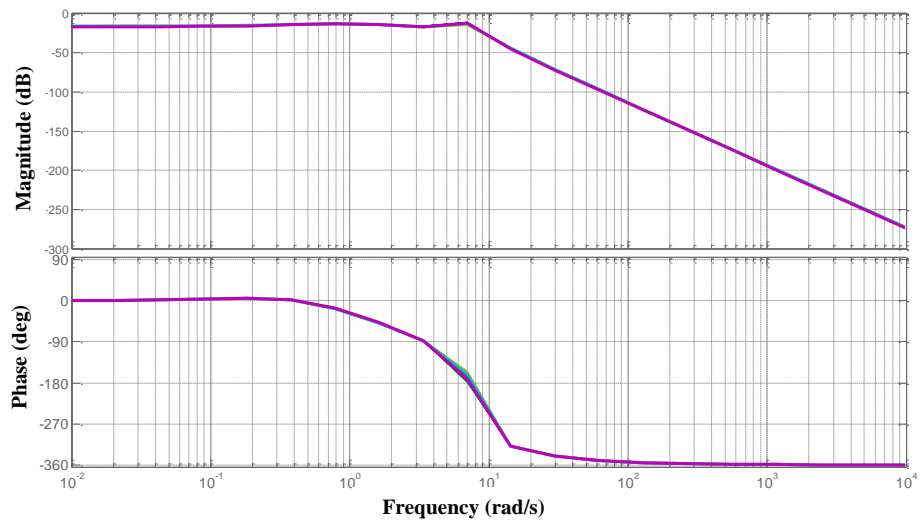
Figure 5-15: The Bode plot of plant by M_2 variations
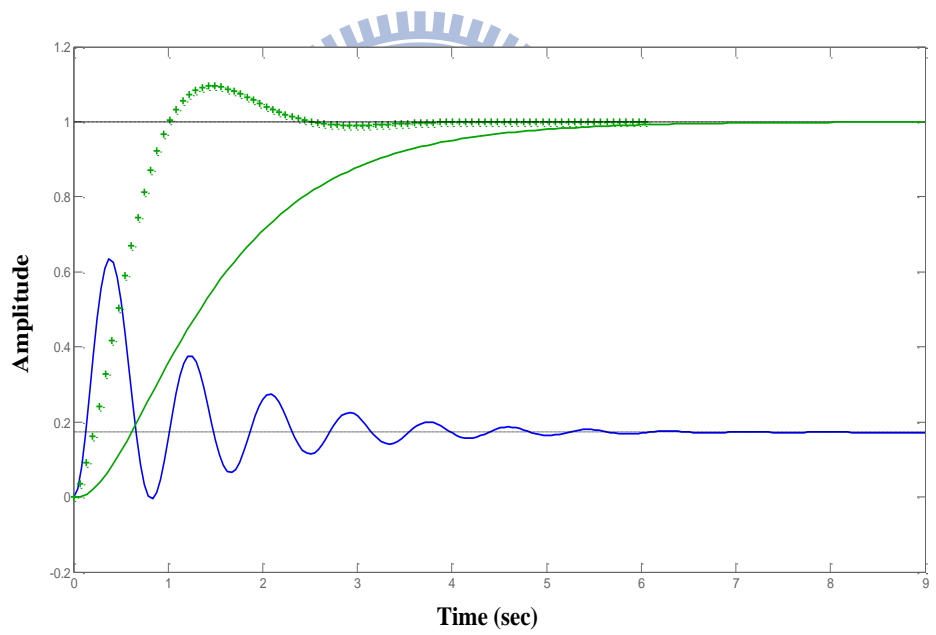
Figure 5-16: y_1 response without prefilter in time domain
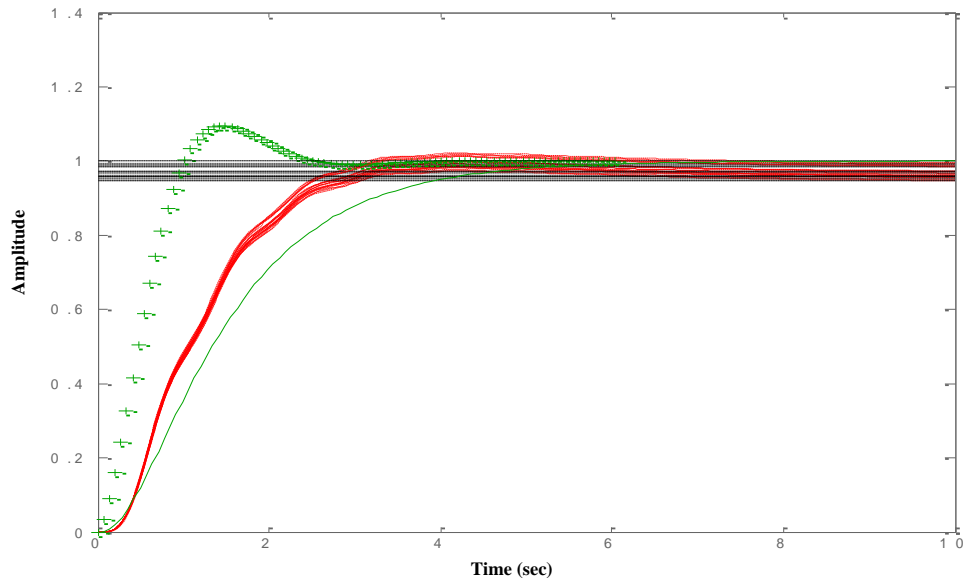
Figure 5-17: y_1 response with prefilter in time domain
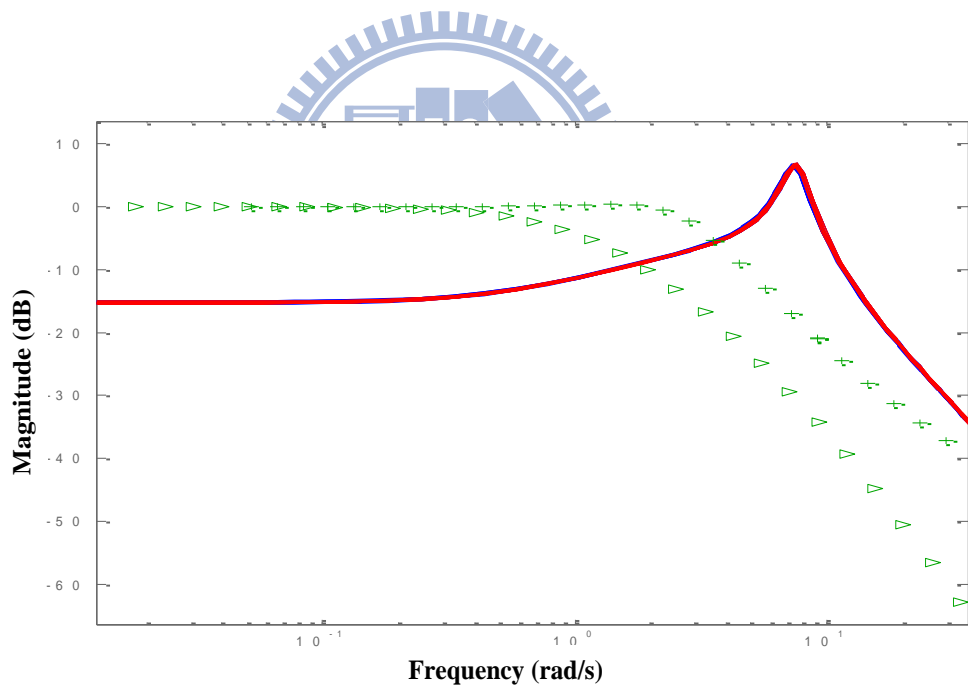
Figure 5-18: y_1 response without prefilter in frequency domain
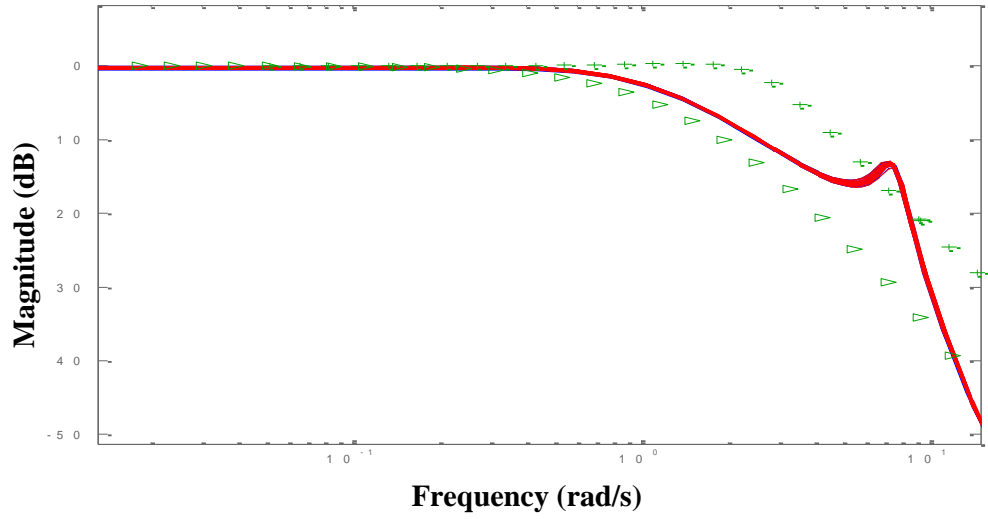
Figure 5-19: y_1 response with prefilter in frequency domain
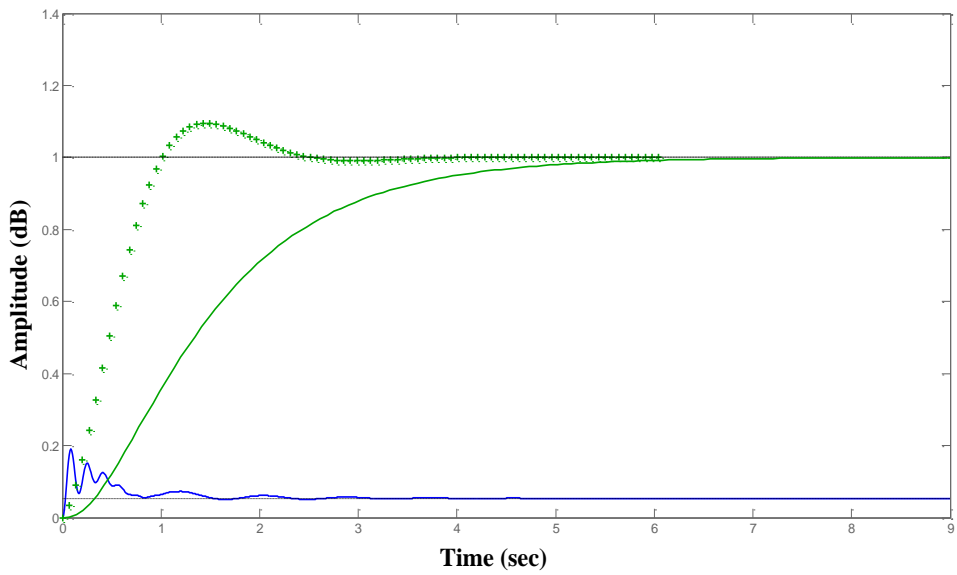
Figure 5-20: y_2 response without prefilter in time domain
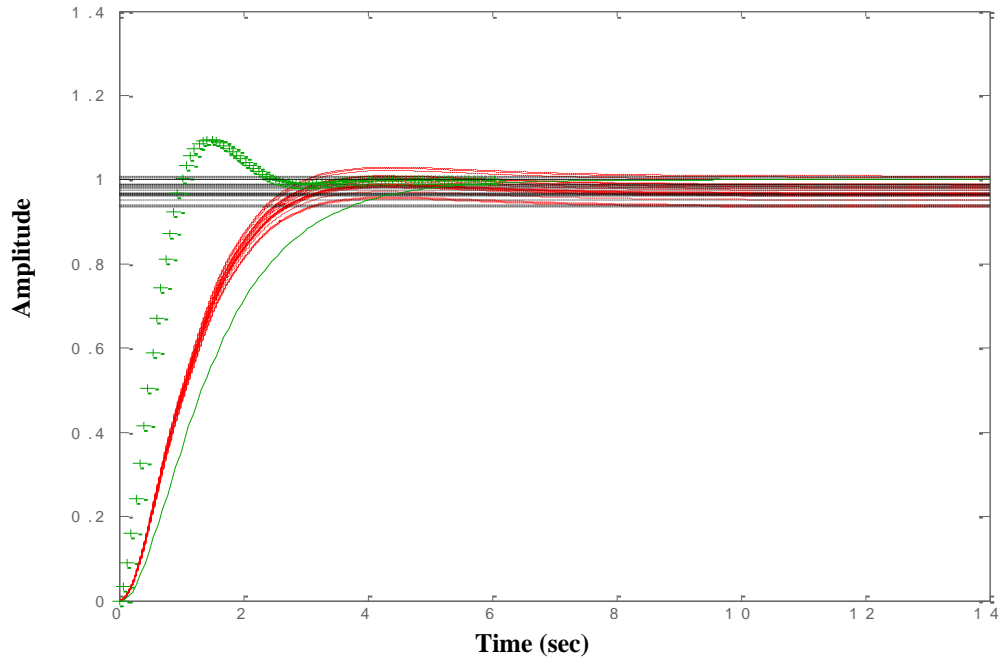
Figure 5-21: y_2 response with prefilter in time domain
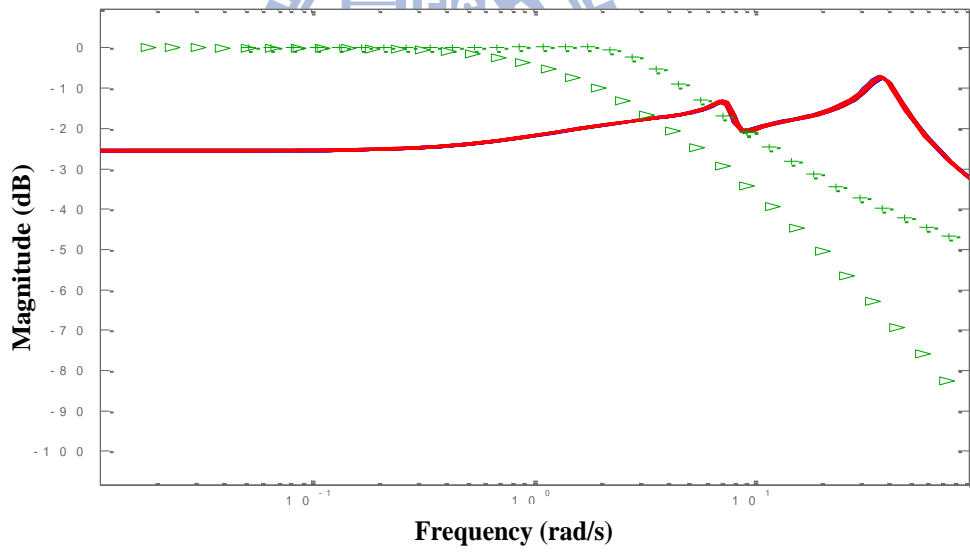
Figure 5-22: y_2 response without prefilter in frequency domain
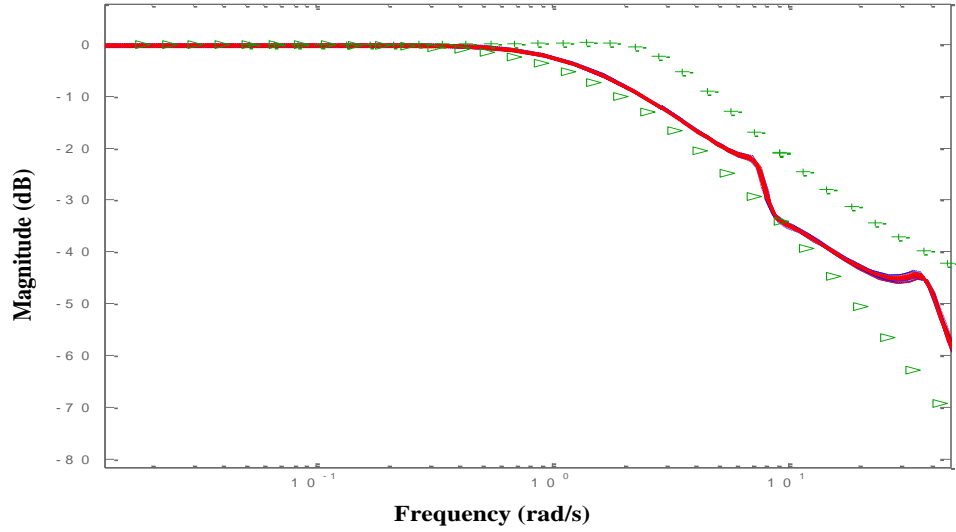
Figure 5-23: y_2 response with prefilter in frequency domain

Next, the stability analysis of MIMO Quarter-car vehicle suspension system is investigated by using Routh-Hurwitz stability criterion [26]. Moreover, in this case, the procedure in Routh's stability criterion is as follows:

The characteristic polynomial in s of MIMO suspension system is expressed as

$$P(s) = (sI - A)$$
$$= \frac{K_1 K_2}{M_1 M_2} + \frac{(B_1 K_2 + B_2 K_1)}{M_1 M_2} s + \frac{B_1 B_2 + K_1 M_1 + K_1 M_2 + K_2 M_1}{M_1 M_2} s^2 + \frac{(B_1 M_1 + B_1 M_2 + B_2 M_1)}{M_1 M_2} s^3 + s^4, \quad (5.1)$$

where the coefficients are real quantities. The stability can be demonstrated by Routh's table, i.e.,

$$\begin{array}{lll}
1 & (B_1 B_2 + K_1 M_1 + K_1 M_2 + K_2 M_1)/(M_1 M_2) & (K_1 K_2)/(M_1 M_2) \\
(B_1 M_1 + B_1 M_2 + B_2 M_1)/(M_1 M_2) & (B_1 K_2 + B_2 K_1)/(M_1 M_2) & 0 \\
D & (B_1 K_1 K_2 M_1 + B_1 K_1 K_2 M_2 + B_2 K_1 K_2 M_1)/(M_1 M_2 (B_1 M_1 + B_1 M_2 + B_2 M_1)) & 0 \\
E & 0 & 0 \\
F & 0 & 0
\end{array} \quad (5.2)$$

where the coefficient D is denoted as

$$D = \frac{(B_1 B_2^2 M_1 + B_1^2 B_2 M_1 + B_1^2 B_2 M_2 + B_1 K_1 M_1^2 + B_1 K_1 M_2^2 + B_1 K_2 M_1^2 + B_2 K_1 M_1^2 + B_2 K_2 M_1^2 + 2 B_1 K_1 M_1 M_2)}{M_1 M_2 (B_1 M_1 + B_1 M_2 + B_2 M_1)}, \text{ and}$$

then $E = \dfrac{numE}{denE}$, the coefficient of numerator $numE$ is represented as

61

$$numE = M_1 M_2 \left(B_1 M_1 + B_1 M_2 + B_2 M_1 \right) \times \left(B_1^3 B_2 K_2 M_1 + B_1^3 B_2 K_2 M_2 + B_1^2 B_2^2 K_1 M_1 + B_1^2 B_2^2 K_1 M_2 \right.$$
$$+ B_1^2 B_2^2 K_2 M_1 + B_1^2 K_2^2 M_1^2 + B_1 B_2^3 K_1 M_1 + B_1 B_2 K_1^2 M_1^2 + 2 B_1 B_2 K_1^2 M_1 M_2$$
$$\left. + B_1 B_2 K_1^2 M_2^2 - 2 B_1 B_2 K_1 K_2 M_1 M_2 + B_1 B_2 K_2^2 M_1^2 + B_2^2 K_1^2 M_1^2 \right)$$

and the coefficient of denominator *denE* is indicated as

$$denE = (B_1 M_1^2 M_2^3 + B_1 M_1^3 M_2^2 + B_2 M_1^3 M_2^2) \times \left(B_1 B_2^2 M_1 + B_1^2 B_2 M_1 + B_1^2 B_2 M_2 + B_1 K_1 M_1^2 \right.$$
$$\left. + B_1 K_1 M_2^2 + B_1 K_2 M_1^2 + B_2 K_1 M_1^2 + B_2 K_2 M_1^2 + 2 B_1 K_1 M_1 M_2 \right)$$

where next $F = \dfrac{numF}{denF}$, the coefficient of numerator *numF* is represented as

$$numF = \left(B_1 M_1^2 M_2^3 + B_1 M_1^3 M_2^2 + B_2 M_1^3 M_2^2 \right) \times \left(B_1 B_2^2 M_1 + B_1^2 B_2 M_1 + B_1^2 B_2 M_2 + B_1 K_1 M_1^2 + B_1 K_1 M_2^2 + B_1 K_2 M_1^2 \right.$$
$$+ B_2 K_1 M_1^2 + B_2 K_2 M_1^2 + 2 B_1 K_1 M_1 M_2 \left. \right) \times \left(B_1^3 B_2 K_1 K_2^2 M_1 + B_1^3 B_2 K_1 K_2^2 M_2 + B_1^2 B_2^2 K_1^2 K_2 M_1 + B_1^2 B_2^2 K_1^2 K_2 M_2 + B_1^2 B_2^2 K_1 K_2^2 M_1 \right.$$
$$+ B_1^2 K_1 K_2^3 M_1^2 + B_1 B_2^3 K_1^2 K_2 M_1 + B_1 B_2 K_1^3 K_2 M_1^2 + 2 B_1 B_2 K_1^3 K_2 M_1 M_2 + B_1 B_2 K_1^3 K_2 M_2^2$$
$$\left. - 2 B_1 B_2 K_1^2 K_2^2 M_1 M_2 + B_1 B_2 K_1 K_2^3 M_1^2 + B_2^2 K_1^3 K_2 M_1^2 \right)$$

and the coefficient of denominator *denF* is indicated as shown

$$denF = M_1 M_2 \left(B_1 M_1 + B_1 M_2 + B_2 M_1 \right) \left(B_1 K_1 M_1^2 M_2^4 + 2 B_1 K_1 M_1^3 M_2^3 + B_1 K_1 M_1^4 M_2^2 + B_1 K_2 M_1^4 M_2^2 \right.$$
$$\left. + B_2 K_1 M_1^4 M_2^2 + B_2 K_2 M_1^4 M_2^2 + B_1 B_2^2 M_1^3 M_2^2 + B_1^2 B_2 M_1^2 M_2^3 + B_1^2 B_2 M_1^3 M_2^2 \right)$$
$$\times (B_1^3 B_2 K_2 M_1 + B_1^3 B_2 K_2 M_2 + B_1^2 B_2^2 K_1 M_1 + B_1^2 B_2^2 K_1 M_2 + B_1^2 B_2^2 K_2 M_1 + B_1^2 K_2^2 M_1^2 + B_1 B_2^3 K_1 M_1$$
$$+ B_1 B_2 K_1^2 M_1^2 + 2 B_1 B_2 K_1^2 M_1 M_2 + B_1 B_2 K_1^2 M_2^2 - 2 B_1 B_2 K_1 K_2 M_1 M_2 + B_1 B_2 K_2^2 M_1^2 + B_2^2 K_1^2 M_1^2)$$

Moreover, because all expressions of the 1st column in Eq. (5.2) of the numerator and

denominator have no sign change, the system is stable.

Chapter 6

CONCLUSION AND FUTURE WORKS

This dissertation develops controllers for one MIMO suspension system, effectively using the QFT/H_∞ approach design procedure. The verification results demonstrate the effectiveness of the MIMO suspension operation scheme, and conform to the requirements. The MIMO suspension system with the developed QFT/H_∞ controller and considered in the presence of disturbances and the noise signals maintain the displacement motion approach to zero subject to the given uncertainty components. The current study offers the key finding for ride quality and handling performance in applying the vehicle suspension system. This research can extend the state-space motion equations of the quarter model to the full-car suspension system. Constructing the dynamical system for the full-car model is more complicated than the quarter model system, and needs to consider translational and rotational motions. This work discusses the possibilities of full-car model MIMO suspension system with its parameter uncertainty responses, and uses the QFT/H_∞ optimization algorithm design-technique to develop the controller. The control of a slalom vehicle under different roll conditions will also be more challenge work in the future.

References:

[1] A. Giua, M. Melas, C. Seatzu, G. Usai, "Design of a predictive semi-active suspension system," *Vehicle System Dynamics*, Vol. 41, No. 4, pp. 277–300, 2004.

[2] A. Giua, C. Seatzu, G. Usai, "Semi-active suspension design with an Optimal Gain Switching target", *Vehicle System Dinamics*, Vol. 31, pp. 213–232, 1999.

[3] K. J. Kitching, D. J. Cole, D. Cebon, "Performance of Semi-Active Damper for Heavy Vehicles", *ASME Journal of Dynamic Systems Measurement and Control,* Vol. 122, pp. 498-506, 2000.

[4] V. Roberti, B. Ouyahia, A. Devallet, "Oleopneumatic suspension with preview semi-active control law", *Proc. Int. Cong. MV2, Active Control in Mechanical Engineering(Lyon, France)*, 1,1993.

[5] Sun Tao, Zhen-yu Huang, Da-yue Chen, and Tang Lei, "Signal frequency based self-tuning fuzzy controller for semi-active suspension system", *Journal of Zhejiang University Science*, 4(4), pp. 426-432, 2003.

[6] Lan Bo, Yu Fan, "Design and Simulation Analysis of LQG Controller of Active Suspension", *Transactions of the Chinese Society of Agricultural Machinery*, pp. 45-49, 2004.

[7] E. Goring, E.C. von Glasner, R. Povel, P. Schutzner, "Intelligent suspension systems for commercial vehicles", *Proc. Int. Cong. MV2, Active Control in Mechanical Engineering (Lyon, France)*, pp. 1–12,1993.

[8] H. Yu, T. C. Tang, D. Rigas, and B. V. Jayawant, "Modeling and Control of Magnetic Suspension Systems", *Proceedings of the 2002 IEEE International Conference on Control Applications*, pp. 944-949, 2002.

[9]   I. M. Horowitz, Synthesis of Feedback Systems, *Academic Press*, New York, 1963.

[10] I. M. Horowitz, Quantitative Feedback Design Theory (QFT), *QFT Publishers*, Colorado, 1992.

[11] C. H. Houpis, and, S. J. Rasmussen, Quantitative Feedback Theory Fundamentals and Applications, *Marcel Dekker, Inc*., New York, 1999.

[12] I. M. Horowitz, and M. J. Sidi, "Synthesis of feedback systems with large plant ignorance for prescribed time-domain tolerances", *Int. J. of Control*, 16(2), pp. 287-309, 1972.

[13] B. C. Kuo, Automatic Control Systems, 7th Edition, *Prentice Hall*, 1995.

[14] Bao-Tung Lin, and Ching-Cheng Teng "Translation of Time-Domain Bounds to Frequency-Domain Bounds in QFT Design", *Proc. of CACS Automatic Control Conference*, Tainan, Taiwan, pp.280-285, Nov. 2002.

[15] K. Glover, and J. C. Doyle, "State-space formulae for all stabilizing controller that satisfy an H_∞-norm bound and relations to risk sensitivity", *Systems & Control Letters*, Vol. 11, pp. 167-172, 1988.

[16] J. C. Doyle, K. Glover, P. P. Khargonekar, and B. A. Francis, "State-space solutions to standard H_2 and H_∞ control problems", *IEEE Trans. Automatic Control*, Vol. 34, No. 8, pp. 831-847, 1989.

[17] K. Zbou, and J. C. Doyle, Essential of Robust Control, *Louisiana State University, California Institute of Technology,* Prentice Hall, Upper Saddle River, New Jersey.

[18] M. J. Sidi, "A combined QFT/H_∞ design technique for TDOF uncertain feedback

systems", *Int. J. of Control*, 75(7), pp. 475-489, 2002.

[19] B. C. Wang, *Synthesis of Multiple Loops Control System*, *Hsin Jyh Book Company*, Taipei, 1994.

[20] P. Gahinet, and P. Apkarian, "A linear matrix inequality approach to H_∞ Control", *Int. J. Robust and Nonlinear Control*, Vol.4 , pp. 421-448, 1994

[21] K. A. Morris, Introduction to Feedback Control, *Harcourt/Academic Press,* Massachusetts, 2001.

[22] Bao-Tung Lin, Han-Min Chang, and Ching-Cheng Teng "A Combined QFT / H_∞ Design Technique", *Proc. of CACS Automatic Control Conference*, DaYeh University, Changhua, Taiwan, 2004.

[23] MATLAB 6.0: User's guide the math works, Inc., 1997.

[24] J. L. Adcock, "Curve Fitter for Pole-Zero Analysis", *Hewlett-Packard Journal*, pp. 33-36, 1987.

[25] A. Shariati, H.D. Taghirad and A. Fatehi, "Decentralized Robust H_∞ Controller Design for a Half-Car Active Suspension System Control", *University of Bath, UK*, September, ID-216, 2004.

[26] Katsuhiko Ogata, Modern control engineering, 2nd edition, Prentice-Hall International Edition.

VITA

JULY 19, 2010

PERSONAL DATA

Name:	Bao-Tung Lin
Date of Birth:	July 26, 1957

EDUCATION

1977/9~1981/6	Receive the B.S. degree in the department of Industrial Education from National Taiwan Normal University.
1992/9~1994/6	Receive the M.S. degree in the institute of Control Engineering from National Chiao-Tung University.
1990/9~2010/7	Receive the Phd. Degree in the institute of Electrical Control Engineering from National Chiao-Tung University.

Research field:

Quantitative feedback control theorem and H-infinity robust control.

PUBLICATION LISTS

Journals:

[1] **Bao-Tung Lin**, and Ching-Cheng Teng, "A Self-Tung Approach to Determining Controller Specification Using A Combined QFT / H_∞ Design Technique" *Advances in Differential Equations and Control Processes*, Vol. 4, No. 2, pp. 95-111, 2009.

[2] **Bao-Tung Lin**, and Ching-Cheng Teng, "A New Design Approach for Low-Order Robust Controllers Based on The Concept of Ordinal Optimization" *International Journal of*

Numerical Methods and Applications, Vol. 3, No. 1, pp. 25-39, 2010.

[3] **Bao-Tung** Lin, Jui-I Tsai, and Ching-Cheng Teng, "QFT / H_∞ Controller Design of a MIMO Suspension System" *Advances in Differential Equations and Control Processes*, Vol. 5, No. 1, pp.49-63, 2010.

[4] **Bao-Tung Lin**, Shih-Meng Chang, and Ching-Cheng Teng "Stability of Ground Vehicles with Steering Angle and Driver-Like Controllers" *International Journal of Electrical Engineering*, Vol. 17, Number 2, PP.161-168, 2010.

[5] Jui-I Tsai, **Bao-Tung Lin**, and Ching-Cheng Teng "An ASIC Implementation for testing of a ladder diagram using a bollean petri net" *Far East Journal of Experimental and Theoretical Artificia lIntelligence*, 2010.

Conferences:

[1] **Bao-Tung Lin**, and Ching-Cheng Teng "Translation of Time-Domain Bounds to Frequency-Domain Bounds in QFT Design", *Proc. of CACS Automatic Control Conference*, Tainan, Taiwan, pp.280-285, Nov. 2002.

[2] **Bao-Tung Lin**, Han-Min Chang, and Ching-Cheng Teng "A Combined QFT / H_∞ Design Technique", *Proc. of CACS Automatic Control Conference*, DaYeh University, Changhua, Taiwan, 2004.

[3] Wen-Nan Huang, **Bao-Tung Lin**, and Ching-Cheng Teng "Self-Tuning Approach of Interval Specification Determination Applying Combined QFT / H_∞ Design Technique", *Proc. of CACS Automatic Control Conference*, Nan Tai University, Tainan, Taiwan, 2005.

[4] **Bao-Tung Lin**, Wen-Nan Huang, and Ching-Cheng Teng "Design of Controllers for MIMO Active-Suspension System Using Combined QFT / H_∞ Approach", *Proc. of CACS Automatic Control Conference*, Tamsui, Taiwan, pp.686-691 Nov. 2006.

Appendix

Program:

```
clear;
nn1 = -5;%[-5%]
nn2 = 4; %[+4%]
m1 = ureal('m1',973,'Percentage',[nn1 nn2]);
m2 = ureal('m2',114,'Percentage',[nn1 nn2]);
k1 = ureal('k1',42720,'Percentage',[nn1 nn2]);
k2 = ureal('k2',101115,'Percentage',[nn1 nn2]);%[system parameters]
c1 = ureal('c1',1095,'Percentage',[nn1 nn2]);
c2 = ureal('c2',14.6,'Percentage',[nn1 nn2]);
%-----------------------%
A = [0 0 1 0
     0 0 0 1
     -k1/m1 k1/m1 -c1/m1 c1/m1
     k1/m2 -(k1+k2)/m2 c1/m2 -(c1+c2)/m2];
B = [0 0;0 0;1/m1 0;0 1/m2];
C = [1 0 0 0;0 1 0 0];
D = zeros(2,2);
plant = uss(A,B,C,D);
%--------------------------------%
nn = 20;[sampling number]
fw =logspace(-2,4,nn);
Bplant = usample(plant,nn);
B_siz = size(Bplant,3);
for iB_siz = 1:B_siz
bode(tf(plant.nomi(1,1))*tf(plant.nomi(2,2)),'r',tf(Bplant(1,2,iB_siz))*tf(Bplant(2,1,iB_siz)),'g
--',fw)
hold on
end
%% to deterimine weighting function WT (W1)
B_sizarr = size(Bplant,3);
for iB_sizarr = 1:B_sizarr
```
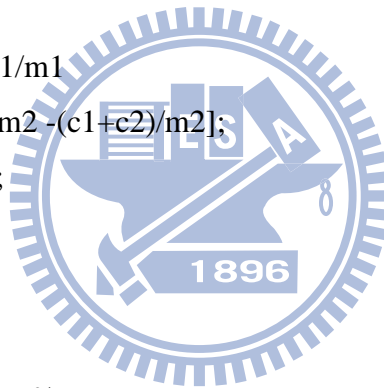
```matlab
PSS025_arr(:,:,iB_sizarr) = (tf(Bplant(:,:,iB_sizarr))-tf(plant.nomi))*inv(tf(plant.nomi));
        [magPSS025(:,:,:,iB_sizarr),phPSS025(:,:,:,iB_sizarr)] =
bode(PSS025_arr(:,:,iB_sizarr),fw);
                    magBP025_11(:,1,iB_sizarr) = magPSS025(1,1,:,iB_sizarr);
                magf_BP02511(:,iB_sizarr) = magBP025_11(:,1,iB_sizarr);
                    magBP025_12(:,1,iB_sizarr) = magPSS025(1,2,:,iB_sizarr);
                magf_BP02512(:,iB_sizarr) = magBP025_12(:,1,iB_sizarr);
                    magBP025_21(:,1,iB_sizarr) = magPSS025(2,1,:,iB_sizarr);
                magf_BP02521(:,iB_sizarr) = magBP025_21(:,1,iB_sizarr);
                    magBP025_22(:,1,iB_sizarr) = magPSS025(2,2,:,iB_sizarr);
                magf_BP02522(:,iB_sizarr) = magBP025_22(:,1,iB_sizarr);


    figure(2);
        bodemag(PSS025_arr(:,:,iB_sizarr),'b',fw)
 hold on;grid on
   title('principal gains of singular value (P-Pn)/Pn    for evaluating Wmpo(w)')
xlabel('frequence response')
ylabel('dB')
        pmag11= magf_BP02511; pmag12= magf_BP02512;
    pmag21= magf_BP02521; pmag22= magf_BP02522;
    %-------------------------------%
n_W = 0.4*[0.4 300];
d_W = [0.55 30];
W = tf(n_W,d_W);
    figure(3)
     sigma(PSS025_arr(2,1,iB_sizarr),'r--',PSS025_arr(1,2,iB_sizarr),'b-.',W,'k',fw)
      hold on
end
%% plant uncertainty variations
for iB_sizarr = 1:B_sizarr
PSS025_arrB(:,:,iB_sizarr) = tf(Bplant(:,:,iB_sizarr));
        [magPSS025B(:,:,:,iB_sizarr),phPSS025B(:,:,:,iB_sizarr)] =
bode(PSS025_arrB(:,:,iB_sizarr),fw);
                    magBP025B_11(:,1,iB_sizarr) = magPSS025B(1,1,:,iB_sizarr);
```

```matlab
            magf_BP025B_11(:,iB_sizarr) = magBP025B_11(:,1,iB_sizarr);
                magBP025B_12(:,1,iB_sizarr) = magPSS025B(1,2,:,iB_sizarr);
            magf_BP025B_12(:,iB_sizarr) = magBP025B_12(:,1,iB_sizarr);
                magBP025B_21(:,1,iB_sizarr) = magPSS025B(2,1,:,iB_sizarr);
            magf_BP025B_21(:,iB_sizarr) = magBP025B_21(:,1,iB_sizarr);
                magBP025B_22(:,1,iB_sizarr) = magPSS025B(2,2,:,iB_sizarr);
            magf_BP025B_22(:,iB_sizarr) = magBP025B_22(:,1,iB_sizarr);


        figure(4);
            bodemag(tf(Bplant(:,:,iB_sizarr)),'b',fw)
    hold on;grid on
        title('uncertainty variation (P))')
xlabel('frequence response')
ylabel('dB')
end
        %%% Wmp=max value (magnitude)
max11=max(magf_BP025B_11);
max12=max(magf_BP025B_12);
max21=max(magf_BP025B_21);
max22=max(magf_BP025B_22);


        %%% wmp=min value (magnitude)
min11=min(magf_BP025B_11);
min12=min(magf_BP025B_12);
min21=min(magf_BP025B_21);
min22=min(magf_BP025B_22);


%%   upper and lower bound
    fw1 = logspace(-3,3,nn);
%%%%%%%%%% upper bound and lower bound
tu11=tf(0.338*[1,20],[1 3.12 6.76]);
tl11=tf(15*[1.96],[conv([1 15],[1 3.08 1.96])]);
%---------------------------%
[mag_tu11,ph_tu11] = bode(tu11,fw1);
```
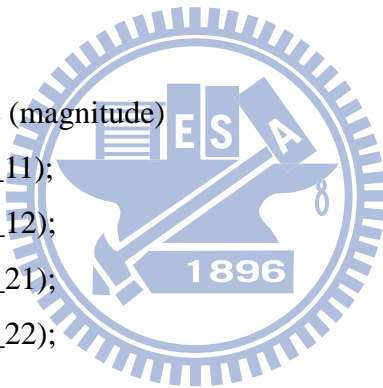
```matlab
[mag_tl11,ph_tl11] = bode(tl11,fw1);


mag_del_t11 = mag2db(mag_tu11)-mag2db(mag_tl11);
  for idel_11 = 1:length(fw1)
% [mag_delt11,ph_delt11] = bode(tu11-tl11,fw1); % diference of upper & lower bounds
  mag_delt11(idel_11,1) = mag_del_t11(1,1,idel_11);
  end
%%%%%%%%%% upper bound and lower bound
tu22=tu11;
tl22=tl11;
%--------------------------------%
[mag_tu22,ph_tu22] = bode(tu22,fw1);
[mag_tl22,ph_tl22] = bode(tl22,fw1);


mag_del_t22 = mag2db(mag_tu22)-mag2db(mag_tl22);
  for idel_11 = 1:length(fw1)
%[mag_delt22,ph_delt22] = bode(tu22-tl22,fw1);
  mag_delt22(idel_11,1) = mag_del_t22(1,1,idel_11);
  end
%--------------------------------%
%%% sensitivity
ss11 = mag_delt11./mag2db(max11-min11)';
ss22 = mag_delt22./mag2db(max22-min22)';

figure(5);
semilogx(fw1,ss11,'g--',fw1,ss22,'r-.')
%%
%ss11=delt11(:)./delp11(:);ss22=delt22(:)./delp22(:);
  %%%% curve fitting
  sys11 = frd(ss11,fw1);
  sys22 = frd(ss22,fw1);


cfs11 = tf(fitmagfrd(sys11,2));
cfs22 = tf(fitmagfrd(sys22,2));
```
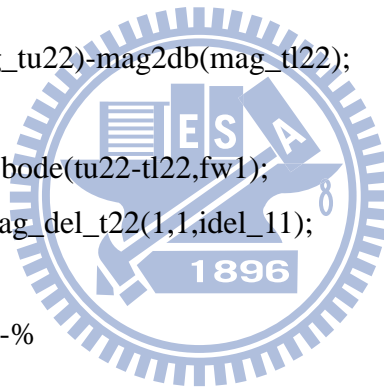
```
[magcfs11,phcfs11] = bode(cfs11,fw1);

n_cfs11 = size(magcfs11,3);

  for is11 = 1:n_cfs11

            magcfs_11(is11,:) = magcfs11(:,1,is11);

  end

[magcfs22,phcfs22] = bode(cfs22,fw1);

n_cfs22 = size(magcfs22,3);

  for is22 = 1:n_cfs22

    magcfs_22(is22,:) = magcfs22(:,1,is22);

    end

%%%% 1/ws

k1=5;    k2=5;

ws11_inv =k1/cfs11;

ws22_inv =k2/cfs22;         % k1 times


%ws11_inv=cfs11;ws22_inv=cfs22;              % one time

[magws11_inv,phws11_inv]=bode(ws11_inv,fw1);

nmagws11_inv = size(magws11_inv,3);

for in_mag11_inv = 1:nmagws11_inv

            magws11f_inv(in_mag11_inv,:) = magws11_inv(:,:,in_mag11_inv);

end


[magws22_inv,phws22_inv]=bode(ws22_inv,fw1);

nmagws22_inv = size(magws22_inv,3);

for in_mag22_inv = 1:nmagws22_inv

            magws22f_inv(in_mag22_inv,:) = magws22_inv(:,:,in_mag22_inv);

end


    ws11 = 1/ws11_inv;

%%%%% w21=ws%%%%%%

figure(6);    % plot curve fitting and 1/ws

subplot(211)

bodemag(ws11,'g',ws11_inv,'b',tu11,'r+',tl11,'r-.',fw1);%semilogx(fw,20*log10(magcfs11(:)),'b

+',fw,20*log10(magws11_inv(:)),'b',fw,20*log10(magcfs22(:)),'g+',fw,20*log10(magws22_in
```

v(:)),'g')

grid on; hold on

   title('principal P11, 1/Ws and Senstivity')

xlabel('frequence   red--p11   blue--1/Ws   green--Senstivity')

ylabel('dB')

%%%%% w22=ws%%%%%%

ws22 = 1/ws22_inv;

subplot(212)

title('principal P11, 1/Ws and Senstivity')

bodemag(ws22,'g',ws22_inv,'b',tu22,'g+',tl22,'g-.',fw1)

grid on; hold on

title('principal P22, 1/Ws and Senstivity')

xlabel('frequence   red--p22   blue-- 1/Ws   green--Senstivity')

ylabel('dB')

%% find K_controller

     %%%   ws-----curve fitting

   sws11=ws11_inv;   sws22=ws22_inv;   %two times

    %   sws11=ltisys('tf',cfd11,conv(cfn11,[1/0.0001

1]));sws22=ltisys('tf',cfd22,conv(cfn22,[1/0.0001 1]));   %one   times

   % sws11=ltisys('tf',[0.7 8.4],[1 0.01]);   sws22=ltisys('tf',[0.7 56],[1 0.01]);   %

new value

 sws = 1.02*append(ws11_inv,ws22_inv);

 %   w1   ------max value=[1 0;0 1]

swT = 1*[W*inv(W) 0;0 W*inv(W)];

     %%wun------try error

    % swun_11=ltisys('tf',0.1,1);swun_22=ltisys('tf',0.1,1);

    swun_11 = tf(120,1); swun_22 = tf(120,1);

   swun = 1*[swun_11 0*swun_11;0*swun_22 swun_22];

%----------------------

systemnames = 'swT plant sws swun';
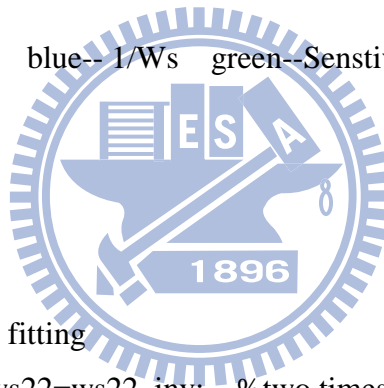
```matlab
inputvar ='[control(2); n(2); d(2)]';

outputvar ='[swun; sws; swT; d+n+plant]';

input_to_plant = '[control]';

input_to_sws = '[-d-n-plant]';

input_to_swun ='[control]';

input_to_swT ='[plant]';

PSS025_ic = sysic;
%================================%
nmeas = 2;

ncon = 2;

hinPSS025_ic = PSS025_ic.nomi;

[PSS_ica,PSS_icb,PSS_icc,PSS_icd] = ssdata(hinPSS025_ic);

hinfPSS025_c = pck(PSS_ica,PSS_icb,PSS_icc,PSS_icd);

 [ns_PSS025,ni_PSS025,no_PSS025] = sinfo(hinfPSS025_c);

%    TF_PSS025 = ssub(hinfPSS025_c,1:2,7:8);

%            [TFaak_PSS025 ,TFbbk_PSS025,TFcck_PSS025,TFddk_PSS025] =
unpck(TF_PSS025);

%    tfPSS025 =
minreal(zpk(ss(TFaak_PSS025 ,TFbbk_PSS025,TFcck_PSS025,TFddk_PSS025)))


addpath(genpath('D:\ExMat_Tools\Control2'))

 [a111,b1,b2,ca,cb,d11,d12,d21,d22]=hinfpar(hinfPSS025_c,[nmeas,ncon]);

   [gopt,K_PSS025] = hinflmi(hinfPSS025_c,[nmeas,ncon]);

 [ns_K,ni_K,no_K] = sinfo(K_PSS025);

   %%% controller

    K_PSS025 = ssub(K_PSS025,1:ni_K,1:no_K);

        [aak_PSS025 ,bbk_PSS025,cck_PSS025,ddk_PSS025] = unpck(K_PSS025);

  tfK_PSS025 = minreal(zpk(ss(aak_PSS025 ,bbk_PSS025,cck_PSS025,ddk_PSS025)))


figure(7);

bodemag(tfK_PSS025)


       %% close loop system without Prefilter and Noise filter

  %%%    K*plant
```

```matlab
    K = -minreal(tfK_PSS025);
  NF = tf(1,[0 1]);
    sNF = append(NF,NF);
  %-------------------
systemnames = 'K plant sNF';
inputvar ='[ r(2); d(2); n(2)]';
outputvar ='[K; plant; d+plant]';
input_to_plant = '[K]';
input_to_sNF = '[r-n-d-plant]';
input_to_K = '[sNF]';
unit_PSS025 = sysic;
%-------------------
[Bunit_PSS025,unit_SamValues] = usample(unit_PSS025,20);
%---------calculation sensor noise T = u/n -----------%
[PSS025_ica,PSS025_unitb,PSS025_unitc,PSS025_unitd] = ssdata(unit_PSS025.nomi);
unitPSS025_c = pck(PSS025_ica,PSS025_unitb,PSS025_unitc,PSS025_unitd);
[ns_unit_PSS025,ni_unit_PSS025,no_unit_PSS025] = sinfo(unitPSS025_c);
tf_Tnoise11 = zpk(tf(unit_PSS025.nomi(1,5)));
tf_Tnoise22 = zpk(tf(unit_PSS025.nomi(2,6)));
figure(8)
subplot(121);pzmap(tf_Tnoise11)
subplot(122);pzmap(tf_Tnoise22)
%--------Input r & Output plant -----------%
figure(9) % step response
subplot(121); step(tf(unit_PSS025.nomi(3,1)),'b',tu11,'g+',tl11,'g')
subplot(122); step(tf(unit_PSS025.nomi(4,2)),'b',tu11,'g+',tl11,'g')
title('step response');hold on
xlabel('time ')
ylabel('magnitude')
%-------Input r & Output plant ----------%
figure(10);
subplot(121);bodemag(unit_PSS025(3,1),'b',Bunit_PSS025(3,1),'r',tu11,'g+',tl11,'g>')
subplot(122);bodemag(unit_PSS025(4,2),'b',Bunit_PSS025(4,2),'r',tu11,'g+',tl11,'g>')
%%
```

```
B_unit_PSS025 = size(Bunit_PSS025,3);
for iB_sizunit_PSS025 = 1:B_unit_PSS025
 [magunitPSS025B(:,:,:,iB_sizunit_PSS025),phunitPSS025B(:,:,:,iB_sizunit_PSS025)] =
bode(Bunit_PSS025(:,:,iB_sizunit_PSS025),fw);
                %%% max value (magnitude) for max|T1| <= Beta %%%
                magBunitP025_31(:,1,iB_sizunit_PSS025) =
magunitPSS025B(3,1,:,iB_sizunit_PSS025);
                magf_BunitP02531(:,iB_sizunit_PSS025) =
magBunitP025_31(:,1,iB_sizunit_PSS025);


                magBunitP025_41(:,1,iB_sizunit_PSS025) =
magunitPSS025B(4,1,:,iB_sizunit_PSS025);
                magf_BunitP02541(:,iB_sizunit_PSS025) =
magBunitP025_41(:,1,iB_sizunit_PSS025);


                magBunitP025_32(:,1,iB_sizunit_PSS025) =
magunitPSS025B(3,2,:,iB_sizunit_PSS025);
                magf_BunitP02532(:,iB_sizunit_PSS025) =
magBunitP025_32(:,1,iB_sizunit_PSS025);


                magBunitP025_42(:,1,iB_sizunit_PSS025) =
magunitPSS025B(4,2,:,iB_sizunit_PSS025);
                magf_BunitP02542(:,iB_sizunit_PSS025) =
magBunitP025_42(:,1,iB_sizunit_PSS025);
                %%% max value (magnitude) for max|Td| <= Gamma %%%
                    magBunitP025_35(:,1,iB_sizunit_PSS025) =
magunitPSS025B(5,3,:,iB_sizunit_PSS025);
                magf_BunitP02535(:,iB_sizunit_PSS025) =
magBunitP025_35(:,1,iB_sizunit_PSS025);


                    magBunitP025_36(:,1,iB_sizunit_PSS025) =
magunitPSS025B(6,3,:,iB_sizunit_PSS025);
                magf_BunitP02536(:,iB_sizunit_PSS025) =
magBunitP025_36(:,1,iB_sizunit_PSS025);
```

```matlab
                magBunitP025_45(:,1,iB_sizunit_PSS025) =
magunitPSS025B(5,4,:,iB_sizunit_PSS025);
                magf_BunitP02545(:,iB_sizunit_PSS025) =
magBunitP025_45(:,1,iB_sizunit_PSS025);


                magBunitP025_46(:,1,iB_sizunit_PSS025) =
magunitPSS025B(6,4,:,iB_sizunit_PSS025);
                magf_BunitP02546(:,iB_sizunit_PSS025) =
magBunitP025_46(:,1,iB_sizunit_PSS025);
end
    %%% max value (magnitude) for max|T1| <= Beta %%%
maxT31 = max(magf_BunitP02531);
maxT32 = max(magf_BunitP02532);
maxT41 = max(magf_BunitP02541);
maxT42 = max(magf_BunitP02542);
 dB_maxT31 = mag2db(maxT31);
  dB_maxT32 = mag2db(maxT32);
  dB_maxT41 = mag2db(maxT41);
  dB_maxT42 = mag2db(maxT42);
%------ Sensitivity function ----%
figure(11);
%%%---- input d(1) and output yd -----%%%
subplot(221);step(tf(unit_PSS025.nomi(5,3)),'b',tf(unit_PSS025.nomi(6,3)),'r');
subplot(223);pzmap(tf(unit_PSS025.nomi(5,3)),tf(unit_PSS025.nomi(6,3)));
%%%---- input d(2) and output yd -----%%%
subplot(222);step(tf(unit_PSS025.nomi(5,4)),'b',tf(unit_PSS025.nomi(6,4)),'r');
subplot(224);pzmap(tf(unit_PSS025.nomi(5,4)),tf(unit_PSS025.nomi(6,4)));
    %%% max value (magnitude) for max|Td| <= Gamma %%%
maxT35 = max(magf_BunitP02535);
maxT36 = max(magf_BunitP02536);
maxT45 = max(magf_BunitP02545);
maxT46 = max(magf_BunitP02546);
 dB_maxT35 = mag2db(maxT35);
```

```matlab
    dB_maxT36 = mag2db(maxT36);
     dB_maxT45 = mag2db(maxT45);
    dB_maxT46 = mag2db(maxT46);
%% closed-loop system with Prefilter
%%prefilter
%pfn11=[1];pfd11=[1];
pfn11= 4*0.5*conv(conv([0 1],[0 2]),[1 2]);
pfd11=conv([1 0.5],[1 9 16]);%[1 6 15 ];


F11 =tf(pfn11,pfd11);
pfn22=2*conv([0 1],[0 2]);
pfd22=[1 1 4];
%pfn22=[15];pfd22=[1 16 15 ];


F22 = tf(pfn22,pfd22);
Pref = append(F11,F22);
%---------------------------------
systemnames = 'Pref K plant';
inputvar ='[ r(2)]';
outputvar ='[plant]';
input_to_plant = '[K]';
input_to_K = '[Pref-plant]';
input_to_Pref = '[r]';
PSS025 = sysic;
%---------------------%
[B_PSS025,SamValues] = usample(PSS025,20);


figure(12)
subplot(121);step(tf(PSS025.nomi(1,1)),'r',tu11,'g+',tl11,'g');
title('unity feedback P11 Bode plot ')
xlabel('frequence ')
ylabel('dB')


subplot(122);step(tf(PSS025.nomi(2,2)),'r',tu11,'g+',tl11,'g');
```
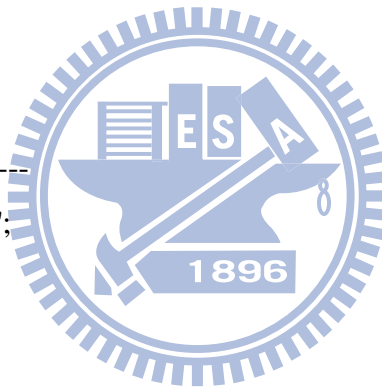
```matlab
title('unity feedback P22 Bode plot ')
xlabel('frequence ')
ylabel('dB')

figure(13)
bodemag(PSS025,'b',B_PSS025,'r',tu11,'g+',tl11,'g>')
%% Acceration calculation ---> s^2*x1
s = tf('s');
Accer_11 = s^2*minreal(tf(PSS025.nomi(1,1)));
Accer_22 = s^2*minreal(tf(PSS025.nomi(2,2)));
figure(14);
subplot(211);step(Accer_11)
subplot(212);step(Accer_22)
%% simulation of m1, m2, k1, k2, c1 and c2 variations
% m1 variations = [924.35, 1011.92];
vari = 5;
m1_vari = 924.35:vari:1011.92;
figure(15);
for im1_vari = 1:length(m1_vari)
m1_PSS025 = usubs(PSS025,'m1',im1_vari);
BAccer_m1 = usample(m1_PSS025,10);
BAccerm1_siz = size(BAccer_m1,3);
for iBAccerm1_siz = 1:BAccerm1_siz
bode(tf(BAccer_m1(1,1,iBAccerm1_siz)),fw)
hold on
end
end
%%
% m2 variations = [108.3, 118.56];
m2_vari = 924.35:vari:1011.92;
figure(16);
for im2_vari = 1:length(m2_vari)
m2_PSS025 = usubs(PSS025,'m2',im2_vari);
BAccer_m2 = usample(m2_PSS025,10);
```
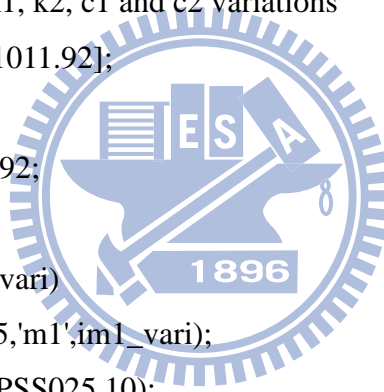
```
BAccerm2_siz = size(BAccer_m2,3);

for iBAccerm2_siz = 1:BAccerm2_siz

bode(tf(BAccer_m2(1,1,iBAccerm2_siz)),fw)

hold on

end

end

%%

% k1 variations = [40584, 44428.8];

vari_1 = 500;

k1_vari = 40584:vari_1:44428.8;

figure(17);

for ik1_vari = 1:length(k1_vari)

k1_PSS025 = usubs(PSS025,'k1',ik1_vari);

BAccer_k1 = usample(k1_PSS025,10);

BAccerk1_siz = size(BAccer_k1,3);

for iBAccerk1_siz = 1:BAccerk1_siz

bode(tf(BAccer_k1(1,1,iBAccerk1_siz)),fw)

hold on

end

end

%%

% k2 variations = [96059.25, 105159.6];

k2_vari = 96059.25:vari_1:105159.6;

figure(18);

for ik2_vari = 1:length(k2_vari)

k2_PSS025 = usubs(PSS025,'k2',ik2_vari);

BAccer_k2 = usample(k2_PSS025,10);

BAccerk2_siz = size(BAccer_k2,3);

for iBAccerk2_siz = 1:BAccerk2_siz

bode(tf(BAccer_k2(1,1,iBAccerk2_siz)),fw)

hold on

end

end

%%
```
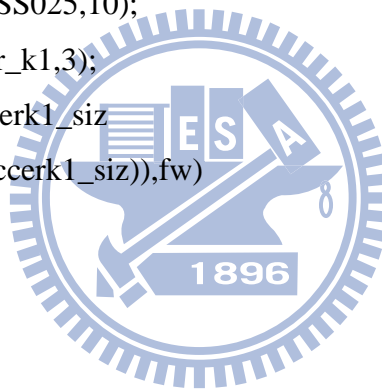
```
% c1 variations = [1040.25, 1138.8];
c1_vari = 1040.25:vari:1138.8;
figure(19);
for ic1_vari = 1:length(c1_vari)
c1_PSS025 = usubs(PSS025,'c1',ic1_vari);
BAccer_c1 = usample(c1_PSS025,10);
BAccerc1_siz = size(BAccer_c1,3);
for iBAccerc1_siz = 1:BAccerc1_siz
bode(tf(BAccer_c1(1,1,iBAccerc1_siz)),fw)
hold on
end
end
%%
% c2 variations = [13.87, 15.184];
c2_vari = 13.87:vari:15.184;
figure(20);
for ic2_vari = 1:length(c2_vari)
c2_PSS025 = usubs(PSS025,'c2',ic2_vari);
BAccer_c2 = usample(c2_PSS025,10);
BAccerc2_siz = size(BAccer_c2,3);
for iBAccerc2_siz = 1:BAccerc2_siz
bode(tf(BAccer_c2(1,1,iBAccerc2_siz)),fw)
hold on
end
end
```