Modeling, Analysis and Control of Cart-Inverted Pendulum Systems

and Fundamental Tradeoffs

by

Soham Sarkar

A Thesis Presented in Partial Fulfillment
of the Requirements for the Degree
Master of Science

Approved July 2021 by the
Graduate Supervisory Committee:

Armando Antonio Rodriguez, Co-Chair
Spring Melody Berman, Co-Chair
Hamidreza Marvi

ARIZONA STATE UNIVERSITY

August 2021

ABSTRACT

This thesis lays down a foundation for more advanced work on bipeds by carefully examining cart-inverted pendulum systems (CIPS, often used to approximate each leg of a biped) and associated closed loop performance tradeoffs. A CIPS is characterized by an instability (associated with the tendency of the pendulum to fall) and a right half plane (RHP, non-minimum phase) zero (associated with the cart displacement x). For such a system, the zero is typically close to (and smaller) than the instability. As such, a classical PK control structure would result in very poor sensitivity properties. It is therefore common to use a hierarchical inner-outer loop structure. As such, this thesis examines how such a structure can be used to improve sensitivity properties beyond a classic PK structure and systematically tradeoff sensitivity properties at the plant input/output. While the instability requires a minimum bandwidth at the plant input, the RHP zero imposes a maximum bandwidth on the cart displacement x. Three CIPs are examined – one with a long, short and an intermediately sized pendulum. We show that while the short pendulum system is the most unstable and requires the largest bandwidth at the plant input for stabilization (hardest to control), it also has the largest RHP zero. Consequently, it will permit the largest cart displacement x-bandwidth, and hence, one can argue that the short pendulum system is easiest to control. Similarly, the long pendulum system is the least unstable and requires smallest bandwidth at the plant input for stabilization (easiest to control). However, because this system also possesses the smallest RHP zero it will permit the smallest cart displacement x-bandwidth, and hence, one can argue that the long pendulum system is the hardest to control. Analogous "intermediate conclusions" can be drawn for the system with the "intermediately sized" pendulum. A set of simple academic examples (growing in plant and controller complexity) are introduced to illustrate basic tradeoffs and guide the presentation of the trade studies.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

INTRODUCTION AND OVERVIEW

1.1   Motivation: What can be modeled by a Cart Inverted Pendulum?

Balancing control problems are of interest for industrial applications such as nuclear reactors, chemical industries, stabilization of elevators(Rodriguez, 2003,Rodriguez, 2004). The space and aerospace industry also uses this problem for launching satellites, rockets, elevators in flights for attitude control (Chrif and Kadda, 2014). It is necessary for these systems to acquire some instability with respect to some crucial scenarios for more maneuverability(Turkoglu and Jafarov, 2007). Systems of interest also involve segways, unicycle, figure skating, roller blading, jet skiing as well as walkers used for locomotion(Younis and Abdelati, 2009). The robotics industry in the recent times have used these control problems to get a reasonable tradeoff between speed, maneuverability and stability with respect to different terrains(Zhang et al., 2016). Most of this balancing control problems involve usage of simplistic models such as the simple pendulum, cart inverted pendulum, see-saws, spring inverted pendulums to infer on control strategies used for having a bigger hand on instability and using it to their advantage. In the literature, the cart-inverted pendulums have been used as simplistic model to reduce model complexity(Irfan et al., 2018).

Unipeds, bipeds and quadrupeds are legged creatures able to perform locomotion with the help of appendages (Mu, 2004b). One can exemplify a pogostick as a means of uniped locomotion (Yu et al., 2011). The uniped locomotion can be explained with the model of simple inverted pendulum to understand its stability(Yu et al., 2011). Moreover, the Spring Loaded Inverted Pendulum model can also be used to

understand the hopping behavior of Unipeds and its useful take the ground interaction forces(TAMADDONI et al., 2010). Furthermore, the textbook models, namely, Spring Loaded Inverted Pendulum and the Simple Inverted Pendulum serve as a useful tool for understanding the tradeoff between stability and energy efficiency(Westervelt et al., 2018). However, for these models in literature, the slip action is considered to be negligible. The systems are analyzed before and after the impact of the foot strike(Chen and Byl, 2012). In the case of slip movement of walkers, these models have less chances of stabilizing themselves(Mu, 2004c). One of the objectives of thesis is to use these textbook models such as the cart-pole sytem to get an overview of essential inferences related to stabilizing the system when the slipping motion occurs(Fung, 2011). The cart-pole systems can approximate the behavior of the leg movement with increasing amount of friction between the ground and feet(Yi et al., 2016).

Although the aforementioned textbook models may be used as viable tool for approximating biped behavior, there is a necessity for understanding the dynamics while performing locomotion(Gu, 2009). The important element is the zero moment point used for understanding the stabilizing action performed by the feet(Mu, 2004c). The zero moment point is the point on the foot about which the net moment of all the forces are equal to be zero(Raibert, 1986). It is enclosed by an area upon which the stability of the entire system thrives(Mu, 2004b). This enclosed area is known as the support polygon(B. He et al., 2019). Thinking of the center of mass as the lumped mass of the cart-pole system and the cart as the support polygon, the behavior of walking can be approximated(Vázquez and Velasco-Villa, 2013). This will be crucial in achieving a reasonable understanding of the speed and maneuverability of the robot under certain conditions under movements involving slip action as well as for balancing control(Mu, 2004a).

2

Hence, this particular textbook model, namely, the cart-pole system is advantageous in approximation of locomotion of legged creatures (Mu, 2004b). This model has been used in the literature since the 1960s(Namasivayam, 2020). However, the most interesting aspect of this particular model involves the usage of a variety of control designs and control architectures(Gandhi et al., 2016). These architecture and designs depict the usefulness and versatility of this model(Gandhi et al., 2016). In contrast, not all of the control designs act as an ally for getting useful properties (Middleton, 1991). Certainly, some fundamental issues associated come into play that make this design difficult to introduce useful control methods(Gandhi et al., 2016,Middleton, 1991). One can address these issues to get robust designs. Philosophically speaking, all designs do have limitations, but some can be made useful by using some simplistic architectures and observing the trade-offs(Middleton, 1991). Inspite of addressing the architecture and trade-offs, fundamental performance limitations such as actuator saturation, non-linearities associated with the model does have an adverse impact on the design(Hu and Lin, 2001). These are the factors that needs to be carefully understood so that the design computed is useful for practical implementation(Rodriguez, 2004).

One can think of this system as a tool for understanding some of the important control architectures and control design methods addressed in the literature too(Middleton, 1991). Thus, taking these in mind, one can address the limitations of a design. Furthermore, to understand the design in a comprehensive way, it is necessarily useful to inspect the parameters involved and magnify the tradeoffs involved(Middleton, 1991). To understand the tradeoffs incrementally, simple academic examples growing in complexity has been used in this thesis. These examples are essentially the subsystems of the cart-pole system. The best takeaway that from this type of approach is that it is possible to exactly quantify the tradeoffs for guiding

us in our research(Gandhi et al., 2016). Hence, before one proceeds, it is necessarily useful to get acquainted with the system properties, characteristics and its behavior about its equilibrium(Persson, 2013). Furthermore, it illuminates the non-linearities associated with a model(Persson, 2013). To understand the importance of this behavior, one needs to look at the simple characteristics of this system by looking closely at its transfer function(Rodriguez, 2003).

The cart inverted problem is very special since it has two poles almost symmetrical about the imaginary axis and two zeros almost symmetrical about the imaginary axis if modelled with respect to negligible friction(Rodriguez, 2003). But the speciality of this problem lies with the fact that it has a right hand plane zero of lesser magnitude than the right hand plane pole(Rodriguez, 2003). Systems similar to this, face tremendous control issues that become increasingly hard to control(Hu and Lin, 2001). This thesis motivates these problems and addresses them incrementally for guiding us through pertaining research. The root cause of this issue lies on the reality that it limits the bandwidth of the system to a maximum(Hu and Lin, 2001). This issue gives rise to using versatile control architectures and demands aggressive and non-aggressive control action putting more burden on the actuators involved(Mondal et al., 2020). The control energy used to stabilize makes it very expensive if conventional control architectures are used(Puttannaiah et al., 2016). In this thesis the main focus is to use the textbook models such as the Cart Inverted Pendulum to have a useful understanding of its stability under various conditions(Mondal et al., 2019). Open loop trade studies and closed loop trade studies provide very useful analysis on crafting a design magnifying some observable trends. These trends give us some insight for improving the properties at the control and at the output for implementation on practical designs (Mondal et al., 2020). This thesis sheds some light on the aforementioned ideas using some simple mathematical analogies.

1.2 Literature Survey

As mentioned earlier, balancing control problems are very crucial for systems that lack mechanical support(Gandhi et al., 2016). To stabilize themselves, it needs actuators and sensors that reduce the error to a minimum(Gandhi et al., 2016). Feedback control is necessarily useful when it comes to error minimization and contributing to this issue(Gandhi et al., 2016). This type of control involves designs that may be a burden on the actuators and sensors involved with the control action(Xin, 2008). However, some design methodologies and control architectures may ease the suffering on the actuators and sensor action(Middleton, 1991). In contrast to our previous statement, there are also some drawbacks and limitations associated with these methodologies and architectures(Irfan et al., 2018). To realise the internal working of these methodologies, there is a need for studying the trends involved with the designs accounted for(Irfan et al., 2018). It gives useful inferences to prevent fundamental performance limitations and the limitations pertaining to each design(Rigatos et al., 2018).

1.2.1 $P - K$ control architecture

This type of architecture has been preliminary used in the literature to minimize the control action and error associated with a feedback control system(Rodriguez, 2003). However, these architectures become very difficult to implement under some characteristics associated with a system(Rodriguez, 2003). One of them is the instability associated with the system which restricts the ranges that can be used by a controller to stabilize the system(Rodriguez, 2004). Secondly, the right hand plane zero serves as another boundary on the gains that can be implemented by a controller(Rodriguez, 2004). Generally speaking, a $P - K$ control architecture in-

volves a controller in series with a plant accompanied by feedback to for minimizing the error, disturbance attenuation, noise reduction and reference command following(Rodriguez, 2003). This architecture will be discussed in the later chapters.

1.2.2   Inner-Outer Loop Control Architecture

In contrast to our previous architecture discussed, another type of control architecture becomes a crucial tool in addressing and avoiding the limitations associated with it(Mondal et al., 2020). This type of architecture involves an inner and an outer loop. The purpose of two loops is to distribute the control action into separate architectures and use each other when it is necessary to trade-off the control action required to get useful designs(Puttannaiah et al., 2016). The inner loop deals with the right hand plane zero by pushing it to the left hand plane(Kwakernaak, 1993). The zeros are a type of energy attractors. Hence, they attract the right hand pole into the left hand plane(Mondal et al., 2020). But the action of moving the right hand pole to the left hand plane is carried out by the outer loop(Mondal et al., 2020). As one proceeds through the chapters, the necessity of this type of architecture is illuminated(Kaya et al., 2007).

1.2.3   Cart Inverted Pendulum Modeling

To understand the cart-pole system in theory, the system needs a mathematical model(Irfan et al., 2018). The urgency of this model is to understand and compute the equations of motion(Irfan et al., 2018). The equations of motion are computed using Euler-Lagrange(Irfan et al., 2018). These equations of motion are a necessity for realizing the relations of outputs and inputs, computing transfer functions, understanding the exchange of energy involved with this system(Persson, 2013). But, most importantly, it is immensely crucial for developing control architectures to solve

its issues with balancing control associated with any complex model that requires balance(Irfan et al., 2018).

1.2.4 PID Control Design

This type of design has been initially used to address systems that involve an unstable pole(Vinodh Kumar and Jerome, 2013). The design, as inferred by its name, contributes to proportional, integral and derivative action on the error to reduce the problem of balancing control(J.-B. He et al., 2000). However, it has limitations too such as sensor noise, but most importantly, it is impossible to build a true integrator(Prasad et al., 2014). Furthermore, it is a heuristic method to evaluate the gains involved. The different types of tuning methods to get the proportional, integral and derivative gains involve the Ziegler-Nichols tuning method, Cohen coon method(Shahrokhi and Zomorrokhi, 2013). This tuning is used for initiating an exhaustive search that will be illuminated in the later chapters of this thesis(Shahrokhi and Zomorrokhi, 2013).

1.2.5 LQR Design

The linear quadratic regulator also known as LQR involves minimizing an infinity horizon control quadratic cost function.This design is very efficient in computing a useful control law for designing closed loop systems(J.-B. He et al., 2000). However, the design involves heuristic parametes to shape its cost to a minimum(Vinodh Kumar and Jerome, 2013). The solutions of this control problem involves solving the algebraic riccati equation associated with the problem The associated solution is very useful to get a useful control law(Prasad et al., 2014). The problem necessarily gives good properties at the plant input. In contrast, even though one might some nice properties at the input, one might get some bad properties at the output(Vinodh Kumar and

7

Jerome, 2013). It does not promise good properties at the output(Gattami and Rantzer, 2005). The selection of weights to reduce the quadratic cost is the main trick to this design method(J.-B. He et al., 2000).Different hierarchical structures are also incorporated using this design. However, the inner-outer loop structure with LQR PI/PID is seen to have better properties than using the conventional $P - K$ structure with this design(J.-B. He et al., 2000). In the later chapters, this problem will be addressed and by using some example related study, it might be possible to get some decent properties at the output.

1.2.6 Equilibrated Designs

Equilibrated designs involve using designs which uses an optimal control problem by implementation of weights on the norms to shape the sensitivities at the input, output and controls(Mondal et al., 2019). This is very effective in rejecting disturbances, command following with a reasonable control action. The speciality of these designs are that the sensitivity at the output and controls are the same which leads to achieving a Pareto-optimal condition(Puttannaiah et al., 2016). By this, it is meant that on fixing the bandwidth and tweaking the parameters, it might be possible to lower and one sensitivity at output and increase the sensitivity at the controls(Bertsekas, 2009). This condition is called the Pareto-optimal condition(Mondal et al., 2019). The general mixed sensitivity $\mathcal{H}^\infty$ control method gives a good range of useful equilibrated designs for getting some inferences on bandwidth and control action and get some good properties at the input and at the output. The method involves Euler parameters to convexify the problem(Boyd and Vandenberghe, 2004). The problem thus formulated, is used to get some use parameters of the system by using convex optimization(Bertsekas, 2015). Comparing these can be very useful for implementing trade-offs and is addressed on the later chapters of this thesis.

1.3 Fundamental Questions to be Addressed

- How the length of the pendulums affect the control system bandwidth? What does it say about the actuator saturation associated with change in length?

- What are the fundamental trade-offs related to the cart-inverted pendulum systems?

- When are the trends useful and exactly quantified?

- How can designs be related to each other to depict useful information?

- How can simple academic examples growing in complexity, be used to guide the research in order to illuminate critical sensitivity trade-offs?

1.4 Contributions of Research

- Thesis illuminates fundamental design trade-offs (peak sensitivity at controls and output) associated with cart-inverted pendulum (unstable, non-minimum phase) system under inner-outer loop PID control.

- This thesis motivates this incremental thinking by using simple academic examples growing in complexity, in order to illuminate critical sensitivity trade-offs This becomes immensely applicable for extensive research study on complicated systems.

- The bandwidth associated with each system governs its stability. Furthermore, it also shows fundamental performance limitations pertaining to each of the systems. Hence, to have a realistic control design, these limitations need to be addressed carefully which is motivated by this thesis.

1.5 Overview of Thesis

- Chapter 2 includes study with examples which enhance incremental thinking to approach the cart inverted pendulum system design. The system is decomposed to simpler transfer functions with increasing levels of complexity to address the hierarchy. Each of the example inherits some similarities and differences from its previous example. However, it is observed that the fundamental similarities remain the same. This is shown by a family of plots to gain insightful information about how the cart inverted pendulum or any other complicated system can be understood with a simplistic approach. The families of plots give us the trends for stability margins, the peak sensitivities, the crossover frequencies, the frequencies corresponding to the respective peak sensitivities at the controls and at the output, delay margins which is very sensitive information for getting some useful designs comprising some good properties.

- Chapter 3 includes the study of the cart-inverted pendulum system involving some trends which are inherited from the conclusions received from the previous examples. The chapter comprises of closed loop trade studies, open loop trade studies, trends that depict closed loop stability properties at the input and at the controls. Different cart inverted pendulum systems are compared to show the control complexity related to each system.

- Chapter 4 includes control designs that contributes to inferences related to each system studied. The designs are compared and related to each other via search procedures and infers some intuitive information about equilibrated sensitivity.

- Chapter 5 summaries the thesis with the results that are received. It also mentions the future scope and directions of research.

Chapter 2

BENCHMARK EXAMPLES - ILLUMINATION OF FUNDAMENTAL INNER-OUTER LOOP HIERARCHICAL DESIGN TRADEOFFS

The cart-pole system transfer function is given by:

$$P = \frac{g_{pi}(z-s)(z+s)}{s^2(s-p)(s+p)} \tag{2.0.1}$$

In this chapter, we highlight certain issues that influence the stability margins of a control loop. The issues are induced by the poles and zeros of the system. The cart pole system has a right hand plane zero (non-minimum phase) and a right hand plane pole (maximum bandwidth). This induces our system to have the following disadvantages:

1. The upward gain margin $\uparrow GM$ is restricted to be less than infinite.

2. The downward gain margin $\downarrow GM$ is restricted to be greater than zero.

To help with these issues, inner-outer loop design is used. The outer loop deals with the position and velocity of the cart. The inner loop deals with the angle and angular velocity of the pendulum. In time, it is noticed how this design is useful in shifting the right and plane zero to the left hand plane. According to the design, the plant is decomposed into the following:

1. Inner Loop Plant P_i.

2. Outer Loop Plant P_o.

11

Keeping the aforementioned disadvantages in mind, the system needs to be inspected carefully. It might be alluring to examine the system with four poles and two zeros with the above issues in mind. Hence, for better understanding how the system behaves, the system is decomposed into simple transfer functions with useful control designs.

1. Example 1:

$$Po = (z - s); Pi = \frac{1}{s - p};$$
(2.0.2)

2. Example 2:

$$Po = \frac{z - s}{s}; Pi = \frac{1}{s - p};$$
(2.0.3)

3. Example 3:

$$Po = \frac{z - s}{s^2}; Pi = \frac{1}{s - p};$$
(2.0.4)

4. Example 4:

$$Po = \frac{z - s}{s^2}; Pi = \frac{1}{s - p};$$
(2.0.5)

5. Example 5:

$$Po = \frac{z - s}{s^2}; Pi = \frac{1}{(s - p)(s + p)};$$
(2.0.6)

6. Example 6:

$$Po = \frac{z - s}{s^2}; Pi = \frac{1}{(s - p)(s + p)};$$
(2.0.7)

7. Example 7:

$$Po = \frac{(z - s)(z + s)}{s^2}; Pi = \frac{g_p i}{(s - p)(s + p)};$$
(2.0.8)

2.1 Example 1 Set Up and Assumptions

Block Diagram. The relevant block diagram can be found in Figure 2.1.1.
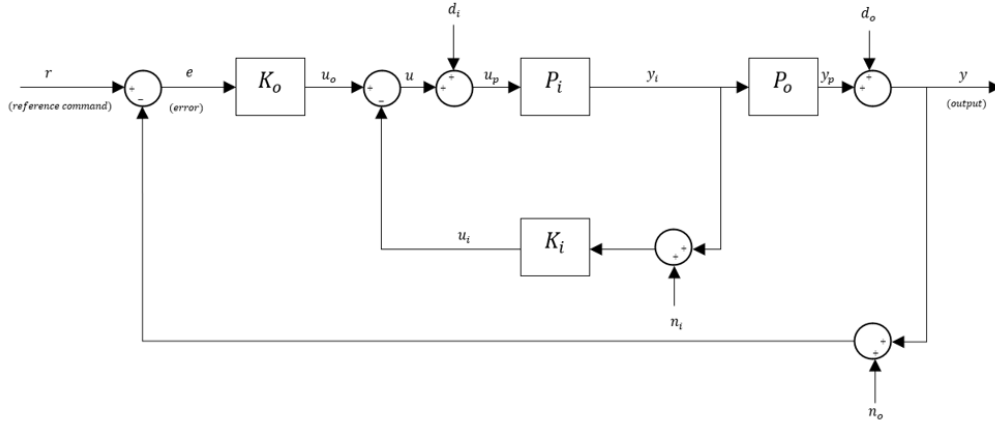
Figure 2.1.1: Block Diagram for Simple System

Nominal Parameters:

Parameter	Nominal Value	Description
p	1	Pole of P_i
z	5	Zero of P_o
g_o	(0.2-3)	Outer Loop Controller K_o Gain (will be varied)
g_i	(5-15)	Inner Loop Controller K_i Gain (will be varied)

Table 2.1.1: Nominal Parameter Values

For this simple example, the nominal parameters are chosen to depict the behavior of the system at certain conditions of the poles and zeros.The range of g_o and g_i values are chosen to show the trends that shed some light on useful corollaries that govern the inner-outer control loop design. Pertaining to the ranges, we also obtain the trends related to the open loop and once the loop is closed. At the end of this section, the range of g_o and g_i values are changed to obtain some insights on peak sensitivities.

13

Nominal System:

$$P_i = \frac{1}{s-p}, \quad K_i = g_i, \quad g_i > 0 \tag{2.1.1}$$

$$P_o = (z-s), \quad K_o = \frac{g_o}{s}, \quad g_o > 0 \tag{2.1.2}$$

Inner Loop Transfer Function:

$$L_i = P_i K_i = \frac{g_i}{s-p} \tag{2.1.3}$$

Inner Loop Sensitivity:

$$S_i = \frac{1}{1+L_i} = \frac{s-p}{s+(g_i-p)} \tag{2.1.4}$$

Inner Loop Complementary Sensitivity:

$$T_i = \frac{L_i}{1+L_i} = \frac{g_i}{s+(g_i-p)} \tag{2.1.5}$$

Loop Transfer Function at Output:

$$L_o = P_o \left(\frac{P_i}{1+P_iK_i}\right) K_o = \frac{g_o}{s} \left[\frac{z-s}{s+(p-g_i)}\right] \tag{2.1.6}$$

Loop Transfer Function at Controls:

$$L_u = K_i P_i + K_o P_o P_i = \left[\frac{1}{s-p}\right]\left[\frac{(g_i-g_o)s+g_oz}{s}\right] \tag{2.1.7}$$

Sensitivity at Output:

$$S_o = S_e = \frac{1}{1 + L_o} = \frac{s(s + g_i - p)}{s^2 + (g_i - p - g_o)s + g_o z} \qquad (2.1.8)$$

Sensitivity at Controls:

$$S_u = \frac{1}{1 + L_u} = \frac{s(s - p)}{s^2 + (g_i - p - g_o)s + g_o z} \qquad (2.1.9)$$

Inner Outer Loop Identity:

$$S_i = \frac{S_u}{S_o} \qquad (2.1.10)$$

Complementary Sensitivity at Output:

$$T_o = T_e = \frac{L_o}{1 + L_o} = \frac{g_o(s - z)}{s^2 + (g_i - p - g_o)s + g_o z} \qquad (2.1.11)$$

Complementary Sensitivity at Controls:

$$T_u = \frac{L_u}{1 + L_u} = \frac{(g_i - g_o)s + g_o z}{s^2 + (g_i - p - g_o)s + g_o z} \qquad (2.1.12)$$

Complementary Sensitivity Identities:

$$T_u - T_o = T_i S_u \qquad (2.1.13)$$

$$T_u - T_i = T_o S_i \qquad (2.1.14)$$

$$S_o = T_{re}; \qquad T_o = T_{ry}; \qquad S_u = T_{d_i u_p}; \qquad T_u = T_{d_i u} \qquad (2.1.15)$$

15

2.1.1 Fundamental Sensitivity Relationships

The closed-loop behavior does provide stability to the system but it does not tell us anything about the properties of the system. However, good closed loop characteristics of the system are obtained by close inspection of the sensitivity and complementary sensitivity of the system. Any control design involving high peak sensitivities can lead to instability, high overshoot and regretful consequences. A useful control design is determined by observing the properties of the loop at controls and the properties of the loop at output. The inner loop sensitivity S_i and inner loop complementary sensitivity T_i has a major role in influencing these properties.

But before proceeding to the major roles the inner loop is involved with, it is necessary to comprehend the dynamics of the inner loop. The inner-loop sensitivity, outer loop sensitivity, control loop sensitivity are given by 2.1.4, 2.1.8, 2.1.9.

From the aforementioned sensitivities, a critical identity is obtained. This identity assists with the trade-off between the properties of the loop at controls and the loop at output.

- **Identity:** $S_u = S_o S_i$

From this identity it is observed that:

- **lemma:** $\|S_i\|_{\mathcal{H}^\infty} \geq 1, \|S_o\|_{\mathcal{H}^\infty} \leq \|S_u\|_{\mathcal{H}^\infty}$: Increasing the peak sensitivity of the inner-loop more than one leads to better properties at controls than at output. Consequently, thus, it is obtained that:

  $\|S_i\|_{\mathcal{H}^\infty} \geq 1, \|T_u\|_{\mathcal{H}^\infty} \leq \|T_o\|_{\mathcal{H}^\infty}$ (since $S_o = 1 - T_o, S_u = 1 - T_u$).

- **lemma:** $\|S_i\|_{\mathcal{H}^\infty} < 1, \|S_u\|_{\mathcal{H}^\infty} < \|S_o\|_{\mathcal{H}^\infty}$: Increasing the peak sensitivity of the inner-loop less than one leads to better properties at output than at controls.

  $\|S_i\|_{\mathcal{H}^\infty} < 1, \|T_o\|_{\mathcal{H}^\infty} < \|T_u\|_{\mathcal{H}^\infty}$ (since $S_o = 1 - T_o, S_u = 1 - T_u$).

1. **Corollary - 1:** $p-$**Non-Aggressive Inner Loop:** $\|S_i\|_{\mathcal{H}^\infty} \geq 1 \iff g_o+p < g_i \leq 2p \iff \|S_o\|_{\mathcal{H}^\infty} \leq \|S_u\|_{\mathcal{H}^\infty}$:

   Proof: *On a closer look at the S_i, observations can be drawn that substituting $g_i = 2p, \|S_i\|_{\mathcal{H}^\infty} = 1$. Hence for values of $\|S_i\|_{\mathcal{H}^\infty} \geq 1, g_i \leq 2p$. Therefore, from the previous discussion, we obtain $\|T_o\|_{\mathcal{H}^\infty} \geq \|T_u\|_{\mathcal{H}^\infty}$ and $\|S_o\|_{\mathcal{H}^\infty} \leq \|S_u\|_{\mathcal{H}^\infty}$. Furthermore, it is noticed that lesser control action is involved in the inner loop, hence, the name non-aggressive inner loop. Since, p is involved majorly, the corollary is named as above .*

2. **Corollary - 2:** $p-$ **Aggressive Inner Loop:** $\|S_i\|_{\mathcal{H}^\infty} < 1 \iff \max\{g_o + p, 2p\} < g_i \iff \|S_o\|_{\mathcal{H}^\infty} > \|S_u\|_{\mathcal{H}^\infty}$.

   Proof: *On a closer look at the S_i, observations can be drawn that substituting $g_i = 2p, \|S_i\|_{\mathcal{H}^\infty} = 1$. Hence for values of $\|S_i\|_{\mathcal{H}^\infty} < 1, g_i < 2p$. Therefore, from the previous discussion, we obtain $\|T_o\|_{\mathcal{H}^\infty} < \|T_u\|_{\mathcal{H}^\infty}$ and $\|S_o\|_{\mathcal{H}^\infty} > \|S_u\|_{\mathcal{H}^\infty}$. Furthermore, it is noticed that more control action is involved in the inner loop, hence, the name aggressive inner loop. Since, p is involved majorly, the corollary is named as above .*

3. **Corollary - 3:** $g-$ **Aggressive Inner Loop:** $\|S_i\|_{\mathcal{H}^\infty} < 1 \iff \max\{g_o + p, 2g_o\} < g_i \iff \|T_o\|_{\mathcal{H}^\infty} < \|T_u\|_{\mathcal{H}^\infty}$.

   Proof: *On a closer look at the S_i, observations can be drawn that substituting $g_i = 2p, \|S_i\|_{\mathcal{H}^\infty} = 1$. Hence for values of $\|S_i\|_{\mathcal{H}^\infty} < 1, g_i < 2p$. Therefore, from the previous discussion, we obtain $\|T_o\|_{\mathcal{H}^\infty} < \|T_u\|_{\mathcal{H}^\infty}$ and $\|S_o\|_{\mathcal{H}^\infty} > \|S_u\|_{\mathcal{H}^\infty}$. .*

4. **Comment: Peak Sensitivities:** $\|S\|_{\mathcal{H}^\infty}, \|T\|_{\mathcal{H}^\infty}$ increase with increasing g_o and decreasing g_i.

This example is the simplest model to demonstrate the advantages of inner-outer loop design. This example is chosen as a stepping stone to the cart-pole model. One can demonstrate the similarities of the cart-pole system and this example.

Contribution of the right hand plane pole

Considering the open-loop system, the right hand plane pole contributes to the instability. It also restricts the $\downarrow GM$ to be greater than zero.

Contribution of the right hand plane zero

Considering the open-loop system, the right hand plane zero contributes to the abiding the pole to move in to the left half plane (from root-locus ideas). It also restricts the $\uparrow GM$ to be less than infinite (non-minimum phase). The requirement states that, it is necessary for a design to push the zero to the left-half plane. Furthermore, as we change the length of the pendulums for the cart system, the right hand pole shifts its position.
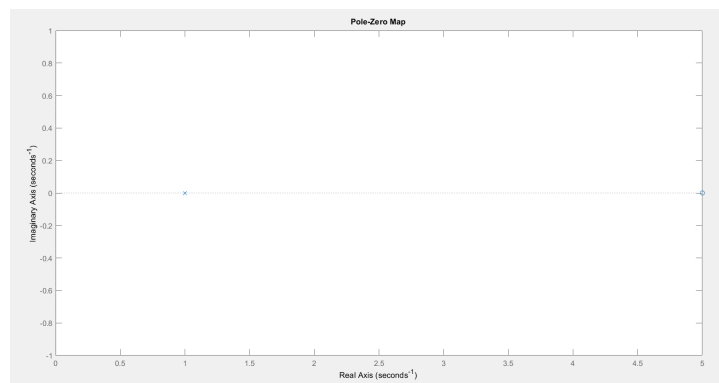
Figure 2.1.2: Nominal System: Open Loop Transfer Function, $p = 1$, $z = 5$, $g_i = 5$, $g_o = 1$

18

Considering the closed-loop system, the inner-outer loop contributes to this issue. Analytically, visualizing through root-locus ideas provides an idea about the controller to be used for the loop at output and the controls.
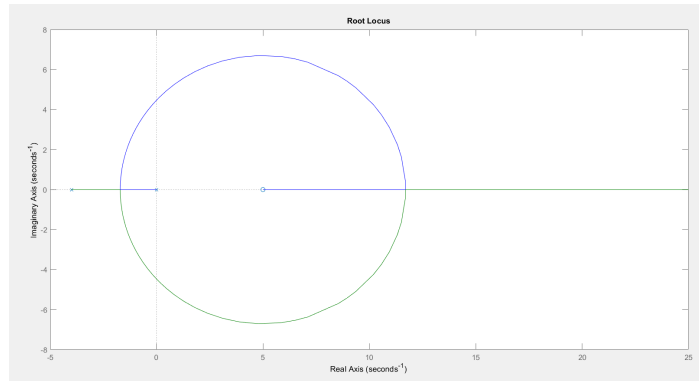
Figure 2.1.3: Nominal System: Root locus of Loop Transfer Function at Output

$$p = 1, \ z = 5, \ g_i = 5, \ g_o = 1$$

Furthermore, observations can be drawn that:

1. If $g_i \approx g_o + p$; the system has poles near the imaginary axis which leads to higher peak sensitivities t for the loop at output and for the loop at controls.
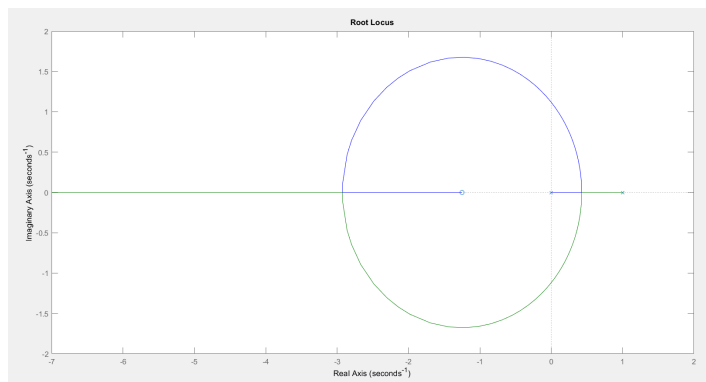
Figure 2.1.4: Nominal System: Root locus of Loop Transfer Function at Controls

$$p = 1, \ z = 5, \ g_i = 5, \ g_o = 1$$

2.1.3   Open Loop Trade Studies

Varying Outer Loop Gain g_o with g_i fixed at $g_i = 5$

Loop at Input:
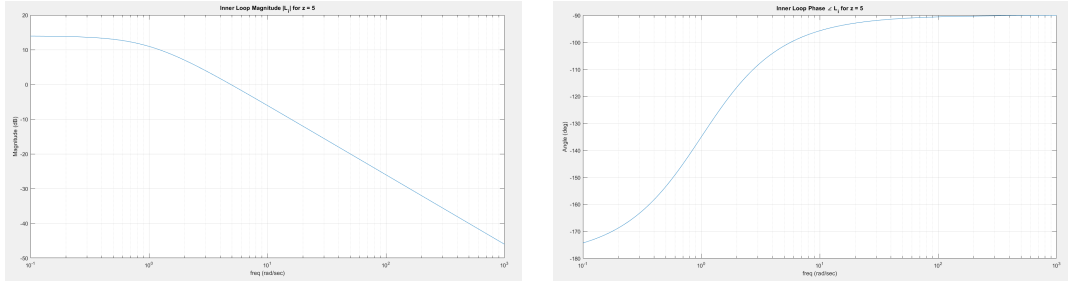
Figure 2.1.5: Inner Loop Magnitude $|L_i|$ and Phase $\angle L_i$

Loop at Output and Loop at Controls:
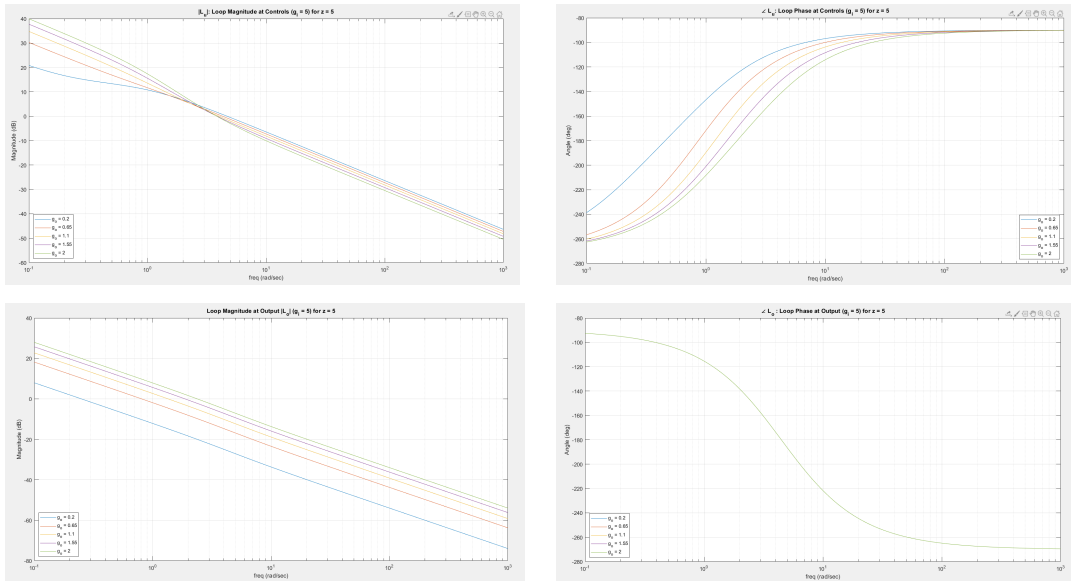
Figure 2.1.6: L_u and L_o: Loop Magnitude and Loop Phase ($g_i = 5$) for $z = 5$

- $|L_o|$, $|L_u|$ increases with increasing g_o and increases, decreases with decreasing g_i respectively. $|L_i|$, $\angle L_o$ is independent of g_o and decreases with decreasing g_i.

Varying Outer Loop Gain g_i with g_o fixed at $g_o = 2$

Loop at Input:
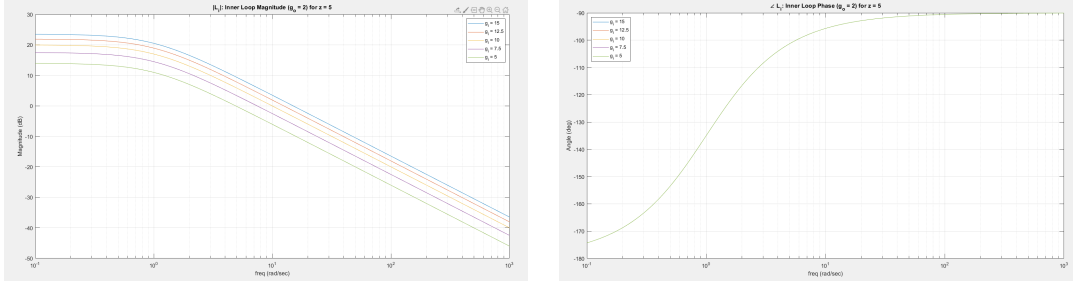
Figure 2.1.7: Inner Loop Magnitude $|L_i|$ and Phase $\angle L_i$

Loop at Output and Loop at Controls:
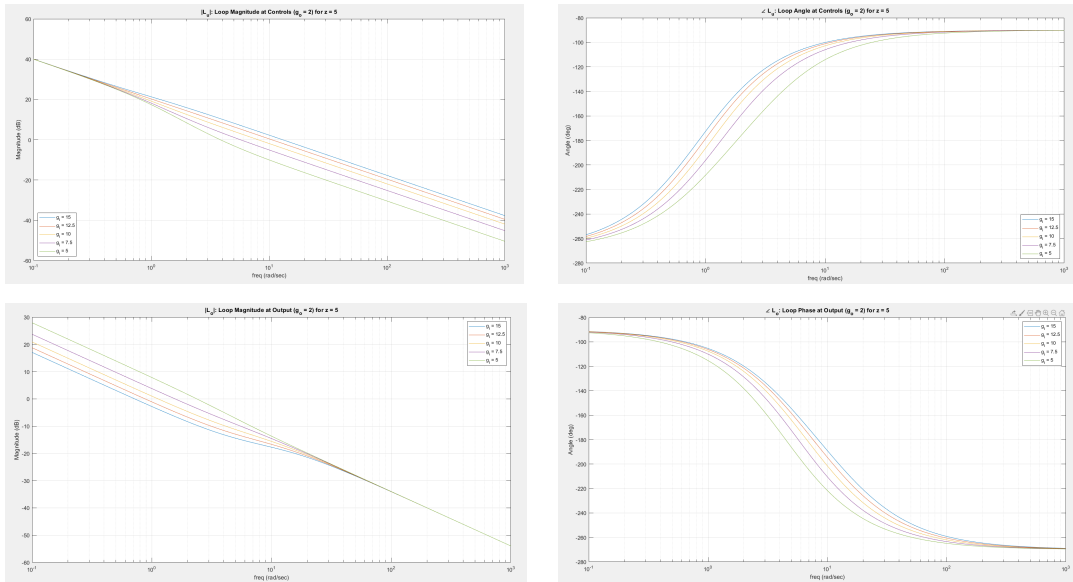
Figure 2.1.8: L_u and L_o: Loop Magnitude and Loop Phase ($g_o = 2$)

- $|L_o|$ increases with increasing g_o and increases with decreasing g_i. $|L_i|$, $\angle L_o$ is independent of g_o and decreases wtih decreasing g_i. $|L_u|$ increases with increasing g_o and decreases with decreasing g_i. $\angle L_u$ decreases with increasing g_o and decreases wtih decreasing g_i.

21

Varying Outer Loop Gain g_o with Inner Loop Gain g_i Fixed at $g_i = 5$

Inner Loop Sensitivity and Complementary Sensitivity
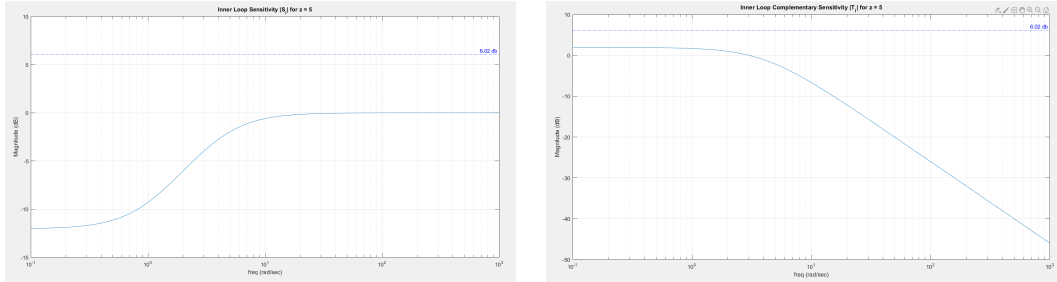
Figure 2.1.9: Inner Loop Sensitivity and Complementary Sensitivity $|S_i|$ and $|T_i|$

Loop Sensitivity at Output and at Controls
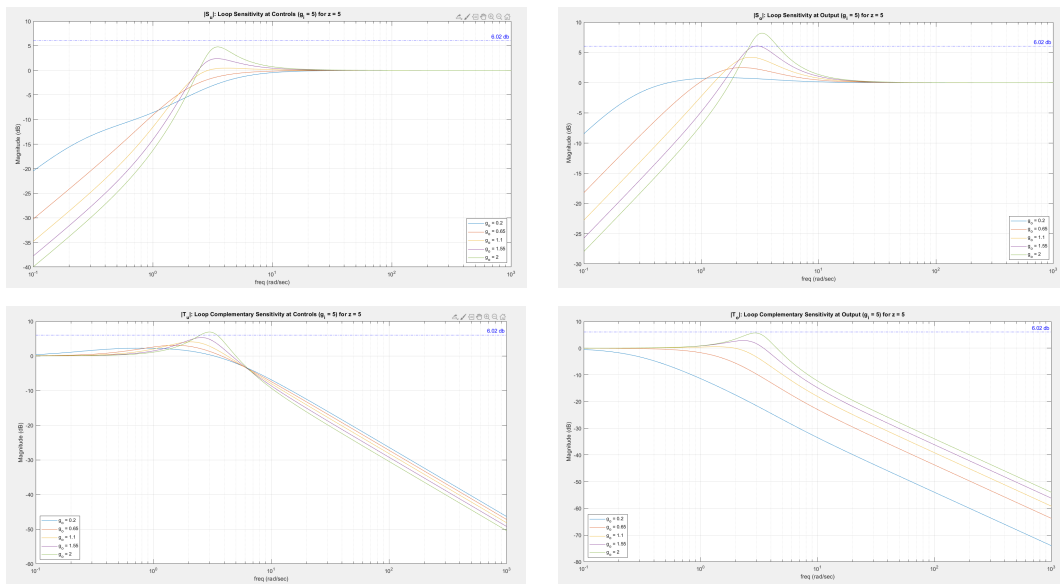
Figure 2.1.10: $|S_o|$ and $|S_u|$: Loop Sensitivity at Output and at Controls ($g_i = 5$)

- $|S_i|$ is independent of g_o and increases with decreasing g_i. It also is independent of z value. $|T_i|$ is independent of g_o, z and decreases with decreasing g_i.

Inner Outer Loop Complementary Sensitivity Ratio

$$\frac{T_u}{T_o} = \frac{\frac{(g_i-g_o)s+g_oz}{s^2+(g_i-p-g_o)s+g_oz}}{\frac{g_o(s-z)}{s^2+(g_i-p-g_o)s+g_oz}} = \frac{(g_i-g_o)s+g_oz}{g_o(z-s)} = \frac{g_is+g_o(z-s)}{g_o(z-s)} \tag{2.1.16}$$

$$= 1 + \frac{g_is}{g_o(z-s)} \tag{2.1.17}$$

$$= 1 + \left(\frac{S_o}{T_o}\right)T_i \tag{2.1.18}$$

$|\frac{T_u}{T_o}|$ **vs** g_i **(g_o - parameter)**
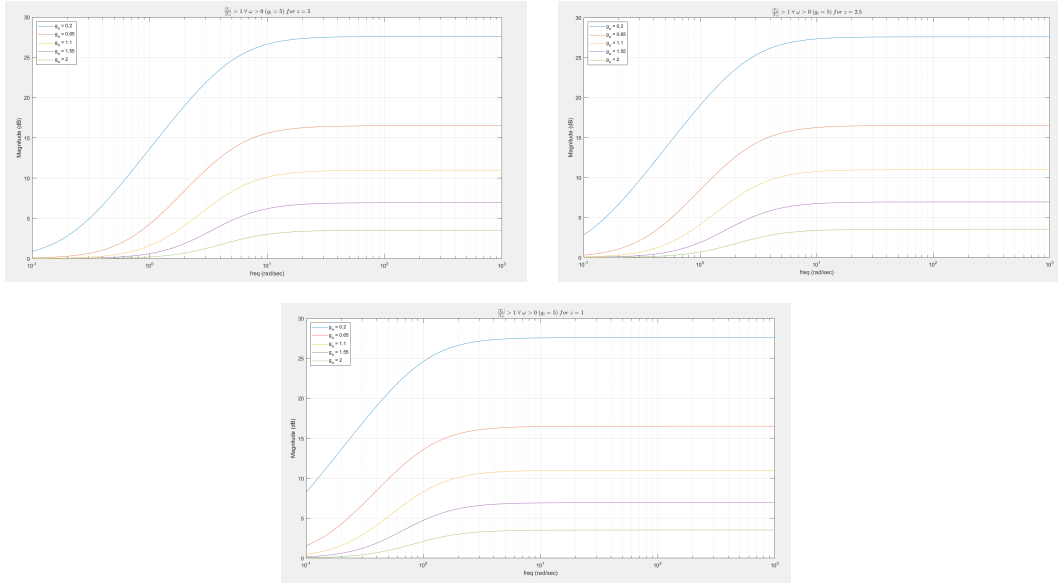
Figure 2.1.11: $|\frac{T_u}{T_o}| > 1 \;\forall\; \omega \; > 0 \; (g_i = 5)$

$$T_u - T_o = T_iS_u \tag{2.1.19}$$

$$T_u - T_i = T_oS_i \tag{2.1.20}$$

- g – Aggressive Inner Loop: $\max\{g_o+p, 2g_o\} \leq g_i \rightarrow |T_u| \geq |T_o|$, g – Aggressive Inner Loop: $\max\{g_o+p, 2g_o\} < g_i \rightarrow |T_u| > |T_o|$ ($\forall\; \omega > 0$), $\left|\frac{T_u}{T_o}\right|_{s=j\infty} = \frac{g_i}{g_o} - 1$ (> 0 since $g_i \geq g_o + p > g_o$), $\left|\frac{T_u}{T_o}\right|$ ($\omega \geq 0$) increases with decreasing g_o.

23

Peak Sensitivity and Complimentary Sensitivity
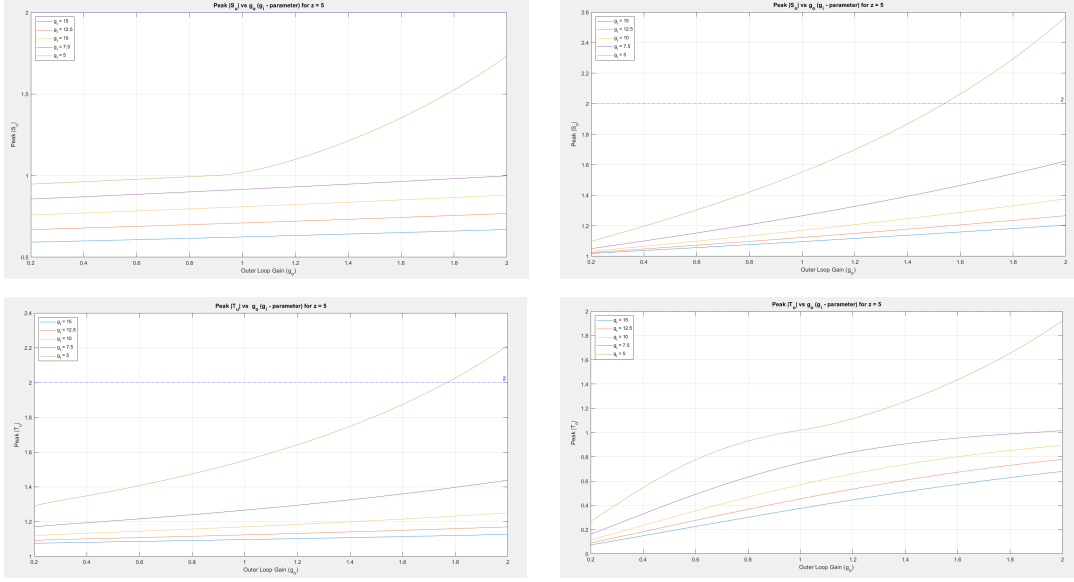
Figure 2.1.12: Peak $|S_u|$, $|S_o|$, $|T_u|$, $|T_o|$ vs g_o (g_i-parameter)

- Peak $|S_o|$ increases with increasing g_o or decreasing g_i, peak $|S_u|$ increases with increasing g_o or decreasing g_i, peak $|T_o|$ increases with increasing g_o or decreasing g_i, peak $|T_u|$ increases with increasing g_o or decreasing g_i. When g_i gets closer to g_i, the inner loop becomes aggressive and also pushes the system to have higher sensitivities. The peaks can be derived by substituting $(j\omega)$ on the denominator of the closed loop transfer function and finding its magnitude. The peaks are denoted as α. The frequencies corresponding to them are denoted as ω_{peak}. Hence we have the following peaks: α_{Su}, α_{So}, α_{Tu}, α_{To}.

1. $\alpha_{Su} = \dfrac{\omega\sqrt{\omega^2+p^2}}{\sqrt{\omega^4+[(g_i-p-g_o)^2-2g_oz]\omega^2+2g_o^2z^2}}$

2. $\alpha_{So} = \dfrac{\omega\sqrt{\omega^2+(g_i-p)^2}}{\sqrt{\omega^4+[(g_i-p-g_o)^2-2g_oz]\omega^2+2g_o^2z^2}}$

3. $\alpha_{Tu} = \dfrac{g_o^2z^2+(g_i-g_o)^2\omega^2}{\sqrt{\omega^4+[(g_i-p-g_o)^2-2g_oz]\omega^2+2g_o^2z^2)}}$

4. $\alpha_{To} = \dfrac{g_o\sqrt{z^2+\omega^2}}{\sqrt{\omega^4+[(g_i-p-g_o)^2-2g_oz]\omega^2+2g_o^2z^2}}$

24

Varying Outer Loop Gain g_i with g_o fixed at $g_o = 2$

Inner Loop Sensitivity and Complementary Sensitivity
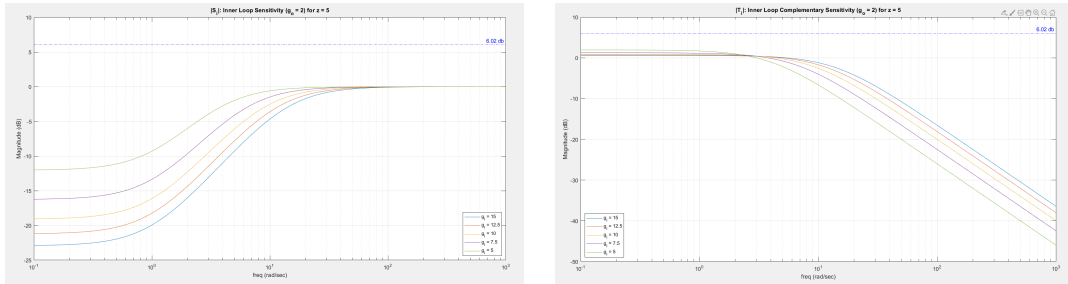
Figure 2.1.13: $|S_i|$ and $|T_i|$: Inner Loop Sensitivity and Complementary Sensitivity

$$(g_o = 2)$$

Loop Sensitivity and Complimentary sensitivity at Output and at Controls
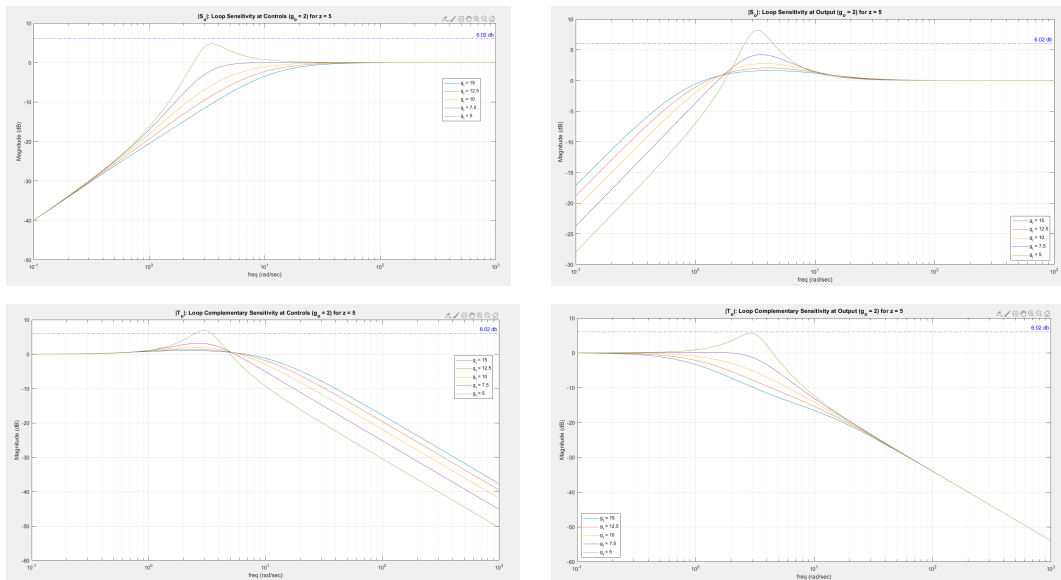
Figure 2.1.14: $|S_u|$, $|S_o|$, $|T_u|$, $|T_o|$: Loop Sensitivity at Output and at Controls

$$(g_o = 2)$$

- Peak $|S_o|$,$|S_u|$, peak $|T_o|$, peak $|T_u|$ increases with increasing g_o or decreasing g_i.

$|\frac{T_u}{T_o}|$ decreases with decreasing g_o. $|\frac{T_u}{T_o}|$ decreases with decreasing g_i. It is alose independent of z. In this example 1, this identity is always greater than 0 for $\omega > 0$. This identity varies with respect to different examples. Choosing appropriate parameters for design, the identity can be shown to be identical to example 1.

$$T_u - T_o = T_i S_u \qquad (2.1.21)$$

$$T_u - T_i = T_o S_i \qquad (2.1.22)$$

$|\frac{T_u}{T_o}|$ **vs** g_i (g_o **- parameter)**
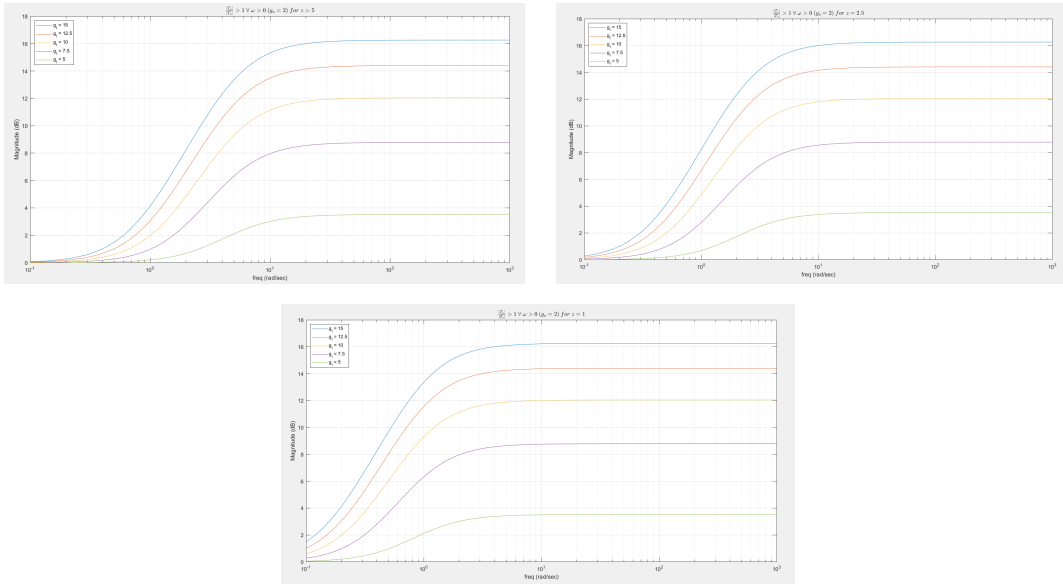
Figure 2.1.15: $|\frac{T_u}{T_o}| > 1 \ \forall \ \omega \ > 0 \ (g_o = 2)$

- $|\frac{T_u}{T_o}|$ decreases with decreasing g_o.

- $|\frac{T_u}{T_o}|$ decreases with decreasing g_i.

- $|\frac{T_u}{T_o}|$ is alo independent of z

- $\left| \frac{T_u}{T_o} \right| \ (\omega > 0)$

26
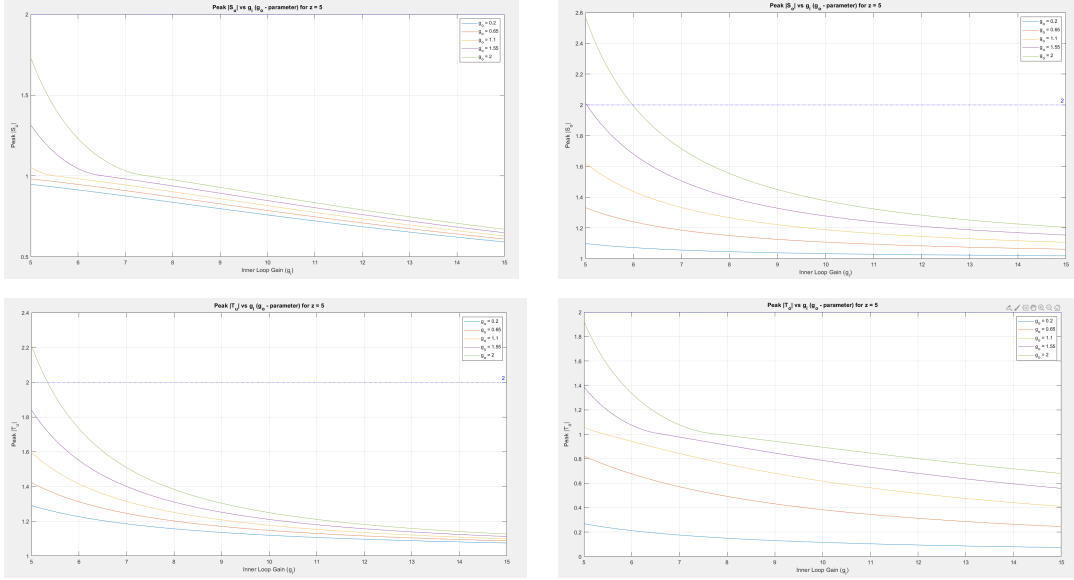
Peak $|S_o|$, $|S_u|$, $|T_o|$, $|T_u|$

Figure 2.1.16: Peak $|S_o|$, $|S_u|$, $|T_o|$, $|T_u|$ vs g_o (g_i-parameter)

- The peaks can be derived by substituting $(j\omega)$ on the denominator of the closed loop transfer function and finding its magnitude. The peaks are denoted as α. The frequencies corresponding to them are denoted as ω_{peak}. Hence we have the following α equations are derived from which one can get the following equations:

  1. $\alpha_{Su} : (\alpha^2 - 1)\omega^4 + [\alpha^2(g_i - p - g_o)^2 - 2\alpha^2 g_o z - p^2]\omega^2 + 2(g_o)^2 z^2 = 0$

  2. $\alpha_{So} : (\alpha^2 - 1)\omega^4 + [\alpha^2(g_i - p - g_o)^2 - 2\alpha^2 g_o z - (g_i - p)^2]\omega^2 + \alpha^2(g_o)^2 z^2 = 0$

  3. $\alpha_{Tu} : (\alpha^2)\omega^4 + [\alpha^2(g_i - p - g_o)^2 - 2*\alpha^2 g_o z - (g_i - g_o)^2]\omega^2 + (\alpha^2 - 1)(g_o)^2 z^2 = 0$

  4. $\alpha_{To} : (\alpha^2)\omega^4 + [\alpha^2(g_i - p - g_o)^2 - 2*\alpha^2 g_o z - (g_o)^2]\omega^2 + (\alpha^2 - 1)(g_o)^2 z^2 = 0$

These α equations are used for drawing the contour peaks with respect to g_o and g_i. The peaks corresponding to the contours are chosen to be 2, 2.2, 2.5 and 3. The values of g_o and g_i can be chosen selecting the peaks one wants to use for their design.

27

$|S_u|$, $|S_o|$ aggressive and non-aggressive behavior

$|S_u|$ aggressive behavior
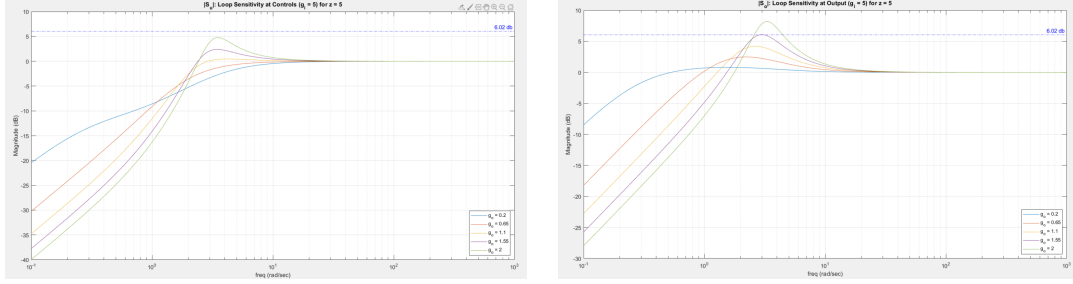
Figure 2.1.17: $|S_u|$ aggressive behavior

$|S_u|$ non aggressive behavior
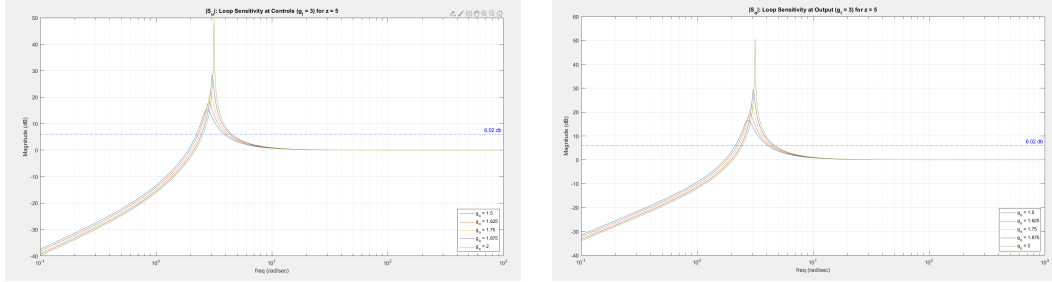
Figure 2.1.18: $|S_u|$ non aggressive behavior

- The frequencies corresponding to them are denoted as ω_{peak}. The ω equations can be derived by substituting $j\omega$ in the denominator of the closed loop transfer functions and computing with respect to ω.

1. $\omega_{Su} = \dfrac{(g_o^2 z^2) + g_o z \sqrt{(g_i p - g_o p + g_o z)(g_o p - g_i p + g_o z + 2p^2)}}{(g_i - g_o)(2p - g_i - g_o)}$

2. $\omega_{So} = \dfrac{(g_o z^2) + z \sqrt{g_o^2 z^2 + (2g_i - 2p - g_o + 2z)(g_i - p)^2 g_o}}{2g_i - 2p - g_o + 2z}$

3. $\omega_{Tu} = \dfrac{-(g_o^2 z^2) - g_o z \sqrt{(g_i p - g_o p + g_o z)(g_o p - g_i p + g_o z + 2(g_i - g_o)^2)}}{(g_i - g_o)^2}$

4. $\omega_{To} = z \sqrt{(g_i - p + z)(2g_o - g_i + p + z)}$

28

Varying Outer Loop Gain g_o with Inner Loop Gain g_i Fixed at $g_i = 5$

Time Domain Analysis: Impulse S_o, S_u, T_o, T_u
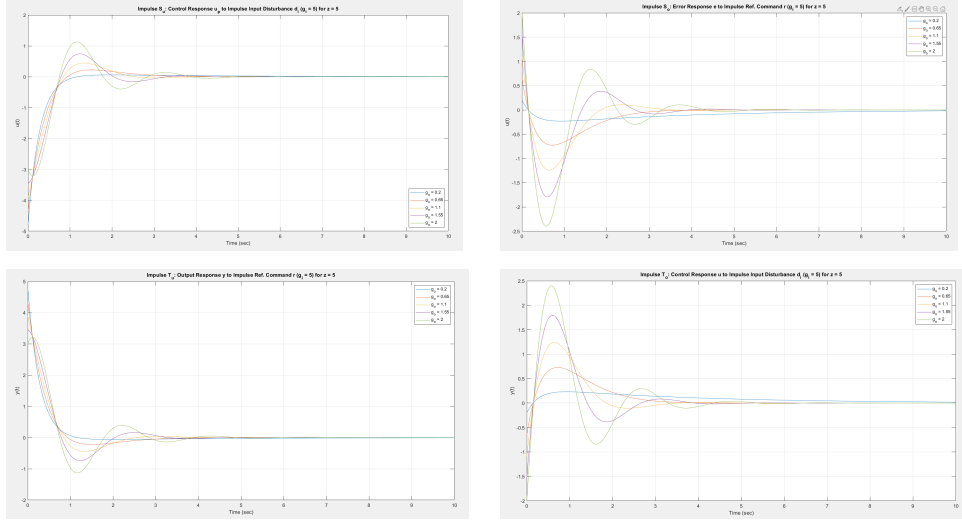
Figure 2.1.19: Impulse S_o, S_u, T_o, T_u $(g_i = 5)$

Time Domain Analysis: Step S_o, S_u, T_o, T_u
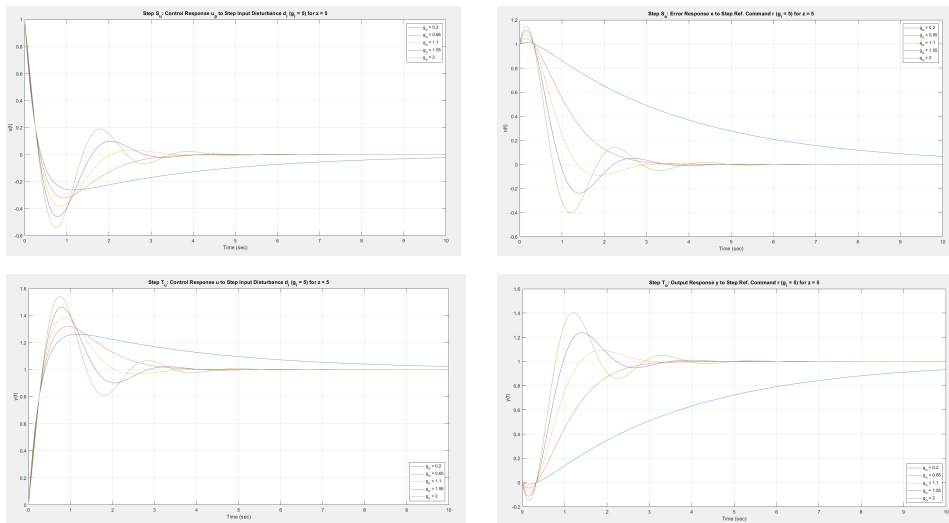
Figure 2.1.20: Step S_o, S_u, T_o, T_u $(g_i = 5)$

29

Varying Outer Loop Gain g_o with Inner Loop Gain g_i Fixed at $g_o = 2$

Time Domain Analysis: Impulse S_o, S_u, T_o, T_u
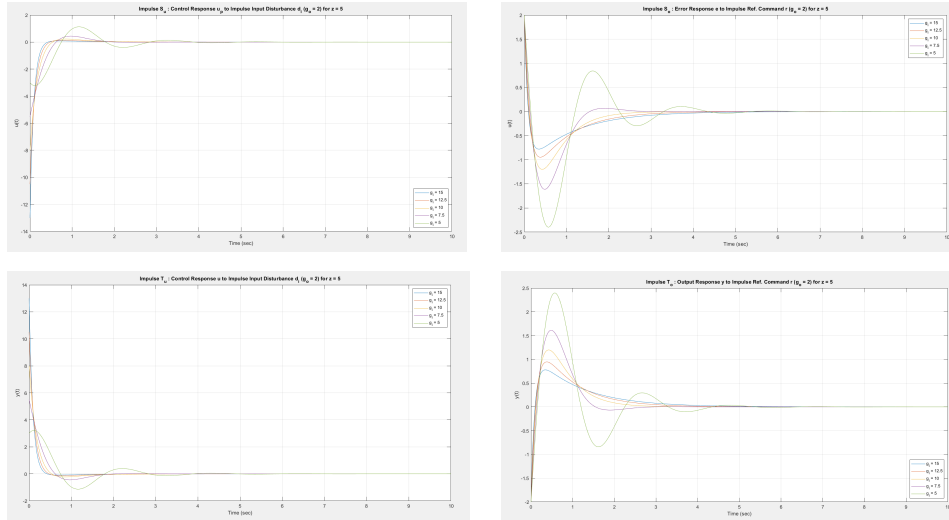
Figure 2.1.21: Impulse S_o, S_u, T_o, T_u ($g_o = 2$)

Time Domain Analysis: Step S_o, S_u, T_o, T_u
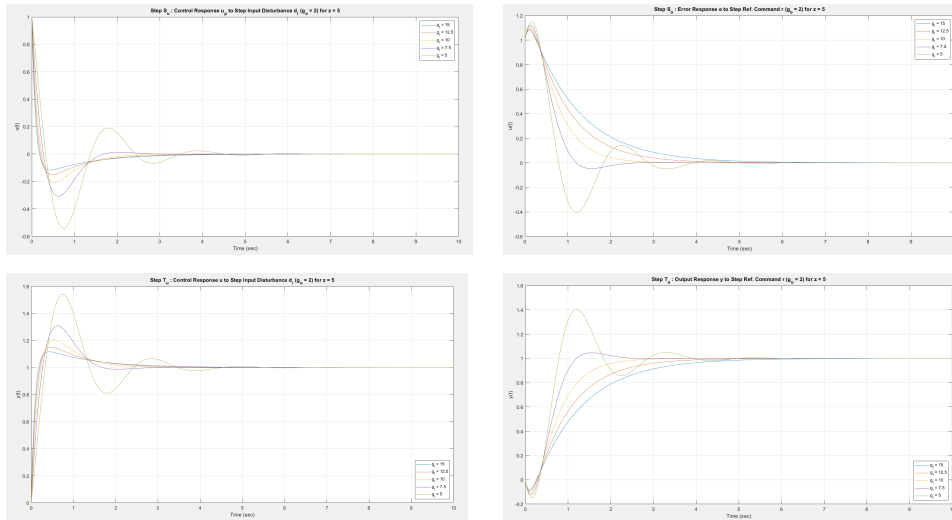
Figure 2.1.22: Step S_o, S_u, T_o, T_u ($g_o = 2$)

- S_u, S_o, T_u, T_o oscillations increase with increasing g_o or decreasing g_i.

2.1.6 Summary and Conclusions

	$g_o \uparrow$	$g_i \downarrow$	$z \downarrow$
$|L_u|$	$\uparrow$	$\downarrow$	$\downarrow$
$\angle L_u$	$\downarrow$	$\downarrow$	$\uparrow$
$|L_o|$	$\uparrow$	$\uparrow$	$\downarrow$
$\angle L_o$	$-$	$\downarrow$	$\downarrow$
$\downarrow GM$	$\uparrow$	$\uparrow$	$-$
$\omega_{\downarrow GM}$	$\uparrow$	$\uparrow$	$\downarrow$
$\uparrow GM$	$\downarrow$	$\downarrow$	$-$
$\omega_{\uparrow GM}$	$-$	$\downarrow$	$\downarrow$
PM_u	$\downarrow$	$\downarrow$	$\uparrow$
PM_o	$\downarrow$	$\downarrow$	$\downarrow$
DM_u	$\uparrow$	$\uparrow$	$\uparrow$
DM_o	$\downarrow$	$\downarrow$	$\uparrow$
ω_{gu}	$\downarrow$	$\downarrow$	$\uparrow$
ω_{go}	$\uparrow$	$\uparrow$	$\downarrow$
$PeakS_u$	$\uparrow$	$\uparrow$	$\downarrow$
$PeakS_o$	$\uparrow$	$\uparrow$	$\downarrow$
$PeakT_u$	$\uparrow$	$\uparrow$	$\downarrow$
$PeakT_o$	$\uparrow$	$\uparrow$	$\downarrow$
$|S_i|$	$-$	$\uparrow$	$-$
$|T_i|$	$-$	$\downarrow$	$-$

Table 2.1.2: Summary: Example 1

2.2   Example 2 Set Up and Assumptions

Block Diagram. The relevant block diagram can be found in Figure 2.2.1.
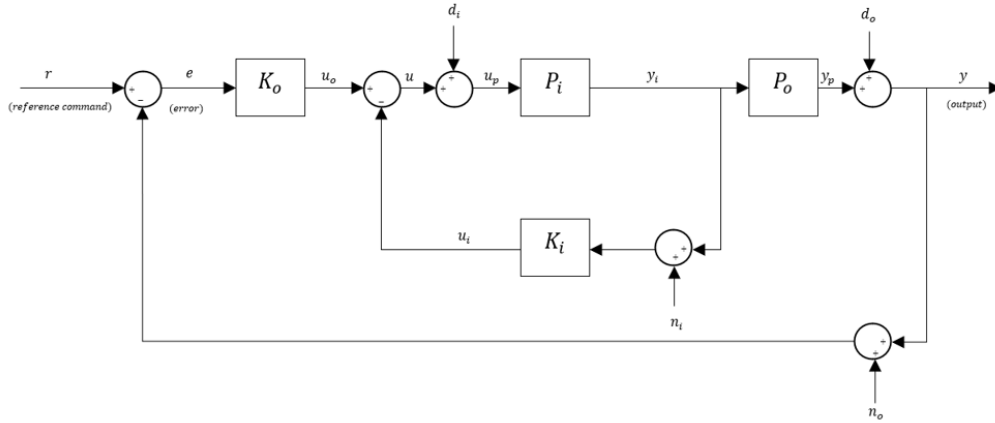
Figure 2.2.1: Block Diagram for Example 2

Nominal Parameters:

Parameter	Nominal Value	Description
p	1	Pole of P_i
z	5	Zero of P_o
g_o	(0.8-2)	Outer Loop Controller K_o Gain (will be varied)
g_i	(5-15)	Inner Loop Controller K_i Gain (will be varied)
z_o	(0.1)	Outer Loop Controller K_o Zero

Table 2.2.1: Nominal Parameter Values

This example is a tad complicated compared to example 1. The analogy developed in this example assists with the examples as we progress through each of them. In agreement with the parameters mentioned above the following trade studies are very insightful with respect to the cart inverted pendulum ideas.

Nominal System:

$$P_i = \frac{1}{s-p}, \quad K_i = g_i, \quad g_i > 0, \quad g_o > 0, \quad z_o > 0 \tag{2.2.1}$$

$$P_o = \frac{z-s}{s}, \quad K_o = \frac{g_o(s+z_o)}{s}, \quad g_i > 0, \quad g_o > 0, \quad z_o > 0 \tag{2.2.2}$$

Inner Loop Transfer Function:

$$L_i = P_i K_i = \frac{g_i}{s-p} \tag{2.2.3}$$

Inner Loop Sensitivity:

$$S_i = \frac{1}{1+L_i} = \frac{s-p}{s+(g_i-p)} \tag{2.2.4}$$

Inner Loop Complementary Sensitivity:

$$T_i = \frac{L_i}{1+L_i} = \frac{g_i}{s+(g_i-p)} \tag{2.2.5}$$

Loop Transfer Function at Output:

$$L_o = P_o \left(\frac{P_i}{1+P_iK_i} \right) K_o = \frac{g_o}{s^2} \left[\frac{(z-s)(s+z_o)}{(s+(g_i-p))} \right] \tag{2.2.6}$$

Loop Transfer Function at Controls:

$$L_u = K_iP_i + K_oP_oP_i = \frac{(g_i-g_o)s^2 + (g_oz - g_oz_o)s + (g_ozz_o)}{s^2(s-p)} \tag{2.2.7}$$

Sensitivity at Output:

33

$$S_o = S_e = \frac{1}{1 + L_o} = \frac{s^2(s + (g_i - p))}{s^3 + (g_i - p - g_o)s^2 + (g_o z - g_o z_o)s + g_o z z_o} \qquad (2.2.8)$$

Sensitivity at Controls:

$$S_u = \frac{1}{1 + L_u} = \frac{s^2(s - p))}{s^3 + (g_i - p - g_o)s^2 + (g_o z - g_o z_o)s + g_o z z_o)} \qquad (2.2.9)$$

Inner Outer Loop Identity:

$$S_i = \frac{S_u}{S_o} \qquad (2.2.10)$$

Complementary Sensitivity at Output:

$$T_o = T_e = \frac{L_o}{1 + L_o} = \frac{g_o(z - s)(s + z_o)}{s^3 + (g_i - p - g_o)s^2 + (g_o z - g_o z_o)s + g_o z z_o} \qquad (2.2.11)$$

Complementary Sensitivity at Controls:

$$T_u = \frac{L_u}{1 + L_u} = \frac{(g_i - g_o)s^2 + (g_o z - g_o z_o)s + (g_o z z_o)}{s^3 + (g_i - p - g_o)s^2 + (g_o z - g_o z_o)s + g_o z z_o} \qquad (2.2.12)$$

Complementary Sensitivity Identities:

$$T_u - T_o = T_i S_u \qquad (2.2.13)$$

$$T_u - T_i = T_o S_i \qquad (2.2.14)$$

$$S_o = T_{re}; \qquad T_o = T_{ry}; \qquad S_u = T_{d_i u_p}; \qquad T_u = T_{d_i u} \qquad (2.2.15)$$

2.2.1 Fundamental Sensitivity Relationships

Similar to example 1. The tradeoff between the properties of the loop at controls and the loop at output is given by the inner-outer loop identity:

- **Identity:** $S_u = S_o S_i$

From this identity it is observed that:

- **lemma:** $\|S_i\|_{\mathcal{H}^\infty} \geq 1, \|S_o\|_{\mathcal{H}^\infty} \leq \|S_u\|_{\mathcal{H}^\infty} : \|S_i\|_{\mathcal{H}^\infty} \geq 1, \|T_u\|_{\mathcal{H}^\infty} \leq \|T_o\|_{\mathcal{H}^\infty}$
  (since $S_o = 1 - T_o, S_u = 1 - T_u$).

- **lemma:** $\|S_i\|_{\mathcal{H}^\infty} < 1, \|S_u\|_{\mathcal{H}^\infty} < \|S_o\|_{\mathcal{H}^\infty} : \|S_i\|_{\mathcal{H}^\infty} < 1, \|T_o\|_{\mathcal{H}^\infty} < \|T_u\|_{\mathcal{H}^\infty}$
  (since $S_o = 1 - T_o, S_u = 1 - T_u$).

Hence, the inner-loop works as a major tool to play between the peak sensitvities at output and controls.

From the above lemmas, and close inspection of the system following corollaries are obtained:

1. **Corollary 1:** $p - $ **Non-Aggressive Inner Loop:** $\|S_i\|_{\mathcal{H}^\infty} \geq 1 \iff g_o + p + \frac{z z_o}{z - z_o} < g_i \leq 2p \iff \|S_o\|_{\mathcal{H}^\infty} \leq \|S_u\|_{\mathcal{H}^\infty}:$

2. **Corollary 2:** $p - $ **Aggressive Inner Loop:** $\|S_i\|_{\mathcal{H}^\infty} < 1 \iff \max\{g_o + p + \frac{z z_o}{z - z_o}, 2p\} < g_i \iff \|S_o\|_{\mathcal{H}^\infty} > \|S_u\|_{\mathcal{H}^\infty}.$

3. **Corollary 3:** $g - $ **Aggressive Inner Loop:** $\|S_i\|_{\mathcal{H}^\infty} < 1 \iff \max\{g_o + p + \frac{z z_o}{z - z_o}, 2g_o\} < g_i \iff \|T_o\|_{\mathcal{H}^\infty} < \|T_u\|_{\mathcal{H}^\infty}.$

4. **Comment: Peak Sensitivities:** $\|S\|_{\mathcal{H}^\infty}, \|T\|_{\mathcal{H}^\infty}$ increase with increasing g_o and decreasing g_i.

Similar to example 1, we get the stability of the inner loop as:
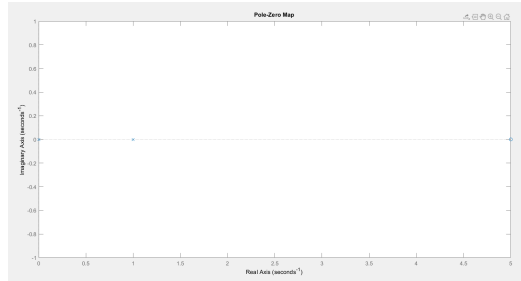
Figure 2.2.2: Nominal System: Open Loop Transfer Function

1. If $g_i \approx g_o + p + \frac{zz_o}{z-z_o}$; the system has poles near the imaginary axis which leads to higher peak sensitivities t for the loop at output and for the loop at controls.
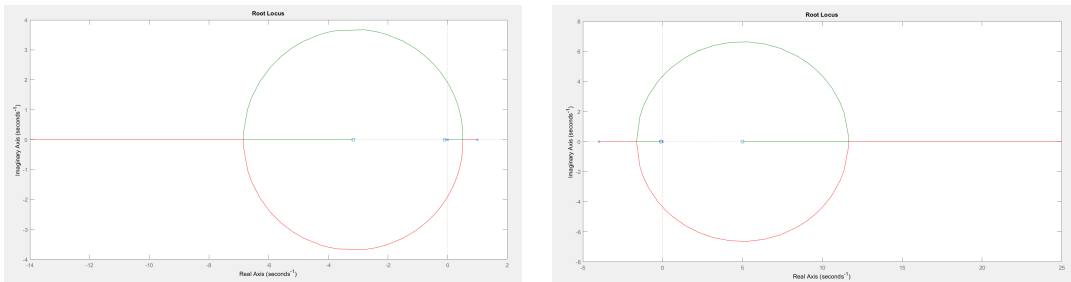
Figure 2.2.3: Nominal System: Root locus of Loop Transfer Function at Controls (left) and at Output (right)

The above plots explain that the $\uparrow GM$ at controls is infinite, $\downarrow GM$ at controls is finite. On the other hand, $\downarrow GM$ at output is zero, $\uparrow GM$ is finite because of the right hand plane zero. Thus, this cause the system to be bandwidth limited.

36

Varying Outer Loop Gain g_o **with** g_i **fixed at** $g_i = 5$

Loop at Input:
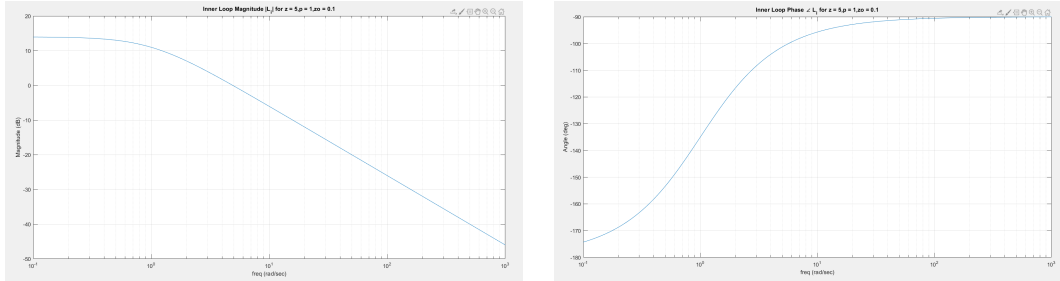
Figure 2.2.4: Inner Loop Magnitude $|L_i|$ and Phase $\angle L_i$

Loop at Output and Loop at Controls:
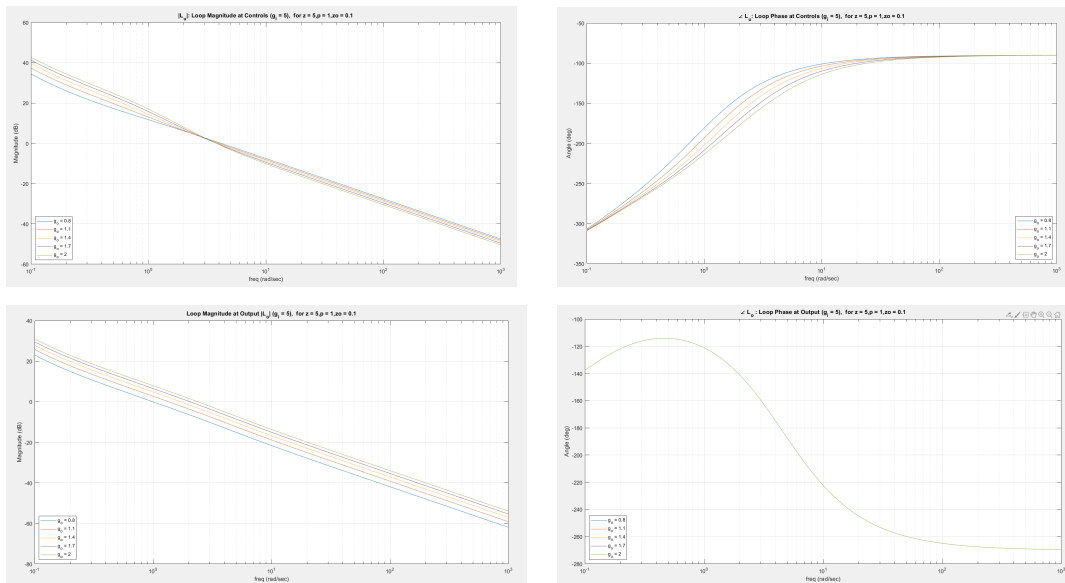
Figure 2.2.5: L_u and L_o: Loop Magnitude and Loop Phase ($g_i = 5$) for $z = 5$

- $|L_u|$, $|L_o|$ increases with increasing g_o and increases, decreases with decreasing g_i respectively. $\angle L_o$ is independent of g_o and decreases wtih decreasing g_i. $\angle L_u$ decreases with increasing g_o and decreases wtih decreasing g_i.

Varying Outer Loop Gain g_i with g_o fixed at $g_o = 2$

Loop at Input:
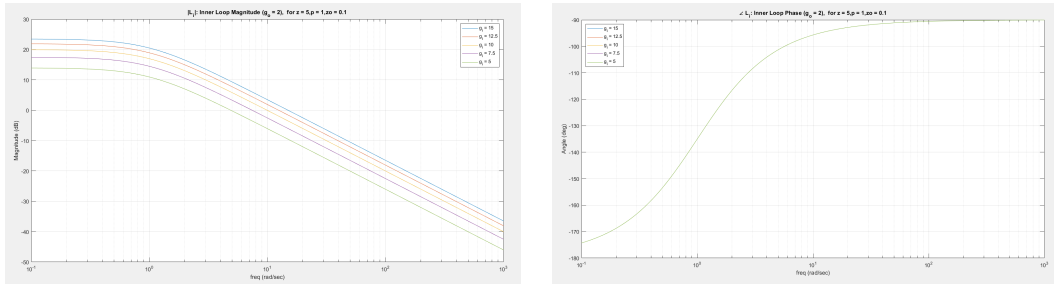
Figure 2.2.6: Inner Loop Magnitude $|L_i|$ and Phase $\angle L_i$

Loop at Output and Loop at Controls:
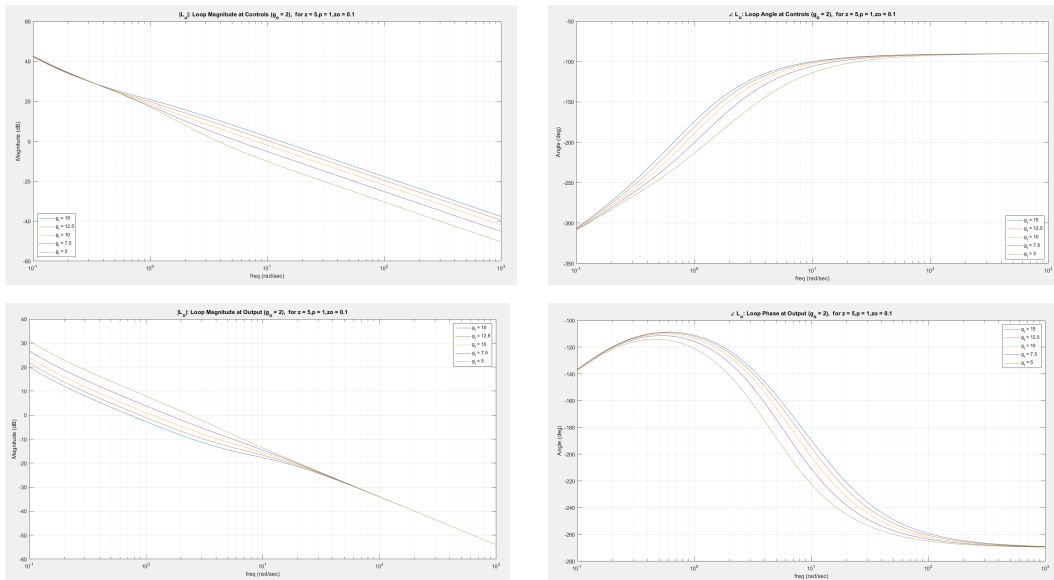
Figure 2.2.7: L_u and L_o: Loop Magnitude and Loop Phase ($g_o = 2$)

- $|L_o|$ increases with increasing g_o and increases with decreasing g_i. $\angle L_o$ is independent of g_o and decreases wtih decreasing g_i.

- $|L_u|$ increases with increasing g_o and decreases with decreasing g_i. $\angle L_u$ decreases with increasing g_o and decreases wtih decreasing g_i.

38

Varying Outer Loop Gain g_o with Inner Loop Gain g_i Fixed at $g_i = 5$

Inner Loop Sensitivity and Complementary Sensitivity
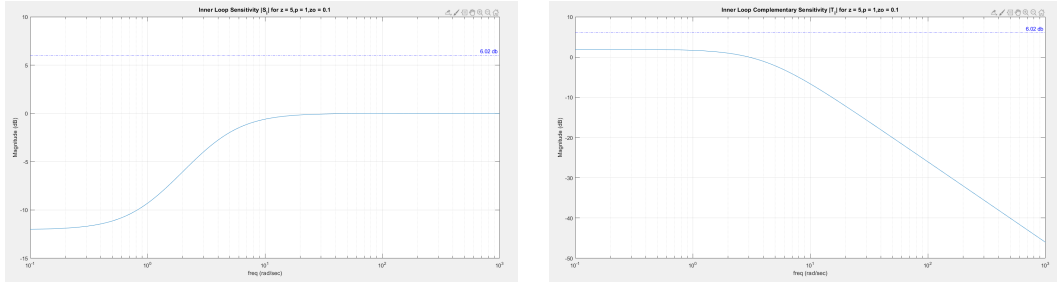
Figure 2.2.8: Inner Loop Sensitivity and Complementary Sensitivity $|S_i|$ and $|T_i|$

Loop Sensitivity at Output and at Controls
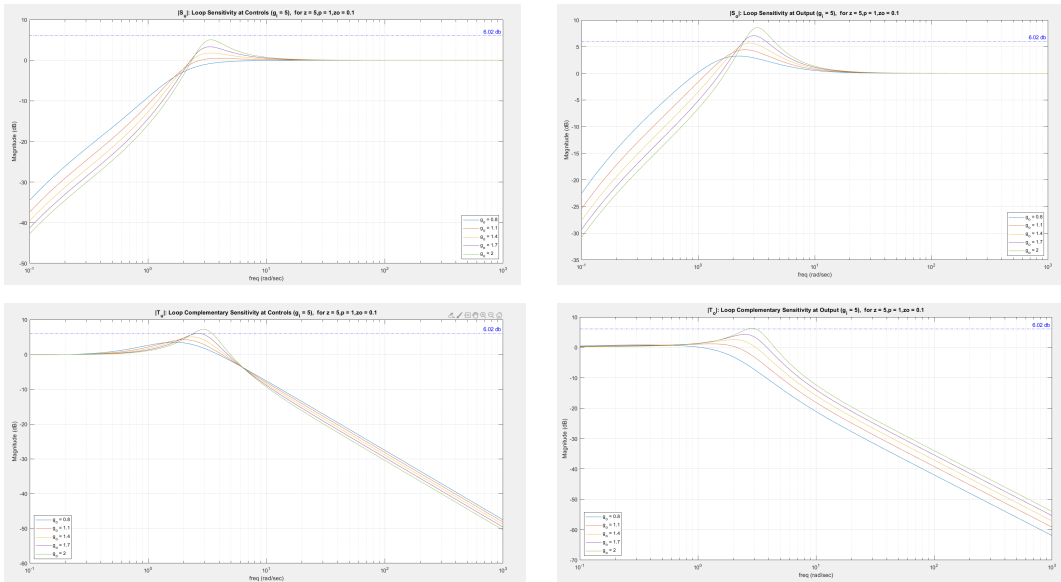
Figure 2.2.9: $|S_o|$ and $|S_u|$: Loop Sensitivity at Output and at Controls ($g_i = 5$)

- Peak S_u, $|S_o|$, T_u, T_o increases with increasing g_o or decreasing g_i.

Inner Outer Loop Complementary Sensitivity Ratio $\left|\frac{T_u}{T_o}\right|$ vs g_i (g_o - parameter)
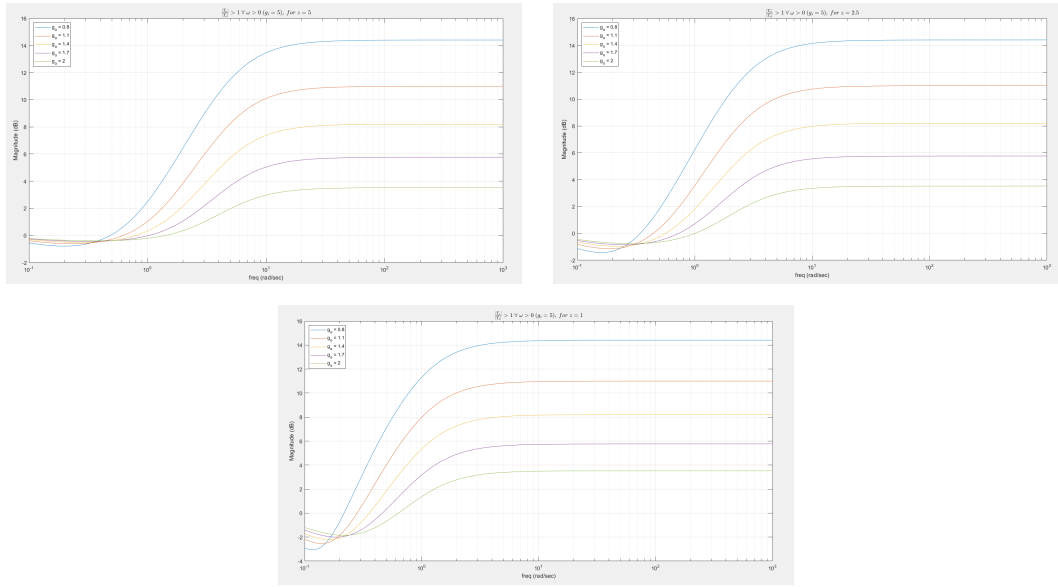
Figure 2.2.10: $\left|\frac{T_u}{T_o}\right| > 1 \ \forall \ \omega \ > 0 \ (g_i = 5)$

Peak Sensitivity and Complimentary Sensitivity
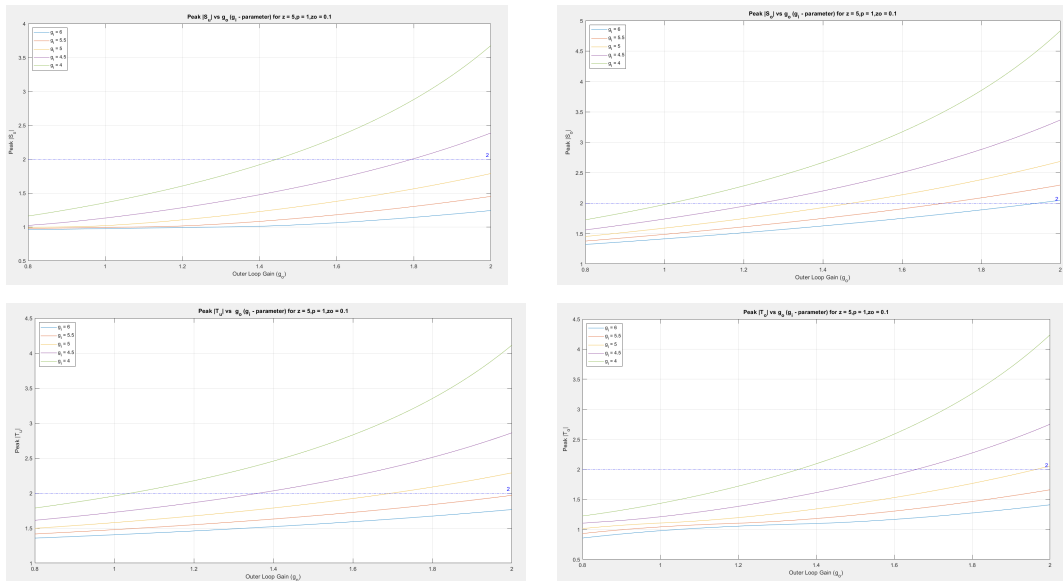
Figure 2.2.11: Peak $|S_u|$, $|S_o|$, $|T_u|$, $|T_o|$ vs g_o (g_i-parameter)

Varying Outer Loop Gain g_i with g_o fixed at $g_o = 2$

Inner Loop Sensitivity and Complementary Sensitivity
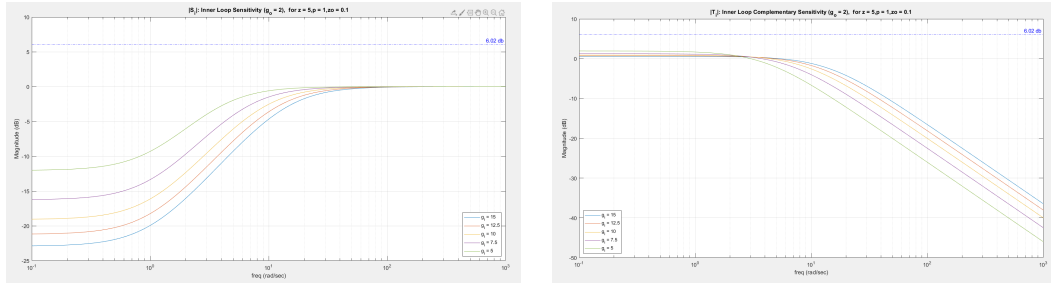
Figure 2.2.12: $|S_i|$ and $|T_i|$: Inner Loop Sensitivity and Complementary Sensitivity

$(g_o = 2)$

Loop Sensitivity and Complimentary sensitivity at Output and at Controls
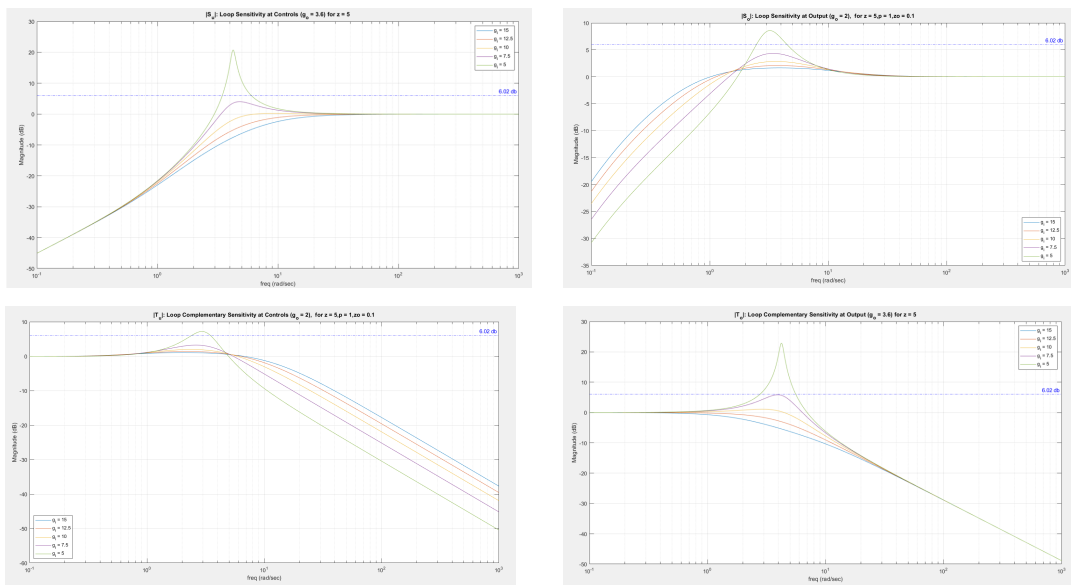
Figure 2.2.13: $|S_u|$, $|S_o|$, $|T_u|$, $|T_o|$: Loop Sensitivity at Output and at Controls

$(g_o = 2)$

$\left|\frac{T_u}{T_o}\right|$ **vs** g_i **(**g_o **- parameter)**
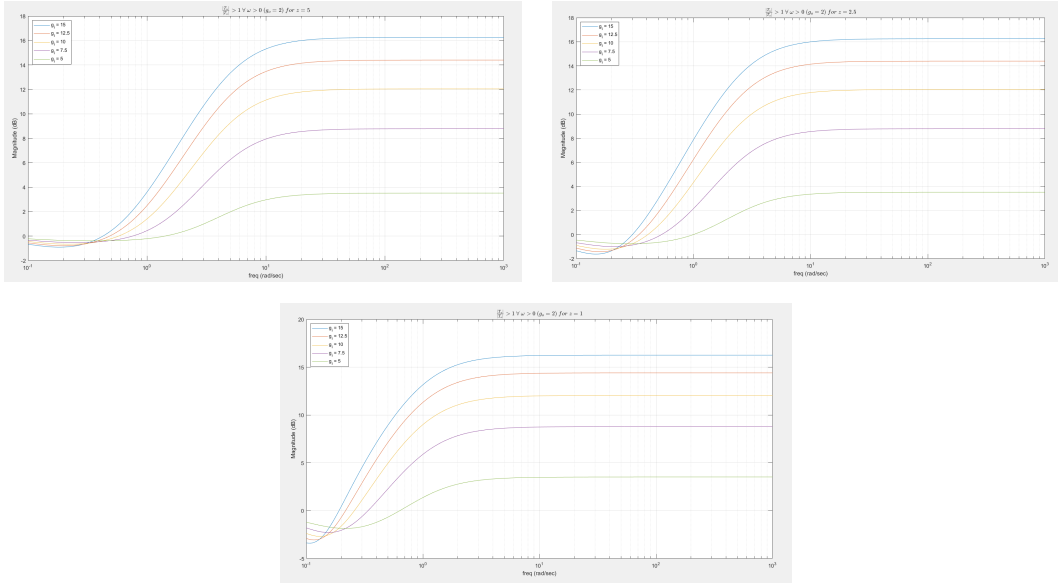
Figure 2.2.14: $\left|\frac{T_u}{T_o}\right| > 1 \;\; \forall \;\; \omega \;\; > 0 \; (g_o = 2)$

Peak $|S_o|$ **,** $|S_u|$**,** $|T_o|$**,** $|T_u|$
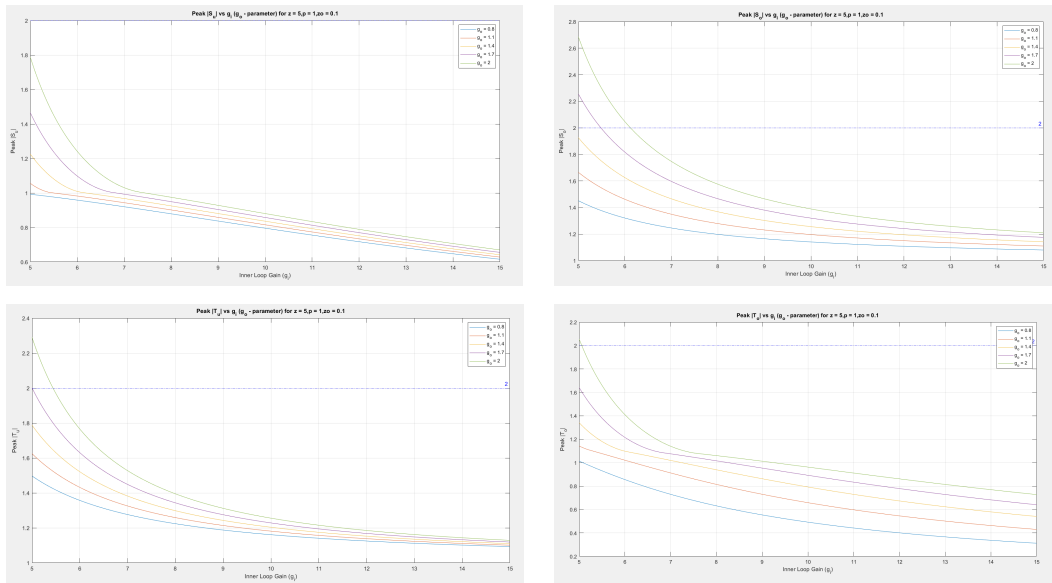
Figure 2.2.15: Peak $|S_o|$, $|S_u|$, $|T_o|$, $|T_u|$ vs g_o (g_i-parameter)

- Peak S_u, $|S_o|$, T_u, T_o increases with increasing g_o or decreasing g_i.

42

$|S_u|$, $|S_o|$ aggressive and non-aggressive behavior

$|S_u|$ aggressive behavior
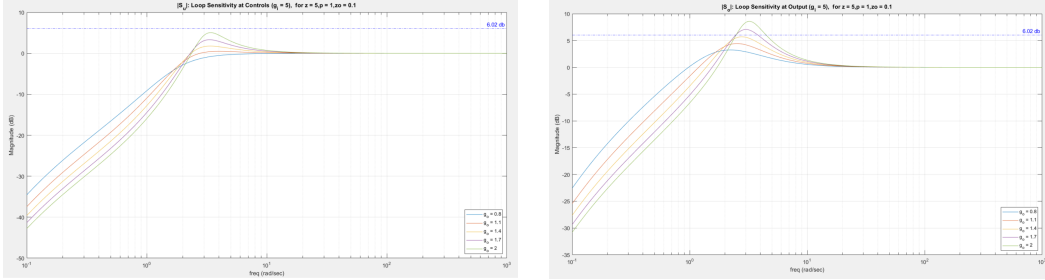
Figure 2.2.16: $|S_u|$ aggressive behavior

$|S_u|$ non aggressive behavior
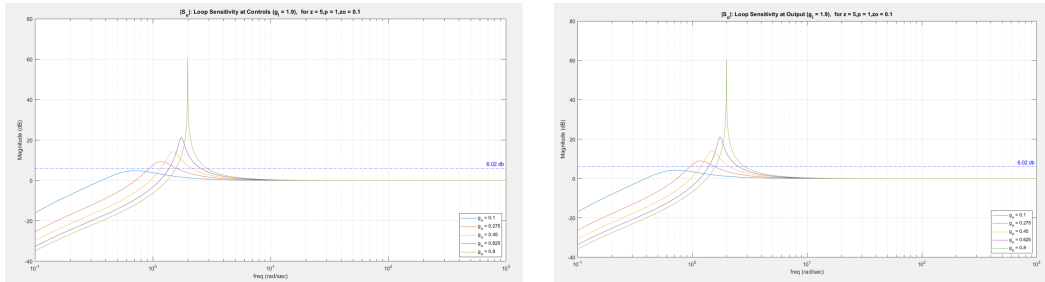
Figure 2.2.17: $|S_u|$ non aggressive behavior

- $|S_i| < 1 \Rightarrow |S_o| > |S_u|$, $|S_i| > 1 \Rightarrow |S_u| > |S_o|$. The peaks are denoted as α. The frequencies corresponding to them are denoted as ω_{peak}. The ω equations can be derived by substituting $j\omega$ in the denominator of the closed loop transfer functions and computing with respect to ω. Peak $|S_o|$, peak $|S_u|$, peak $|T_o|$, peak $|T_u|$ increases with increasing g_o or decreasing g_i.

- $|L_i|$ is independent of g_o, decreases with g_i and independent of z value.

- $|S_i|$ is independent of g_o and increases with decreasing g_i. It also is independent of z value. $|T_i|$ is independent of g_o and decreases with decreasing g_i

43

2.2.5 Closed Loop Time Response Trade Studies

Varying Outer Loop Gain g_o with Inner Loop Gain g_i Fixed at $g_i = 5$

Time Domain Analysis: Impulse S_o, S_u, T_o, T_u

Figure 2.2.18: Impulse S_o, S_u, T_o, T_u ($g_i = 5$)

Time Domain Analysis: Step S_o, S_u, T_o, T_u
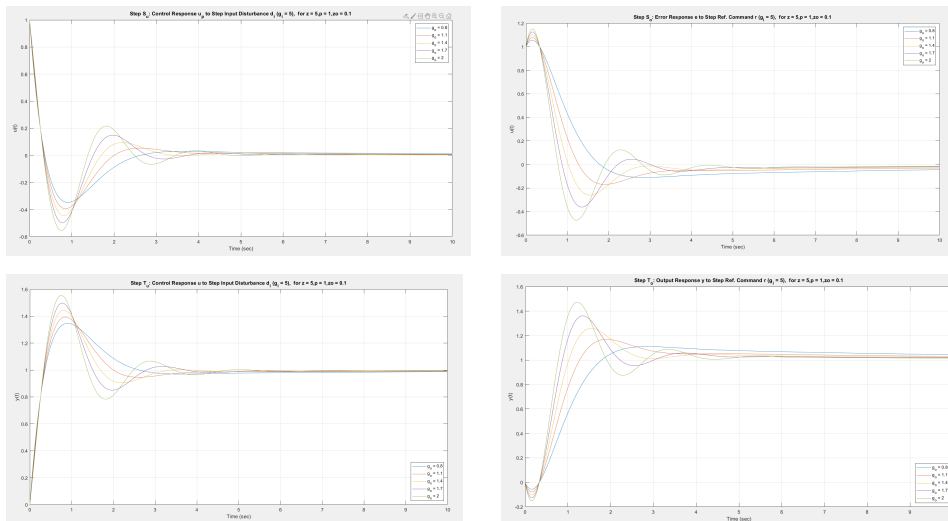
Figure 2.2.19: Step S_o, S_u, T_o, T_u ($g_i = 5$)

Varying Outer Loop Gain g_o with Inner Loop Gain g_i Fixed at $g_o = 2$

Time Domain Analysis: Impulse S_o, S_u, T_o, T_u
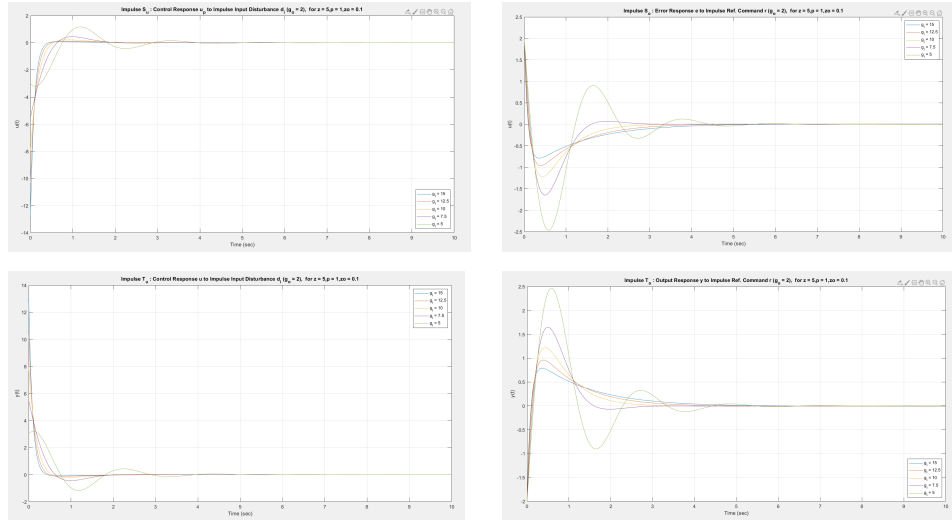
Figure 2.2.20: Impulse S_o, S_u, T_o, T_u $(g_o = 2)$

Time Domain Analysis: Step S_o, S_u, T_o, T_u
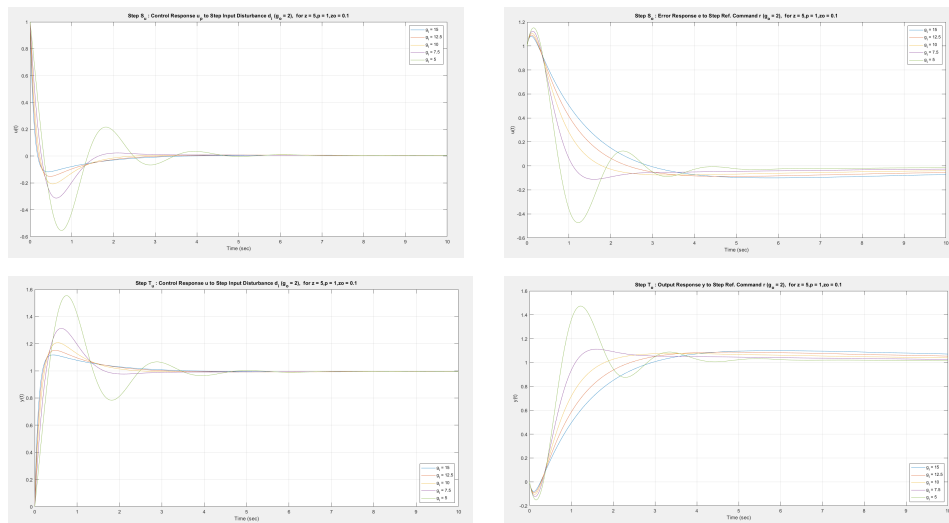
Figure 2.2.21: Step S_o, S_u, T_o, T_u $(g_o = 2)$

- Zero steady state error due to integrator in K_o.

45

2.2.6 Summary and Conclusions

	$g_o \uparrow$	$g_i \downarrow$	$z \downarrow$
$|L_u|$	$\uparrow$	$\downarrow$	$\downarrow$
$\angle L_u$	$\downarrow$	$\downarrow$	$\uparrow$
$|L_o|$	$\uparrow$	$\uparrow$	$\downarrow$
$\angle L_o$	$-$	$\downarrow$	$\downarrow$
$\downarrow GM$	$\uparrow$	$\uparrow$	$-$
$\omega_{\downarrow GM}$	$\uparrow$	$\uparrow$	$\downarrow$
$\uparrow GM$	$\downarrow$	$\downarrow$	$-$
$\omega_{\uparrow GM}$	$-$	$\downarrow$	$\downarrow$
PM_u	$\downarrow$	$\downarrow$	$\uparrow$
PM_o	$\downarrow$	$\downarrow$	$\downarrow$
DM_u	$\uparrow$	$\uparrow$	$\uparrow$
DM_o	$\downarrow$	$\downarrow$	$\uparrow$
ω_{gu}	$\downarrow$	$\downarrow$	$\uparrow$
ω_{go}	$\uparrow$	$\uparrow$	$\downarrow$
$PeakS_u$	$\uparrow$	$\uparrow$	$\downarrow$
$PeakS_o$	$\uparrow$	$\uparrow$	$\downarrow$
$PeakT_u$	$\uparrow$	$\uparrow$	$\downarrow$
$PeakT_o$	$\uparrow$	$\uparrow$	$\downarrow$
$|S_i|$	$-$	$\uparrow$	$-$
$|T_i|$	$-$	$\downarrow$	$-$

Table 2.2.2: Summary: Example 2

2.3   Example 3 Set Up and Assumptions

Block Diagram. The relevant block diagram can be found in Figure 2.3.1.
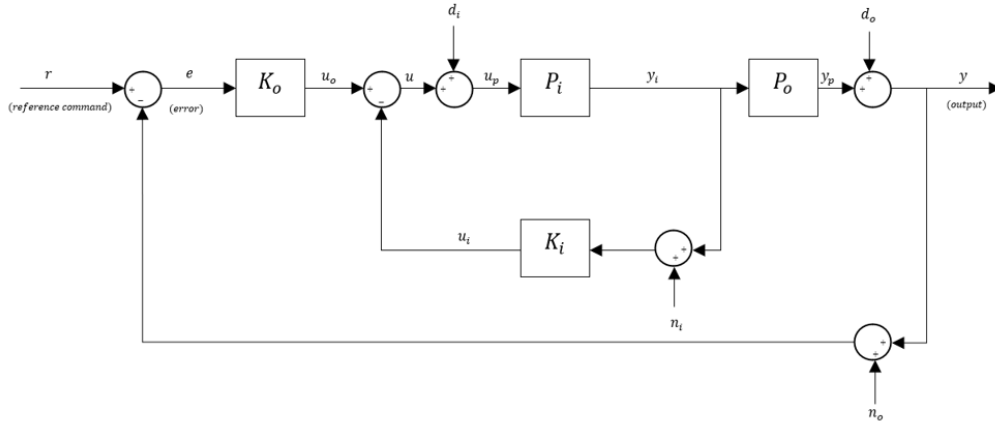
Figure 2.3.1: Block Diagram for Simple System

Nominal Parameters:

Parameter	Nominal Value	Description
p	1	Pole of P_i
z	5	Zero of P_o
g_o	(0.8-2)	Outer Loop Controller K_o Gain (will be varied)
g_i	(5-15)	Inner Loop Controller K_i Gain (will be varied)
z_{o1}	(0.01)	Outer Loop Controller K_o zero
z_{o2}	(0.01)	Outer Loop Controller K_o zero

Table 2.3.1: Nominal Parameter Values

This example introduces two zeros in the outer loop. The analogy developed in this example assists with the examples as we progress through each of them. In agreement with the parameters mentioned above the following trade studies are very insightful with respect to the cart inverted pendulum ideas.

47

Nominal System:

$$P_i = \frac{1}{s - p}, \quad K_i = g_i, \quad g_i > 0 \tag{2.3.1}$$

$$P_o = \frac{z - s}{s^2}, \quad K_o = \frac{g_o(s + z_{o1})(s + z_{o2})}{s}, \quad g_o > 0, \quad z_{o1} > 0, \quad z_{o2} > 0 \tag{2.3.2}$$

Inner Loop Transfer Function:

$$L_i = P_i K_i = \frac{g_i}{s - p} \tag{2.3.3}$$

Inner Loop Sensitivity:

$$S_i = \frac{1}{1 + L_i} = \frac{s - p}{s + (g_i - p)} \tag{2.3.4}$$

Inner Loop Complementary Sensitivity:

$$T_i = \frac{L_i}{1 + L_i} = \frac{g_i}{s + (g_i - p)} \tag{2.3.5}$$

Loop Transfer Function at Output:

$$L_o = P_o \left(\frac{P_i}{1 + P_i K_i} \right) K_o = \frac{g_o}{s^3} \left[\frac{(z - s)(s + z_{o1})(s + z_{o2})}{s + (g_i - p)} \right] \tag{2.3.6}$$

Loop Transfer Function at Controls:

$$L_u = K_i P_i + K_o P_o P_i = \frac{(b_1)s^3 + g_o(b_2)s^2 + g_o(b_3)s + g_o z z_{o1} z_{o2}}{s^3(s - p)} \tag{2.3.7}$$

Sensitivity at Output:

$$S_o = S_e = \frac{1}{1 + L_o} = \frac{s^3(s + g_i - p)}{s^4 + (b_1)s^3 + g_o(b_2)s^2 + g_o(b_3)s + g_o z z_{o1} z_{o2}} \qquad (2.3.8)$$

Sensitivity at Controls:

$$S_u = \frac{1}{1 + L_u} = \frac{s^3(s - p)}{s^4 + (b_1)s^3 + g_o(b_2)s^2 + g_o(b_3)s + g_o z z_{o1} z_{o2}} \qquad (2.3.9)$$

Inner Outer Loop Identity:

$$S_i = \frac{S_u}{S_o} \qquad (2.3.10)$$

Complementary Sensitivity at Output:

$$T_o = T_e = \frac{L_o}{1 + L_o} = \frac{g_o(z - s)(s + z_{o1})(s + z_{o2})}{s^4 + (b_1)s^3 + g_o(b_2)s^2 + g_o(b_3)s + g_o z z_{o1} z_{o2}} \qquad (2.3.11)$$

Complementary Sensitivity at Controls:

$$T_u = \frac{L_u}{1 + L_u} = \frac{(b1)s^3 + g_o(b_2)s + g_o z z_{o1} z_{o2}}{s^4 + (b_1)s^3 + g_o(b_2)s^2 + g_o(b_3} \qquad (2.3.12)$$

$$b_1 = g_i - g_o; b_2 = z - z_{o1} - z_{o2} \qquad (2.3.13)$$

$$b_3 = z z_{o1} + z z_{o2} - z_{o1} z_{o2} \qquad (2.3.14)$$

2.3.1 Fundamental Sensitivity Relationships

Similar to example 1. The tradeoff between the properties of the loop at controls and the loop at output is given by the inner-outer loop identity:

- **Identity:** $S_u = S_o S_i$

From this identity it is observed that:

- **lemma:** $\|S_i\|_{\mathcal{H}^\infty} \geq 1, \|S_o\|_{\mathcal{H}^\infty} \leq \|S_u\|_{\mathcal{H}^\infty}$: $\|S_i\|_{\mathcal{H}^\infty} \geq 1, \|T_u\|_{\mathcal{H}^\infty} \leq \|T_o\|_{\mathcal{H}^\infty}$ (since $S_o = 1 - T_o, S_u = 1 - T_u$).

- **lemma:** $\|S_i\|_{\mathcal{H}^\infty} < 1, \|S_u\|_{\mathcal{H}^\infty} < \|S_o\|_{\mathcal{H}^\infty}$: $\|S_i\|_{\mathcal{H}^\infty} < 1, \|T_o\|_{\mathcal{H}^\infty} < \|T_u\|_{\mathcal{H}^\infty}$ (since $S_o = 1 - T_o, S_u = 1 - T_u$).

Hence, the inner-loop works as a major tool to play between the peak sensitvities at output and controls.

From the above lemmas, and close inspection of the system following corollaries are obtained:

1. z-condition $= \dfrac{2(zz_o 1 + z z_{o2} - z_{o1} z_{o2})}{z - z_{o1} - z_{o2} + \sqrt{((z - z_{o1} - z_{o2})^2 - \frac{4 z z_{o1} z_{o2}}{g_o}))}}$

2. **Corollary 1: $p - $ Non-Aggressive Inner Loop:** $\|S_i\|_{\mathcal{H}^\infty} \geq 1 \iff g_o + p + z - condition < g_i \leq 2p \iff \|S_o\|_{\mathcal{H}^\infty} \leq \|S_u\|_{\mathcal{H}^\infty}$:

3. **Corollary 2: $p - $ Aggressive Inner Loop:** $\|S_i\|_{\mathcal{H}^\infty} < 1 \iff \max\{g_o + p + z - condition, 2p\} < g_i \iff \|S_o\|_{\mathcal{H}^\infty} > \|S_u\|_{\mathcal{H}^\infty}$.

4. **Corollary 3: $g - $ Aggressive Inner Loop:** $\|S_i\|_{\mathcal{H}^\infty} < 1 \iff \max\{g_o + p + z - condition, 2g_o\} < g_i \iff \|T_o\|_{\mathcal{H}^\infty} < \|T_u\|_{\mathcal{H}^\infty}$.

5. **Comment: Peak Sensitivities:** $\|S\|_{\mathcal{H}^\infty}, \|T\|_{\mathcal{H}^\infty}$ increase with increasing g_o and decreasing g_i.

2.3.2   Closed Loop Stability

Similar to example 1, we get the stability of the inner loop as:
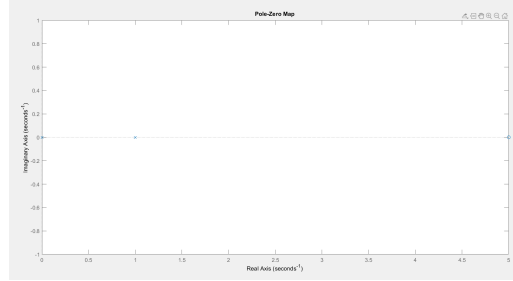
Figure 2.3.2: Nominal System: Open Loop Transfer Function

1. If $g_i \approx g_o + p + z - condition$; the system has poles near the imaginary axis which leads to higher peak sensitivities t for the loop at output and for the loop at controls.
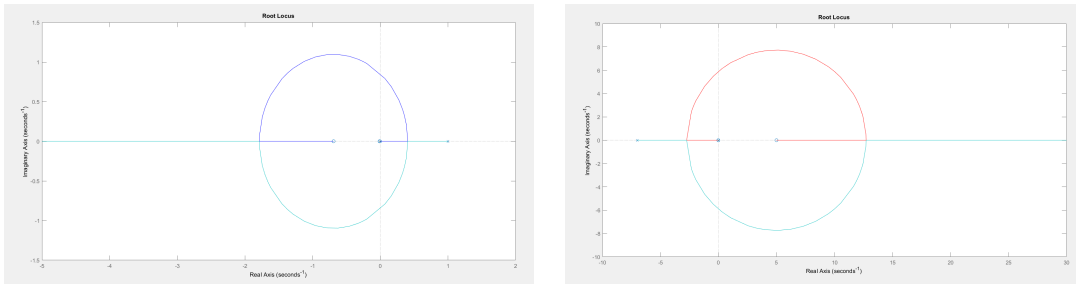
Figure 2.3.3: Nominal System: Root locus of Loop Transfer Function at Controls (left) and at Output (right)

The above plots explain that the $\uparrow GM$ at controls is infinite, $\downarrow GM$ at controls is finite. On the other hand, $\downarrow GM$ at output is zero, $\uparrow GM$ is finite because of the right hand plane zero. Thus, this cause the system to be bandwidth limited.

51

Varying Outer Loop Gain g_o with g_i fixed at $g_i = 5$

Loop at Input:
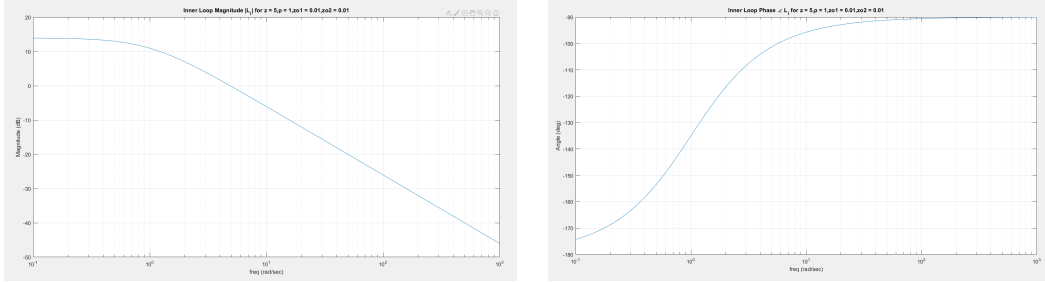
Figure 2.3.4: Inner Loop Magnitude $|L_i|$ and Phase $\angle L_i$

Loop at Output and Loop at Controls:
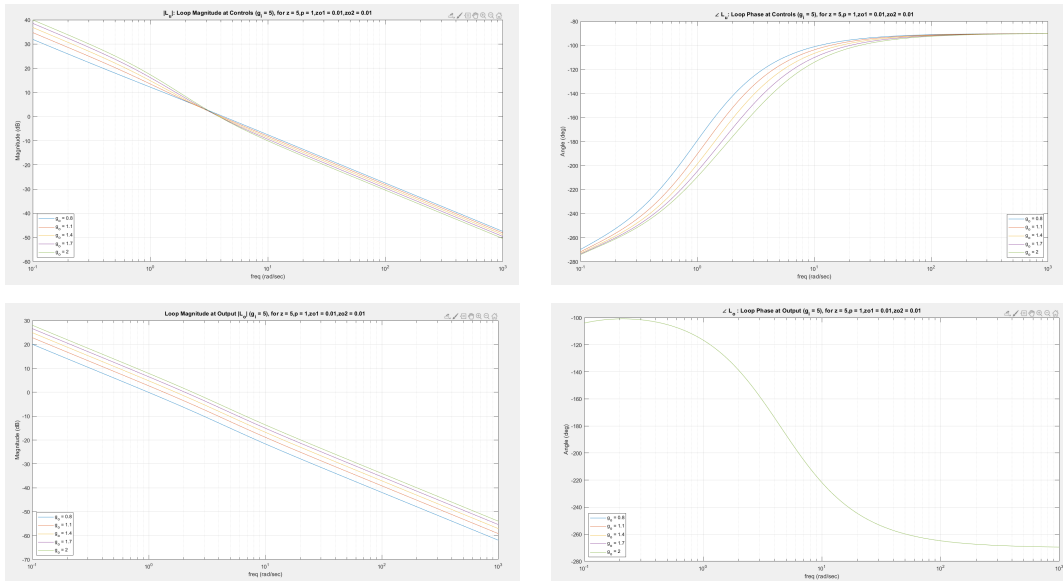
Figure 2.3.5: L_u and L_o: Loop Magnitude and Loop Phase ($g_i = 5$) for $z = 5$

- $|L_o|$ increases with increasing g_o and increases with decreasing g_i. $\angle L_o$ is independent of g_o and decreases wtih decreasing g_i.

Varying Outer Loop Gain g_i **with** g_o **fixed at** $g_o = 2$

Loop at Input:
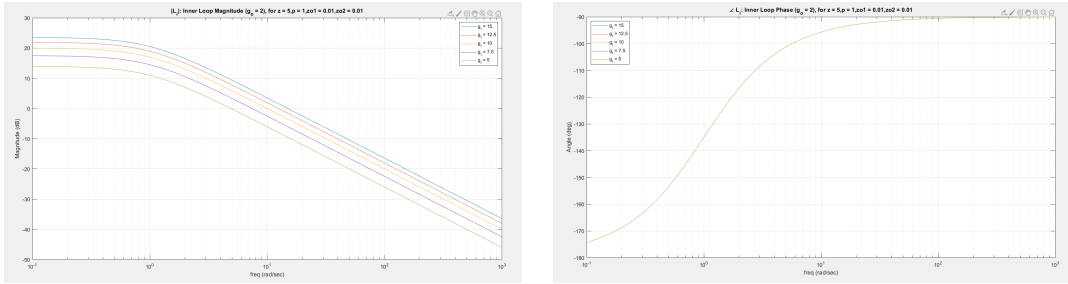
Figure 2.3.6: Inner Loop Magnitude $|L_i|$ and Phase $\angle L_i$

Loop at Output and Loop at Controls:
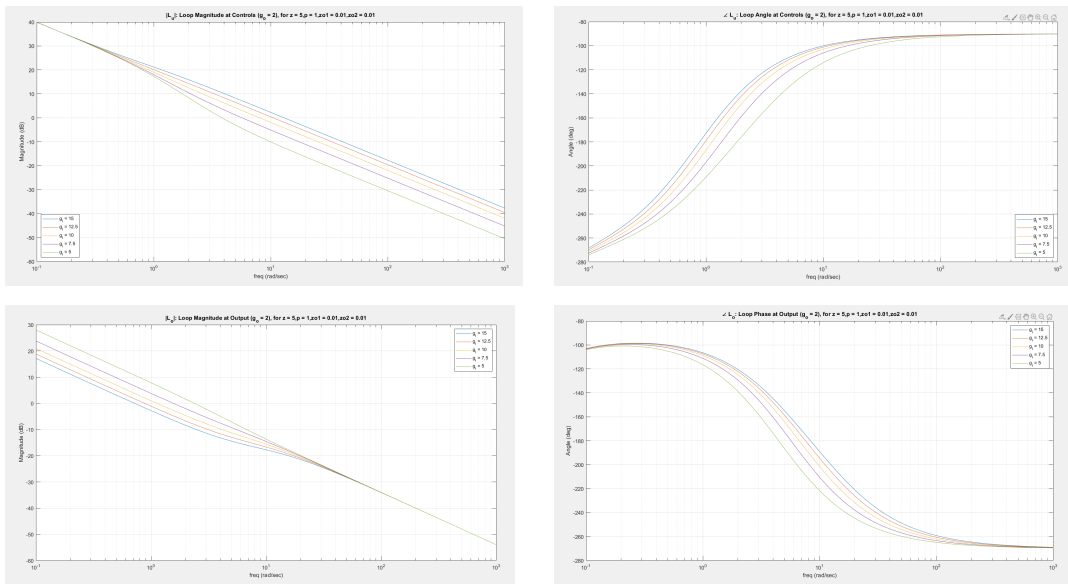
Figure 2.3.7: L_u and L_o: Loop Magnitude and Loop Phase $(g_o = 2)$

- $|L_o|$ increases with increasing g_o and increases with decreasing g_i. $\angle L_o$ is independent of g_o and decreases wtih decreasing g_i. $|L_u|$ increases with increasing g_o and decreases with decreasing g_i. $\angle L_u$ decreases with increasing g_o and decreases wtih decreasing g_i.

Varying Outer Loop Gain g_o with Inner Loop Gain g_i Fixed at $g_i = 5$

Inner Loop Sensitivity and Complementary Sensitivity
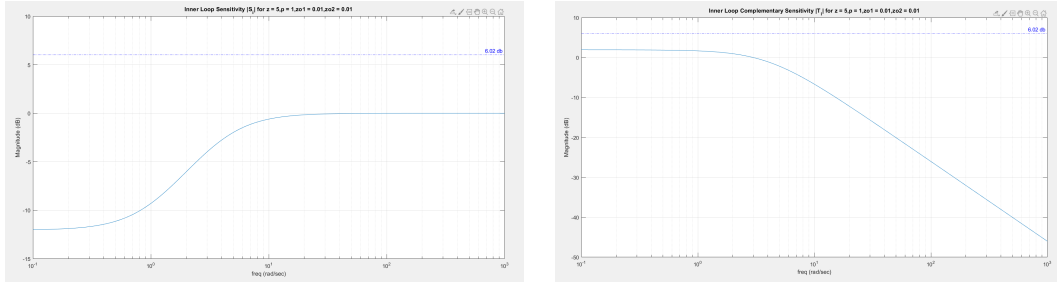
Figure 2.3.8: Inner Loop Sensitivity and Complementary Sensitivity $|S_i|$ and $|T_i|$

Loop Sensitivity at Output and at Controls
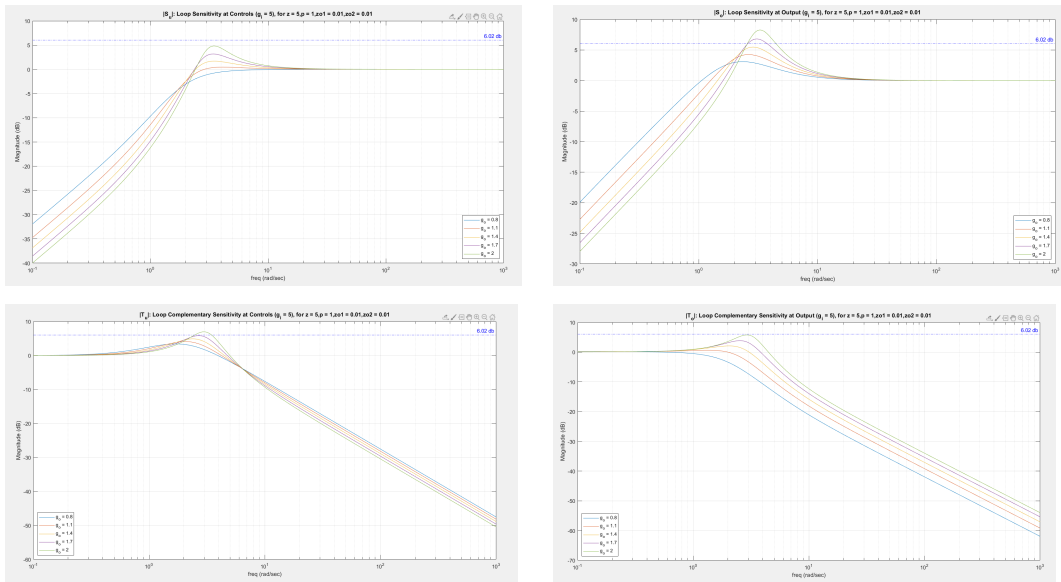
Figure 2.3.9: $|S_o|$ and $|S_u|$: Loop Sensitivity at Output and at Controls ($g_i = 5$)

- Peak $|S_o|$, $|S_u|$ increases with increasing g_o or decreasing g_i, peak $|T_o|$, $|T_u|$ increases with increasing g_o or decreasing g_i.

54

Inner Outer Loop Complementary Sensitivity Ratio $\left|\frac{T_u}{T_o}\right|$ vs g_i (g_o - parameter)
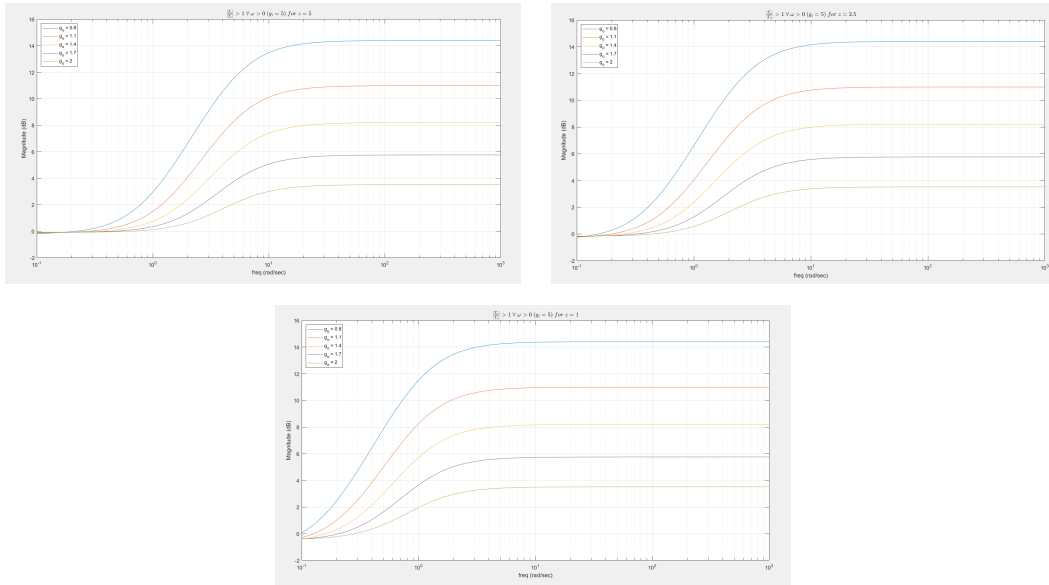
Figure 2.3.10: $\left|\frac{T_u}{T_o}\right| > 1 \ \forall \ \omega \ > 0 \ (g_i = 5)$

Peak Sensitivity and Complimentary Sensitivity
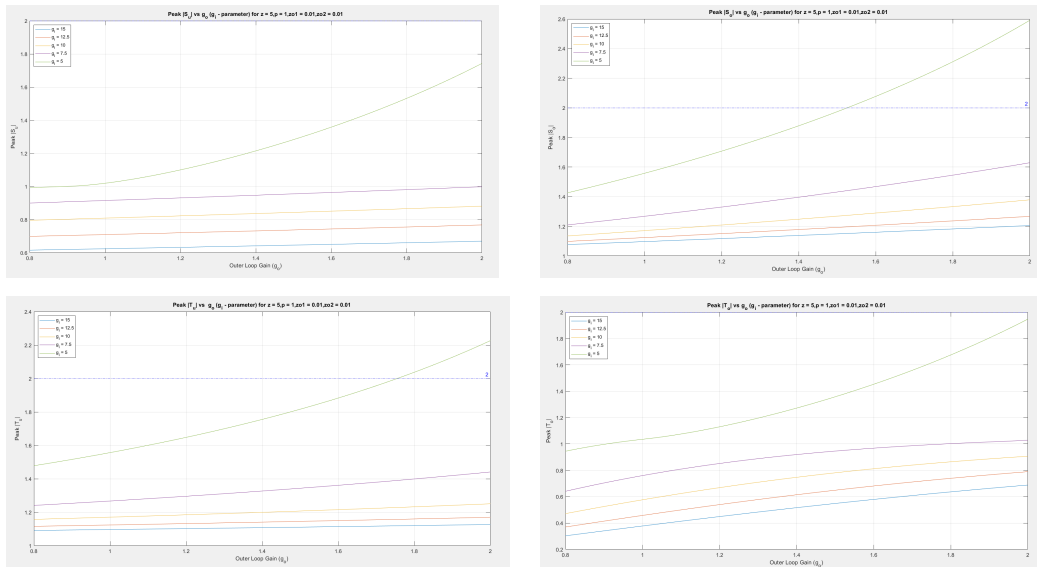
Figure 2.3.11: Peak $|S_u|$, $|S_o|$, $|T_u|$, $|T_o|$ vs g_o (g_i-parameter)

Varying Outer Loop Gain g_i with g_o fixed at $g_o = 2$

Inner Loop Sensitivity and Complementary Sensitivity

Figure 2.3.12: $|S_i|$ and $|T_i|$: Inner Loop Sensitivity and Complementary Sensitivity $(g_o = 2)$

Loop Sensitivity and Complimentary sensitivity at Output and at Controls
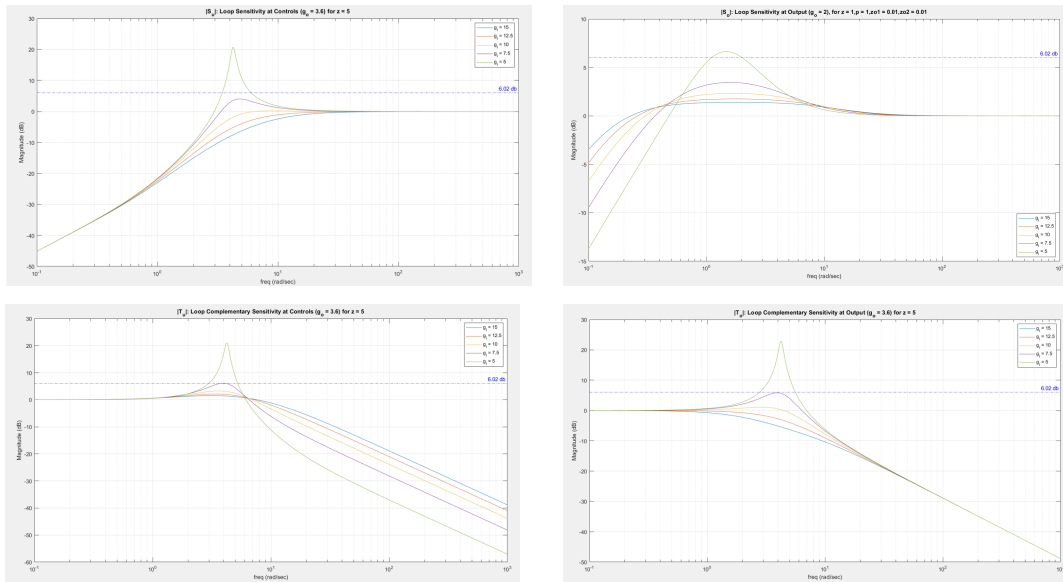
Figure 2.3.13: $|S_u|$, $|S_o|$, $|T_u|$, $|T_o|$: Loop Sensitivity at Output and at Controls $(g_o = 2)$

- $|S_i|,|T_i|$ is independent of g_o and decreases with decreasing g_i, independent of z.

56

$|\frac{T_u}{T_o}|$ **vs** g_i **(**g_o **- parameter)**
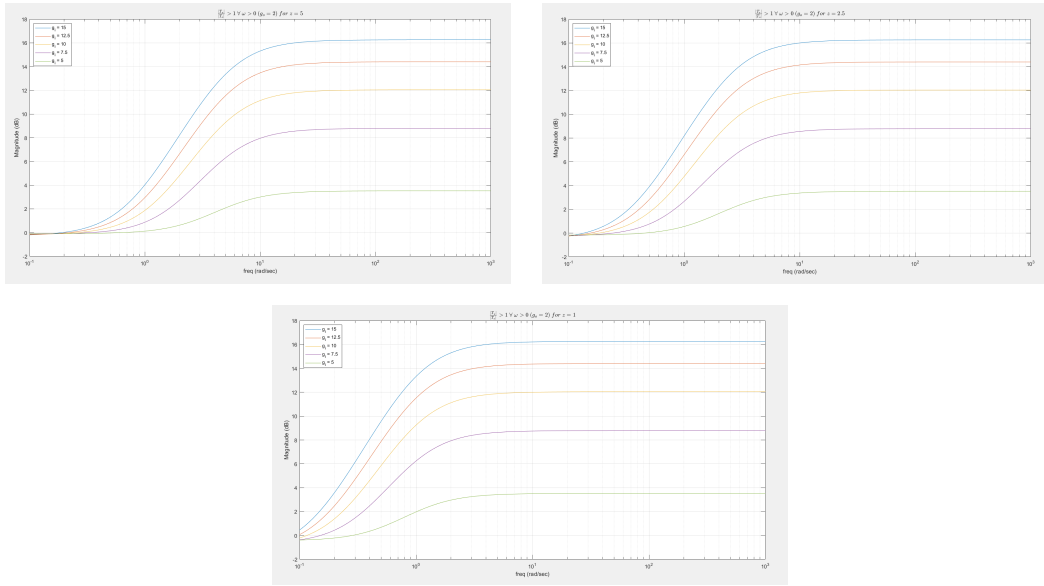
Figure 2.3.14: $|\frac{T_u}{T_o}| > 1 \ \forall \ \omega \ > 0 \ (g_o = 2)$

Peak $|S_o|$ **,** $|S_u|$, $|T_o|$, $|T_u|$

Figure 2.3.15: Peak $|S_o|$, $|S_u|$, $|T_o|$, $|T_u|$ vs g_o (g_i-parameter)

- Peak $|S_u|$, $|S_o|$, $|T_u|$, $|T_o|$ increases with increasing g_o or decreasing g_i.

$|S_u|$, $|S_o|$ aggressive and non-aggressive behavior

$|S_u|$ aggressive behavior
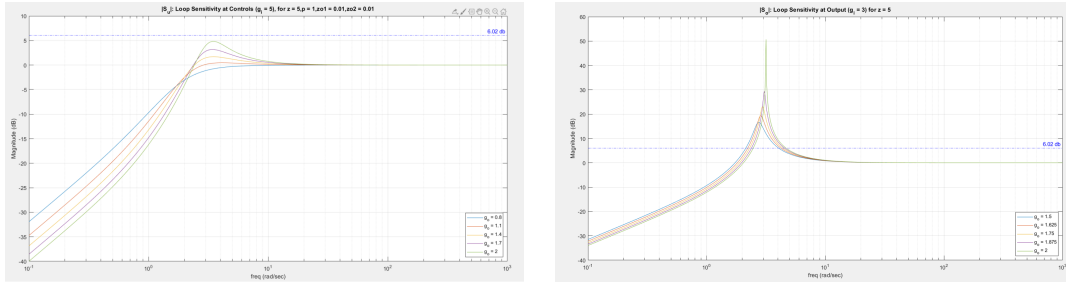
Figure 2.3.16: $|S_u|$ aggressive behavior

$|S_u|$ non aggressive behavior
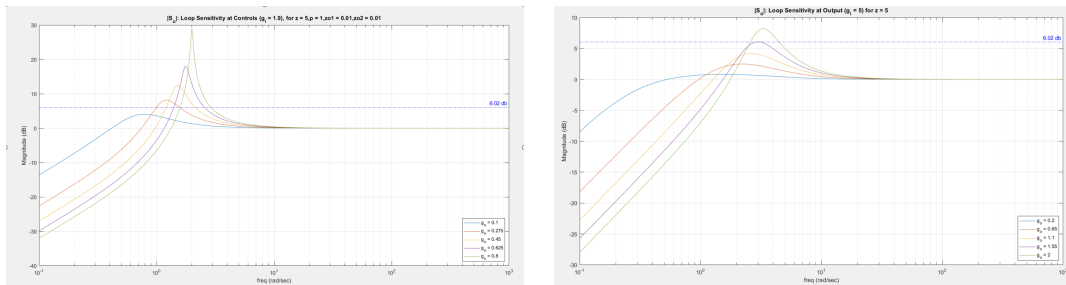
Figure 2.3.17: $|S_u|$ non aggressive behavior

- $|S_i| < 1 \Rightarrow |S_o| > |S_u|$, $|S_i| > 1 \Rightarrow |S_u| > |S_o|$. The peaks are denoted as α. The frequencies corresponding to them are denoted as ω_{peak}. The ω equations can be derived by substituting $j\omega$ in the denominator of the closed loop transfer functions and computing with respect to ω. Peak $|S_o|$, peak $|S_u|$, peak $|T_o|$, peak $|T_u|$ increases with increasing g_o or decreasing g_i. $|L_i|$ is independent of g_o, decreases with g_i and independent of z value.

- $|S_i|$ is independent of g_o and increases with decreasing g_i

Varying Outer Loop Gain g_o with Inner Loop Gain g_i Fixed at $g_i = 5$

Time Domain Analysis: Impulse S_o, S_u, T_o, T_u

Figure 2.3.18: Impulse S_o, S_u, T_o, T_u $(g_i = 5)$

Time Domain Analysis: Step S_o, S_u, T_o, T_u
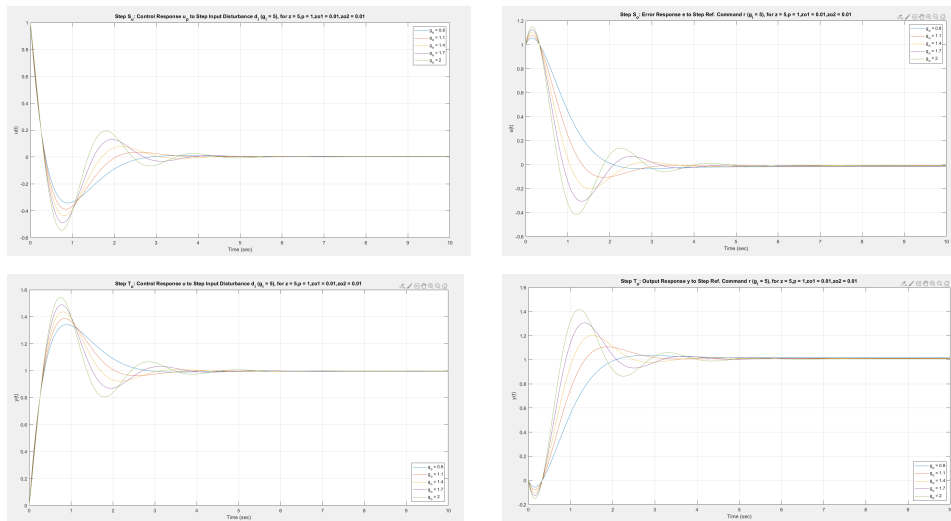
Figure 2.3.19: Step S_o, S_u, T_o, T_u $(g_i = 5)$

Varying Outer Loop Gain g_o with Inner Loop Gain g_i Fixed at $g_o = 2$

Time Domain Analysis: Impulse S_o, S_u, T_o, T_u
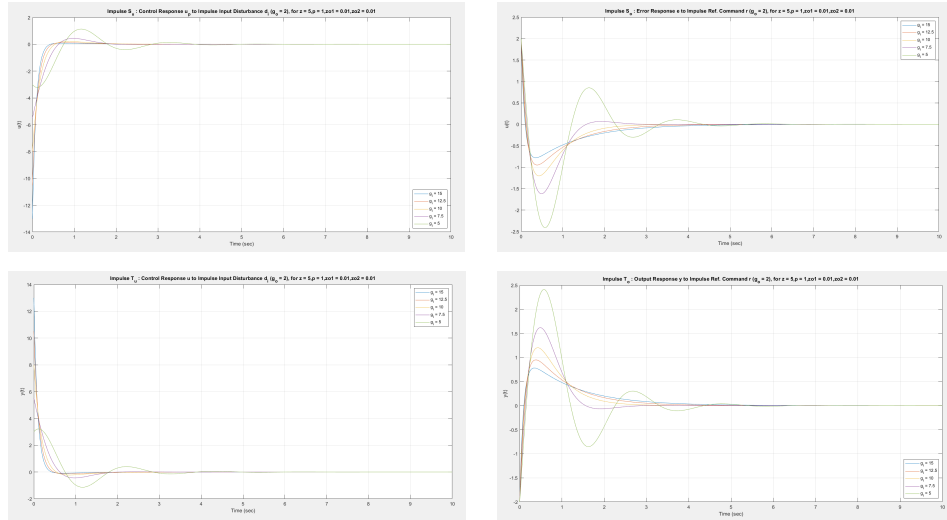
Figure 2.3.20: Impulse S_o, S_u, T_o, T_u ($g_o = 2$)

Time Domain Analysis: Step S_o, S_u, T_o, T_u

Figure 2.3.21: Step S_o, S_u, T_o, T_u ($g_o = 2$)

- , Initial transient undershoot increases with increasing g_o.

2.3.6 Summary and Conclusions

	$g_o \uparrow$	$g_i \downarrow$	$z \downarrow$
$|L_u|$	$\uparrow$	$\downarrow$	$\downarrow$
$\angle L_u$	$\downarrow$	$\downarrow$	$\uparrow$
$|L_o|$	$\uparrow$	$\uparrow$	$\downarrow$
$\angle L_o$	$-$	$\downarrow$	$\downarrow$
$\downarrow GM$	$\uparrow$	$\uparrow$	$-$
$\omega_{\downarrow GM}$	$\uparrow$	$\uparrow$	$\downarrow$
$\uparrow GM$	$\downarrow$	$\downarrow$	$-$
$\omega_{\uparrow GM}$	$-$	$\downarrow$	$\downarrow$
PM_u	$\downarrow$	$\downarrow$	$\uparrow$
PM_o	$\downarrow$	$\downarrow$	$\downarrow$
DM_u	$\uparrow$	$\uparrow$	$\uparrow$
DM_o	$\downarrow$	$\downarrow$	$\uparrow$
ω_{gu}	$\downarrow$	$\downarrow$	$\uparrow$
ω_{go}	$\uparrow$	$\uparrow$	$\downarrow$
$PeakS_u$	$\uparrow$	$\uparrow$	$\downarrow$
$PeakS_o$	$\uparrow$	$\uparrow$	$\downarrow$
$PeakT_u$	$\uparrow$	$\uparrow$	$\downarrow$
$PeakT_o$	$\uparrow$	$\uparrow$	$\downarrow$
$|S_i|$	$-$	$\uparrow$	$-$
$|T_i|$	$-$	$\downarrow$	$-$

Table 2.3.2: Summary: Example 3

2.4   Example 4 Set Up and Assumptions

Block Diagram. The relevant block diagram can be found in Figure 2.4.1.

Figure 2.4.1: Block Diagram for Simple System

Nominal Parameters:

Parameter	Nominal Value	Description
p	1	Pole of P_i
z	5	Zero of P_o
g_o	(0.8-2)	Outer Loop Controller K_o Gain (will be varied)
g_i	(5-15)	Inner Loop Controller K_i Gain (will be varied)
z_i	(0.1)	Inner Loop Controller K_i Zero
z_{o1}	(0.01)	Outer Loop Controller K_o Zero
z_{o2}	(0.02)	Outer Loop Controller K_o Zero

Table 2.4.1: Nominal Parameter Values

This example introduces two zeros in the outer loop and one zero in the inner loop. In agreement with the parameters mentioned above the following trade studies are very insightful with respect to the cart inverted pendulum ideas.

Nominal System:

$$P_i = \frac{1}{(s-p)}, \quad K_i = \frac{g_i(s+z_i)}{s}, \quad g_i > 0, z_i > 0 \tag{2.4.1}$$

$$P_o = \frac{z-s}{s^2}, \quad K_o = \frac{g_o(s+z_{o1})(s+z_{o2}))}{s}, \quad g_o > 0, \quad z_{o1} > 0, \quad z_{o2} > 0 \tag{2.4.2}$$

Inner Loop Transfer Function:

$$L_i = P_i K_i = \frac{g_i(s+z_i)}{s(s-p)} \tag{2.4.3}$$

Inner Loop Sensitivity:

$$S_i = \frac{1}{1+L_i} = \frac{s(s-p)}{s^2 + (g_i - p)s + g_i z_i} \tag{2.4.4}$$

Inner Loop Complementary Sensitivity:

$$T_i = \frac{L_i}{1+L_i} = \frac{g_i(s+z_i)}{s^2 + (g_i - p)s + g_i z_i} \tag{2.4.5}$$

Loop Transfer Function at Output:

$$L_o = P_o \left(\frac{P_i}{1 + P_i K_i} \right) K_o = \frac{g_o(z-s)(s+z_{o1})(s+z_{o2})}{s^2(s^2 + (g_i - p)s + g_i z_i)} \tag{2.4.6}$$

Loop Transfer Function at Controls:

$$L_u = K_i P_i + K_o P_o P_i = \frac{(g_i - g_o)s^3 + as^2 + bs + g_o z z_{o1} z_{o2}}{s^3(s-p)} \tag{2.4.7}$$

Sensitivity at Output:

$$S_o = S_e = \frac{1}{1 + L_o} = \frac{s^2(s^2 + (g_i - p)s + g_i z_i)}{s^4 + (g_i - p - g_o)s^3 + as^2 + bs + g_o z z_{o1} z_{o2}} \qquad (2.4.8)$$

Sensitivity at Controls:

$$S_u = \frac{1}{1 + L_u} = \frac{s^3(s - p)}{s^4 + (g_i - p - g_o)s^3 + as^2 + bs + g_o z z_{o1} z_{o2}} \qquad (2.4.9)$$

Inner Outer Loop Identity:

$$S_i = \frac{S_u}{S_o} \qquad (2.4.10)$$

Complementary Sensitivity at Output:

$$T_o = T_e = \frac{L_o}{1 + L_o} = \frac{g_o(z - s)(s + z_{o1})(s + z_{o2})}{s^4 + (g_i - p - g_o)s^3 + as^2 + bs + g_o z z_{o1} z_{o2}} \qquad (2.4.11)$$

Complementary Sensitivity at Controls:

$$T_u = \frac{L_u}{1 + L_u} = \frac{(g_i - g_o)s^3 + as^2 + bs + g_o z z_{o1} z_{o2}}{s^4 + (g_i - p - g_o)s^3 + as^2 + bs + g_o z z_{o1} z_{o2}} \qquad (2.4.12)$$

Complementary Sensitivity Identities:

$$T_u - T_o = T_i S_u \qquad (2.4.13)$$

$$T_u - T_i = T_o S_i \qquad (2.4.14)$$

$$a = g_o z + g_i z_i - g_o z_{o1} - g_o z_{o2}; b = g_o z z_{o1} + g_o z z_{o2} - g_o z_{o1} z_{o2}; \qquad (2.4.15)$$

$$S_o = T_{re}; \qquad T_o = T_{ry}; \qquad S_u = T_{d_i u_p}; \qquad T_u = T_{d_i u} \qquad (2.4.16)$$

2.4.1 Fundamental Sensitivity Relationships

Similar to example 1. The tradeoff between the properties of the loop at controls and the loop at output is given by the inner-outer loop identity:

- **Identity:** $S_u = S_o S_i$

From this identity it is observed that:

- **lemma:** $\|S_i\|_{\mathcal{H}^\infty} \geq 1, \|S_o\|_{\mathcal{H}^\infty} \leq \|S_u\|_{\mathcal{H}^\infty} : \|S_i\|_{\mathcal{H}^\infty} \geq 1, \|T_u\|_{\mathcal{H}^\infty} \leq \|T_o\|_{\mathcal{H}^\infty}$ (since $S_o = 1 - T_o, S_u = 1 - T_u$).

- **lemma:** $\|S_i\|_{\mathcal{H}^\infty} < 1, \|S_u\|_{\mathcal{H}^\infty} < \|S_o\|_{\mathcal{H}^\infty} : \|S_i\|_{\mathcal{H}^\infty} < 1, \|T_o\|_{\mathcal{H}^\infty} < \|T_u\|_{\mathcal{H}^\infty}$ (since $S_o = 1 - T_o, S_u = 1 - T_u$).

1. g_i - stability is very complicated to compute analytically, since the system has more parameters. So it is computed numerically. The code for computing it numerically is given in the appendix.

2. **Corollary 1:** $p -$ **Non-Aggressive Inner Loop:** $\|S_i\|_{\mathcal{H}^\infty} \geq 1 \iff g_i - stability < g_i \leq 2p \iff \|S_o\|_{\mathcal{H}^\infty} \leq \|S_u\|_{\mathcal{H}^\infty} ; z_i \approx 0:$

3. **Corollary 2:** $p -$ **Aggressive Inner Loop:** $\|S_i\|_{\mathcal{H}^\infty} < 1 \iff \max\{g_i - stability, 2p\} < g_i \iff \|S_o\|_{\mathcal{H}^\infty} > \|S_u\|_{\mathcal{H}^\infty} ; z_i \approx 0.$

4. **Corollary 3:** $g -$ **Aggressive Inner Loop:** $\|S_i\|_{\mathcal{H}^\infty} < 1 \iff \max\{g_i - stability, 2g_o\} < g_i \iff \|T_o\|_{\mathcal{H}^\infty} < \|T_u\|_{\mathcal{H}^\infty} ; z_i \approx 0.$

5. **Comment: Peak Sensitivities:** $\|S\|_{\mathcal{H}^\infty} , \|T\|_{\mathcal{H}^\infty}$ increase with increasing g_o and decreasing g_i.

2.4.2   Closed Loop Stability

Similar to example 1, we get the stability of the inner loop as:

Figure 2.4.2: Nominal System: Open Loop Transfer Function

1. If $g_i \approx stabilitycondition$; the system has poles near the imaginary axis which leads to higher peak sensitivities t for the loop at output and for the loop at controls.

Figure 2.4.3: Nominal System: Root locus of Loop Transfer Function at Controls (left) and at Output (right)

The above plots explain that the $\uparrow GM$ at controls is infinite, $\downarrow GM$ at controls is finite. On the other hand, $\downarrow GM$ at output is zero, $\uparrow GM$ is finite because of the right hand plane zero. Thus, this cause the system to be bandwidth limited.

2.4.3   Open Loop Trade Studies

Varying Outer Loop Gain g_o with g_i fixed at $g_i = 5$

Loop at Input:

Figure 2.4.4: Inner Loop Magnitude $|L_i|$ and Phase $\angle L_i$

Loop at Output and Loop at Controls:

Figure 2.4.5: L_u and L_o: Loop Magnitude and Loop Phase ($g_i = 5$) for $z = 5$

- Peak $|S_u|$, $|S_o|$, $|T_u|$, $|T_o|$ increases with increasing g_o or decreasing g_i.

Varying Outer Loop Gain g_i **with** g_o **fixed at** $g_o = 2$

Loop at Input:

Figure 2.4.6: Inner Loop Magnitude $|L_i|$ and Phase $\angle L_i$

Loop at Output and Loop at Controls:
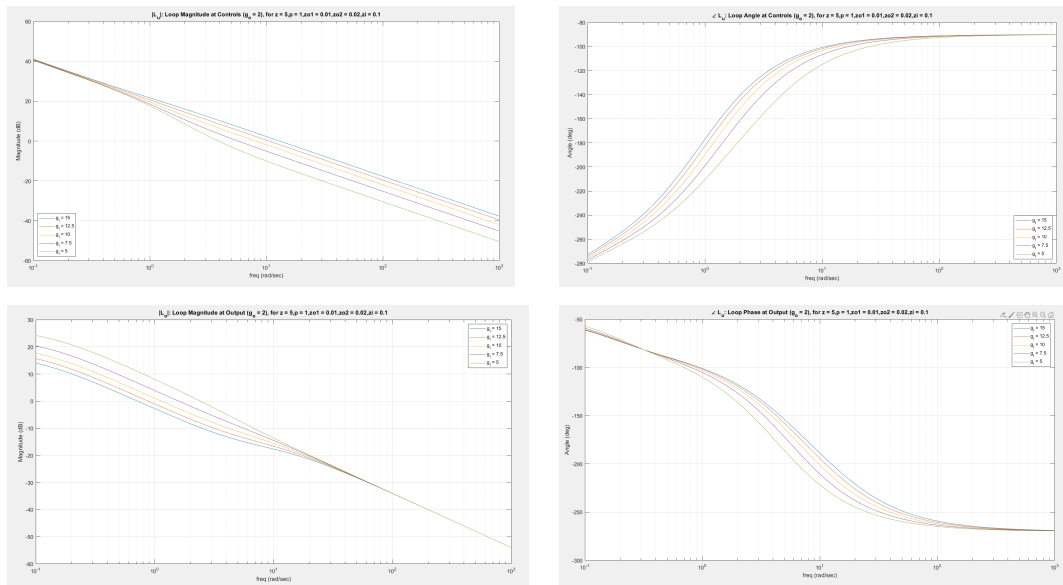
Figure 2.4.7: L_u and L_o: Loop Magnitude and Loop Phase ($g_o = 2$)

- $|L_o|$ increases with increasing g_o, ω_{go} decreases with increasing g_o, $\angle L_o$ is independent of g_o, PM is independent of g_o.

Varying Outer Loop Gain g_o with Inner Loop Gain g_i Fixed at $g_i = 5$

Inner Loop Sensitivity and Complementary Sensitivity
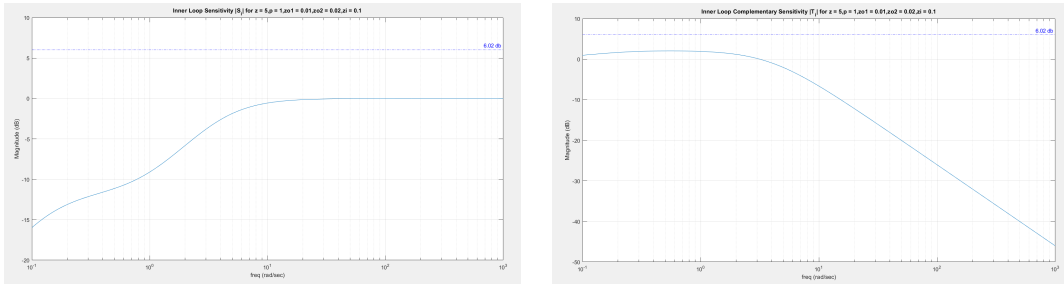
Figure 2.4.8: Inner Loop Sensitivity and Complementary Sensitivity $|S_i|$ and $|T_i|$

Loop Sensitivity at Output and at Controls
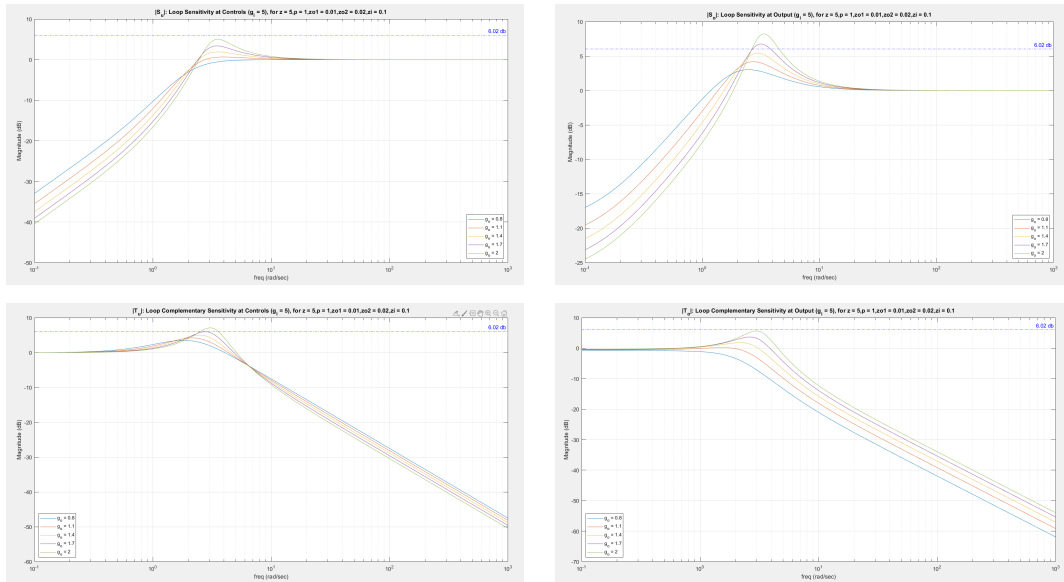
Figure 2.4.9: $|S_o|$ and $|S_u|$: Loop Sensitivity at Output and at Controls ($g_i = 5$)

- Peak $|S_o|$, $|S_u|$, $|T_o|$, $|T_u|$ increases with increasing g_o or decreasing g_i.

Inner Outer Loop Complementary Sensitivity Ratio $\left|\frac{T_u}{T_o}\right|$ vs g_i (g_o - parameter)
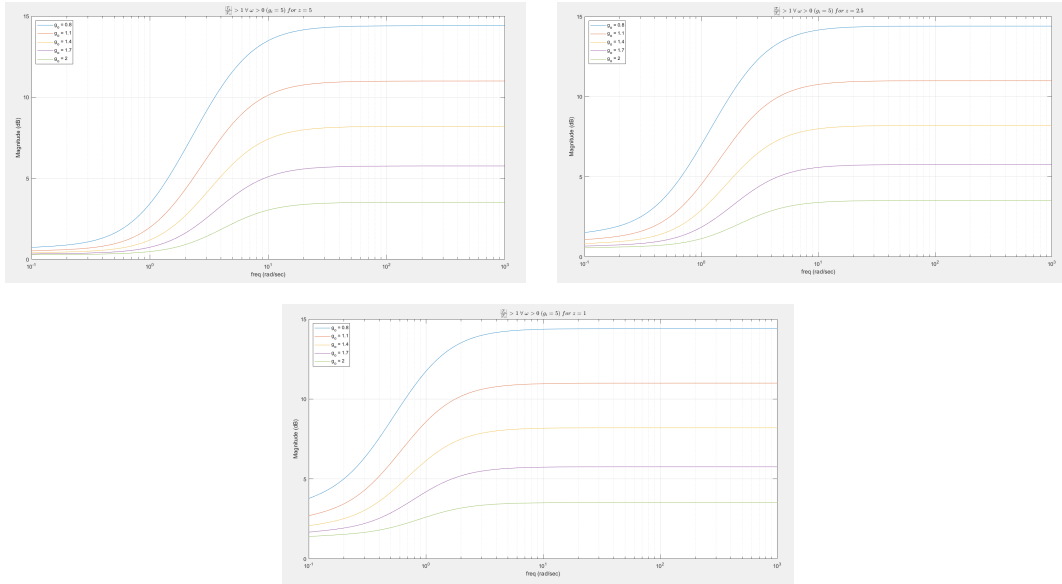
Figure 2.4.10: $\left|\frac{T_u}{T_o}\right| > 1 \ \forall \ \omega \ > 0 \ (g_i = 5)$

Peak Sensitivity and Complimentary Sensitivity
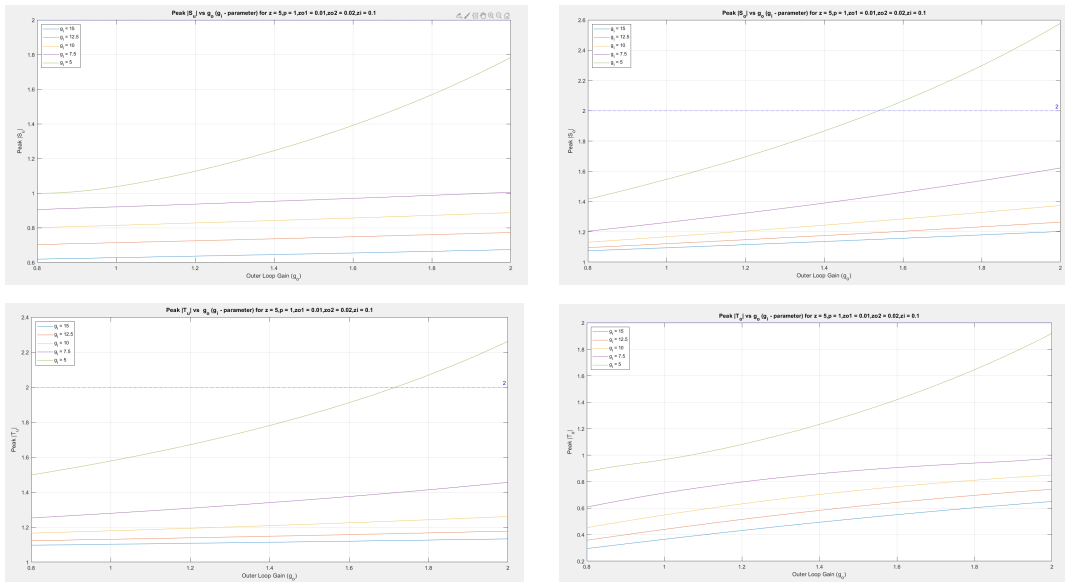
Figure 2.4.11: Peak $|S_u|$, $|S_o|$, $|T_u|$, $|T_o|$ vs g_o (g_i-parameter)

Varying Outer Loop Gain g_i with g_o fixed at $g_o = 2$

Inner Loop Sensitivity and Complementary Sensitivity
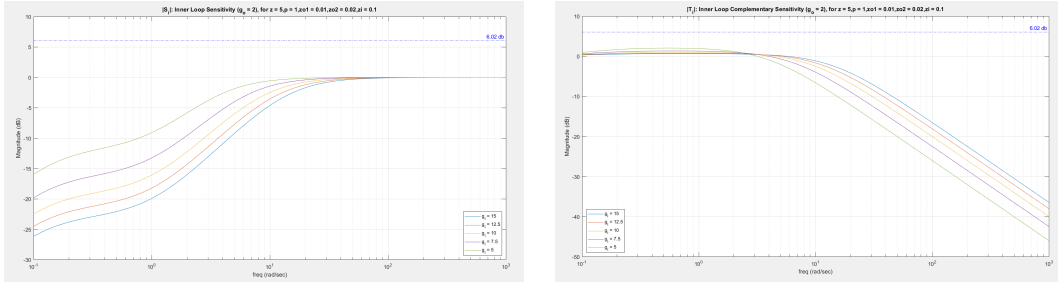
Figure 2.4.12: $|S_i|$ and $|T_i|$: Inner Loop Sensitivity and Complementary Sensitivity $(g_o = 2)$

Loop Sensitivity and Complimentary sensitivity at Output and at Controls
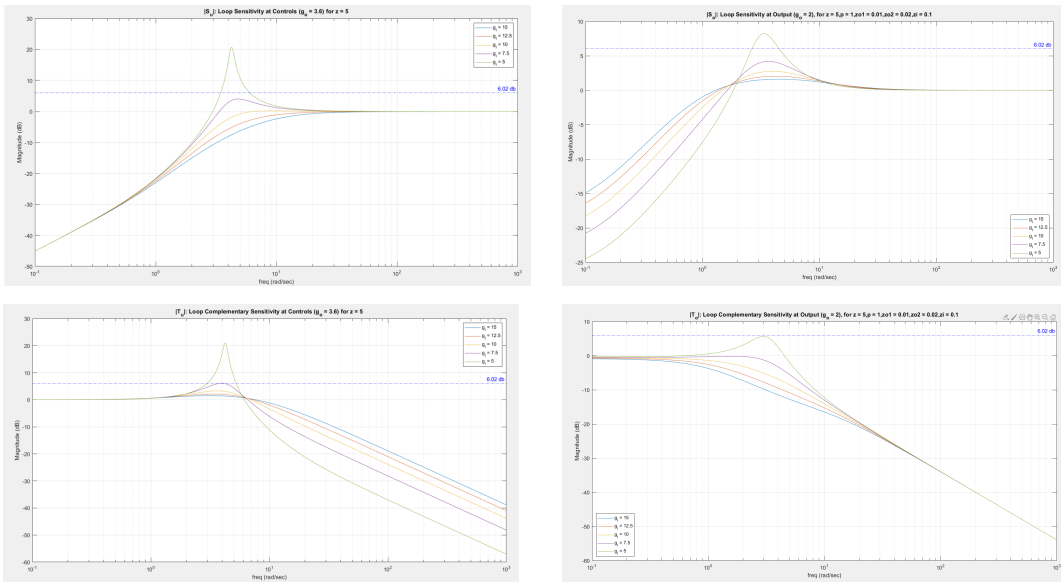
Figure 2.4.13: $|S_u|$, $|S_o|$, $|T_u|$, $|T_o|$: Loop Sensitivity at Output and at Controls $(g_o = 2)$

- $|S_i| < 1 \rightarrow |S_o| > |S_u| \ \forall \ \omega$.

71

$\left|\frac{T_u}{T_o}\right|$ **vs** g_i **(**g_o **- parameter)**
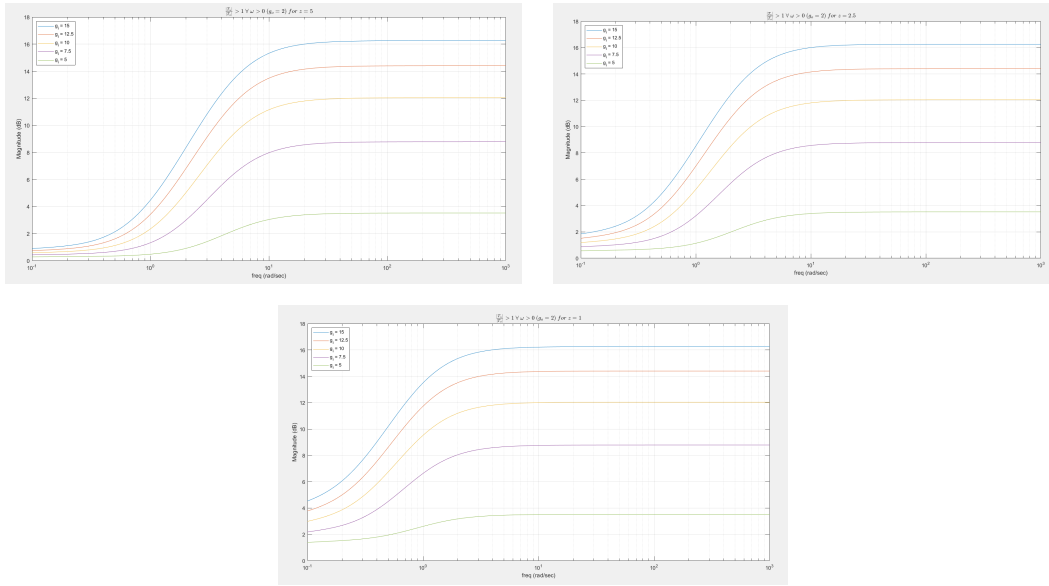
Figure 2.4.14: $\left|\frac{T_u}{T_o}\right| > 1 \ \forall \ \omega \ > 0 \ (g_o = 2)$

Peak $|S_o|$ **,** $|S_u|$, $|T_o|$, $|T_u|$
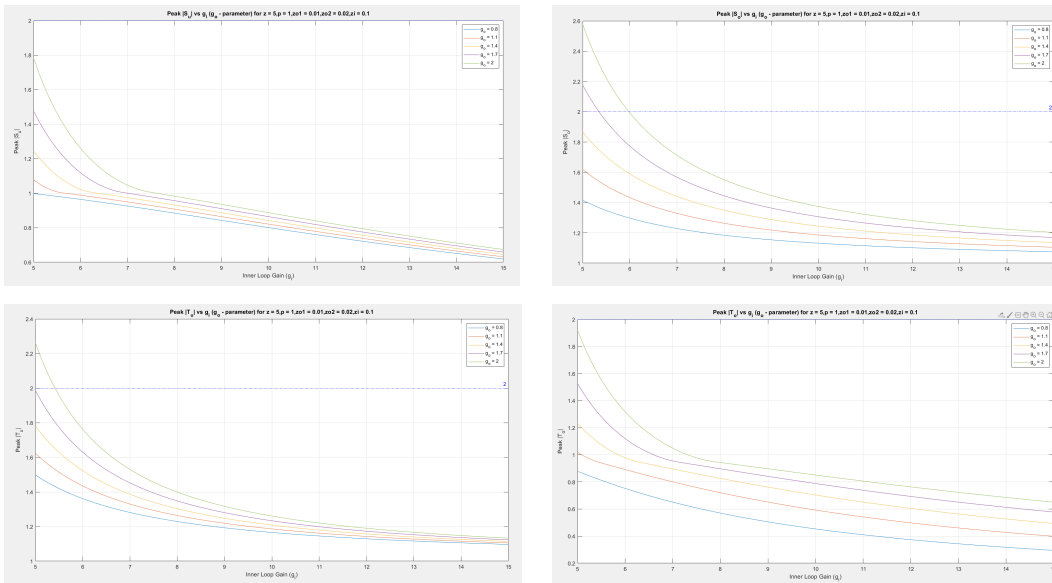
Figure 2.4.15: Peak $|S_o|$, $|S_u|$, $|T_o|$, $|T_u|$ vs g_o (g_i-parameter)

- Peak $|S_o|$, $|S_u|$, $|T_o|$, $|T_u|$ increases with increasing g_o or decreasing g_i.

Varying Outer Loop Gain g_o with Inner Loop Gain g_i Fixed at $g_i = 5$

Time Domain Analysis: Impulse S_o, S_u, T_o, T_u
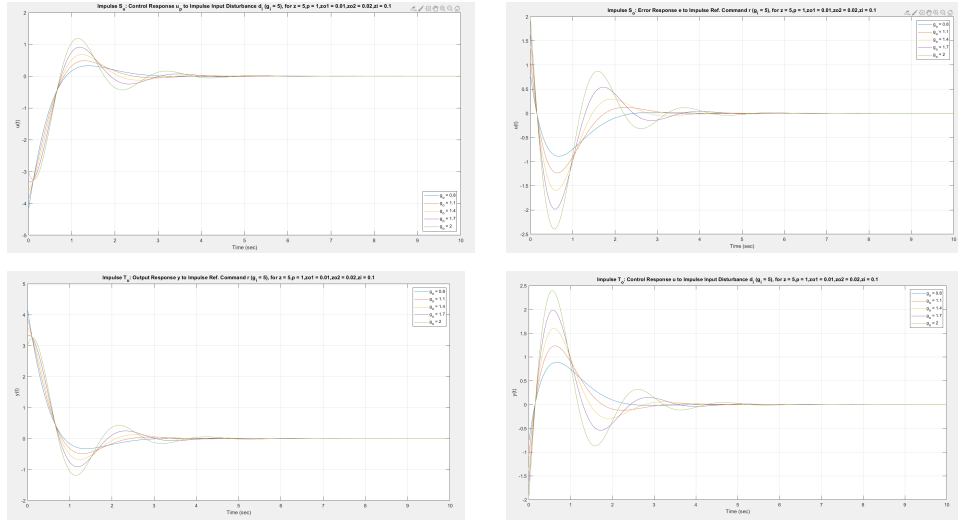
Figure 2.4.16: Impulse S_o, S_u, T_o, T_u ($g_i = 5$)

Time Domain Analysis: Step S_o, S_u, T_o, T_u
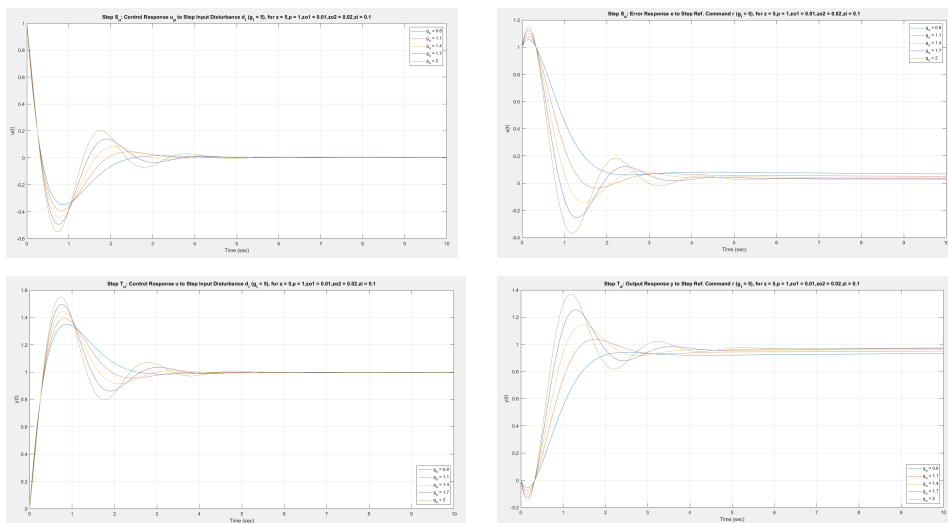
Figure 2.4.17: Step S_o, S_u, T_o, T_u ($g_i = 5$)

Varying Outer Loop Gain g_o with Inner Loop Gain g_i Fixed at $g_o = 2$

Time Domain Analysis: Impulse S_o, S_u, T_o, T_u
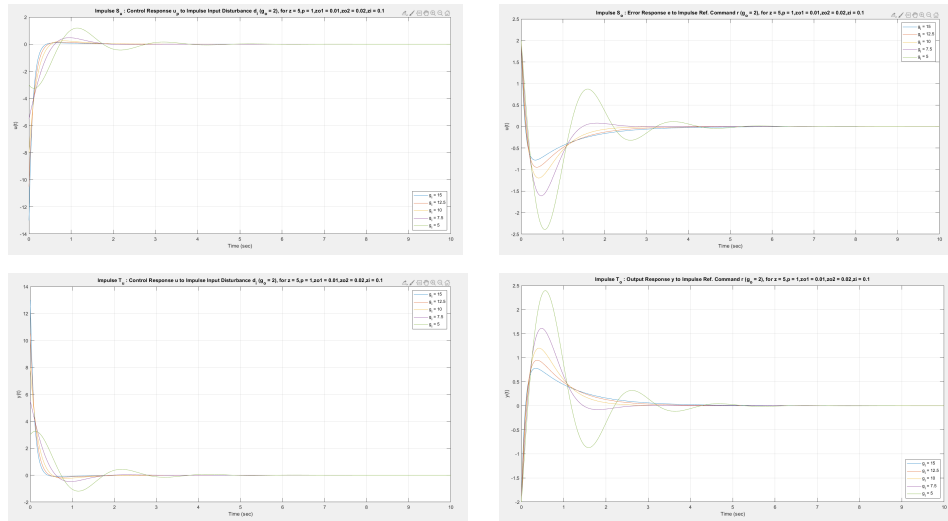
Figure 2.4.18: Impulse S_o, S_u, T_o, T_u $(g_o = 2)$

Time Domain Analysis: Step S_o, S_u, T_o, T_u
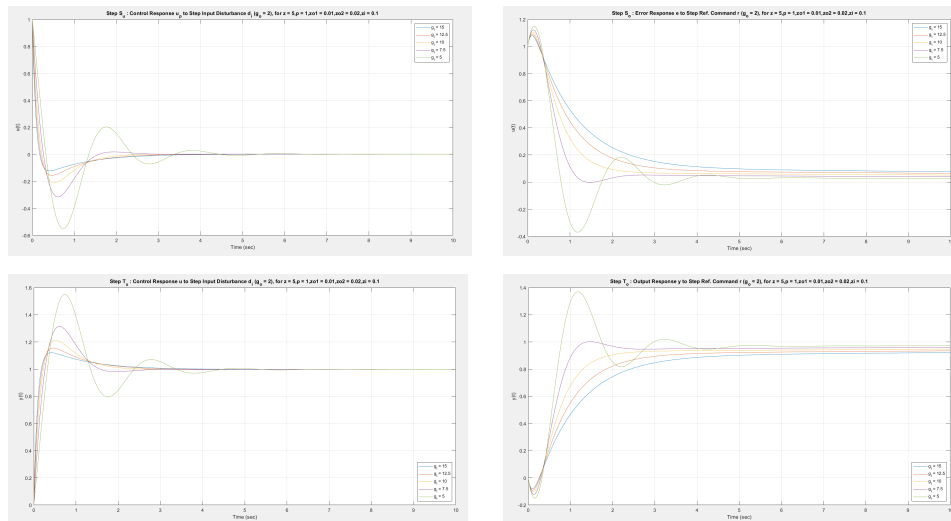
Figure 2.4.19: Step S_o, S_u, T_o, T_u $(g_o = 2)$

- Initial transient undershoot increases with increasing g_o.

74

2.4.6 Summary and Conclusions

	$g_o \uparrow$	$g_i \downarrow$	$z \downarrow$
$|L_u|$	$\uparrow$	$\downarrow$	$\downarrow$
$\angle L_u$	$\downarrow$	$\downarrow$	$\uparrow$
$|L_o|$	$\uparrow$	$\uparrow$	$\downarrow$
$\angle L_o$	$-$	$\downarrow$	$\downarrow$
$\downarrow GM$	$\uparrow$	$\uparrow$	$-$
$\omega_{\downarrow GM}$	$\uparrow$	$\uparrow$	$\downarrow$
$\uparrow GM$	$\downarrow$	$\downarrow$	$-$
$\omega_{\uparrow GM}$	$-$	$\downarrow$	$\downarrow$
PM_u	$\downarrow$	$\downarrow$	$\uparrow$
PM_o	$\downarrow$	$\downarrow$	$\downarrow$
DM_u	$\uparrow$	$\uparrow$	$\uparrow$
DM_o	$\downarrow$	$\downarrow$	$\uparrow$
ω_{gu}	$\downarrow$	$\downarrow$	$\uparrow$
ω_{go}	$\uparrow$	$\uparrow$	$\downarrow$
$PeakS_u$	$\uparrow$	$\uparrow$	$\downarrow$
$PeakS_o$	$\uparrow$	$\uparrow$	$\downarrow$
$PeakT_u$	$\uparrow$	$\uparrow$	$\downarrow$
$PeakT_o$	$\uparrow$	$\uparrow$	$\downarrow$
$|S_i|$	$-$	$\uparrow$	$-$
$|T_i|$	$-$	$\downarrow$	$-$

Table 2.4.2: Summary: Example 4

2.5   Example 5 Set Up and Assumptions

Block Diagram. The relevant block diagram can be found in Figure 2.5.1.
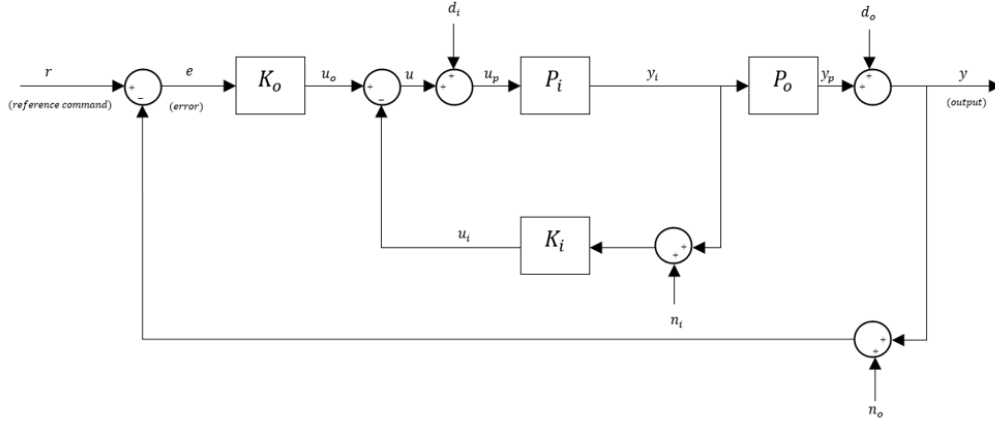
Figure 2.5.1: Block Diagram for Simple System

Nominal Parameters:

Parameter	Nominal Value	Description
p	1	Pole of P_i
z	5	Zero of P_o
g_o	(0.8-2)	Outer Loop Controller K_o Gain (will be varied)
g_i	(5-15)	Inner Loop Controller K_i Gain (will be varied)
z_i	(2)	Inner Loop Controller K_i Zero
z_{o1}	(0.01)	Outer Loop Controller K_o Zero
z_{o2}	(0.02)	Outer Loop Controller K_o Zero

Table 2.5.1: Nominal Parameter Values

This example introduces two zeros in the outer loop and one zero in the inner loop. In agreement with the parameters mentioned above the following trade studies are very insightful with respect to the cart inverted pendulum ideas.

Nominal System:

$$P_i = \frac{1}{(s-p)(s+p)}, \quad K_i = g_i(s+z_i), \quad g_i > 0, z_i > 0 \tag{2.5.1}$$

$$P_o = \frac{z-s}{s^2}, \quad K_o = \frac{g_o(s+z_{o1})(s+z_{o2}))}{s}, \quad g_o > 0, \quad z_{o1} > 0, \quad z_{o2} > 0 \tag{2.5.2}$$

Inner Loop Transfer Function:

$$L_i = P_i K_i = \frac{g_i(s+z_i)}{(s-p)(s+p)} \tag{2.5.3}$$

Inner Loop Sensitivity:

$$S_i = \frac{1}{1+L_i} = \frac{(s-p)(s+p)}{s^2 + g_i s + g_i z_i - p^2} \tag{2.5.4}$$

Inner Loop Complementary Sensitivity:

$$T_i = \frac{L_i}{1+L_i} = \frac{g_i(s+z_i)}{s^2 + g_i s + g_i z_i - p^2} \tag{2.5.5}$$

Loop Transfer Function at Output:

$$L_o = P_o \left(\frac{P_i}{1 + P_i K_i} \right) K_o = \frac{g_o(z-s)(s+z_{o1})(s+z_{o2})}{s^3(s^2 + g_i s + g_i z_i - p^2)} \tag{2.5.6}$$

Loop Transfer Function at Controls:

$$L_u = K_i P_i + K_o P_o P_i = \frac{g_i s^4 + (g_i z_i - g_o)s^3 + c1 s^2 + c2 s + g_o z z_{o1} z_{o2}}{s^3(s-p)(s+p)} \tag{2.5.7}$$

Sensitivity at Output:

$$S_o = S_e = \frac{1}{1 + L_o} = \frac{s^3(s^2 + g_i s + g_i z_i - p^2)}{s^5 + g_i s^4 + (g_i z_i - g_o)s^3 + c1s^2 + c2s + g_o z z_{o1} z_{o2}} \qquad (2.5.8)$$

Sensitivity at Controls:

$$S_u = \frac{1}{1 + L_u} = \frac{s^3(s - p)(s + p)}{s^5 + g_i s^4 + (g_i z_i - g_o)s^3 + c1s^2 + c2s + g_o z z_{o1} z_{o2}} \qquad (2.5.9)$$

Inner Outer Loop Identity:

$$S_i = \frac{S_u}{S_o} \qquad (2.5.10)$$

Complementary Sensitivity at Output:

$$T_o = T_e == \frac{L_o}{1 + L_o} = \frac{g_o(z - s)(s + z_{o1})(s + z_{o2})}{s^5 + g_i s^4 + (g_i z_i - g_o)s^3 + c1s^2 + c2s + g_o z z_{o1} z_{o2}} \qquad (2.5.11)$$

Complementary Sensitivity at Controls:

$$T_u = \frac{L_u}{1 + L_u} = \frac{g_i s^4 + (g_i z_i - g_o)s^3 + c1s^2 + c2s + g_o z z_{o1} z_{o2}}{s^5 + g_i s^4 + (g_i z_i - g_o)s^3 + c1s^2 + c2s + g_o z z_{o1} z_{o2}} \qquad (2.5.12)$$

Complementary Sensitivity Identities:

$$T_u - T_o = T_i S_u \qquad (2.5.13)$$

$$T_u - T_i = T_o S_i \qquad (2.5.14)$$

$$c1 = (g_o z - g_o z_{o1} - g_o z_{o2}); c2 = (g_o z - g_o z_{o1} - g_o z_{o2}); \qquad (2.5.15)$$

$$S_o = T_{re}; \qquad T_o = T_{ry}; \qquad S_u = T_{d_i u_p}; \qquad T_u = T_{d_i u} \qquad (2.5.16)$$

2.5.1 Fundamental Sensitivity Relationships

Similar to example 1. The tradeoff between the properties of the loop at controls and the loop at output is given by the inner-outer loop identity:

- **Identity:** $S_u = S_o S_i$

From this identity it is observed that:

- **lemma:** $\|S_i\|_{\mathcal{H}^\infty} \geq 1, \|S_o\|_{\mathcal{H}^\infty} \leq \|S_u\|_{\mathcal{H}^\infty} : \|S_i\|_{\mathcal{H}^\infty} \geq 1, \|T_u\|_{\mathcal{H}^\infty} \leq \|T_o\|_{\mathcal{H}^\infty}$
  (since $S_o = 1 - T_o, S_u = 1 - T_u$).

- **lemma:** $\|S_i\|_{\mathcal{H}^\infty} < 1, \|S_u\|_{\mathcal{H}^\infty} < \|S_o\|_{\mathcal{H}^\infty}: \|S_i\|_{\mathcal{H}^\infty} < 1, \|T_o\|_{\mathcal{H}^\infty} < \|T_u\|_{\mathcal{H}^\infty}$
  (since $S_o = 1 - T_o, S_u = 1 - T_u$).

1. g_i - stability is very complicated to compute analytically, since the system has more parameters. So it is computed numerically. The code for computing it numerically is given in the appendix.

2. **Corollary 1:** $p - $ **Non-Aggressive Inner Loop:** $\|S_i\|_{\mathcal{H}^\infty} \geq 1 \iff g_i - $ $stability < g_i \leq 2p \iff \|S_o\|_{\mathcal{H}^\infty} \leq \|S_u\|_{\mathcal{H}^\infty} ; z_i \approx 0:$

3. **Corollary 2:** $p - $ **Aggressive Inner Loop:** $\|S_i\|_{\mathcal{H}^\infty} < 1 \iff \max\{g_i - $ $stability, 2p\} < g_i \iff \|S_o\|_{\mathcal{H}^\infty} > \|S_u\|_{\mathcal{H}^\infty} ; z_i \approx 0.$

4. **Corollary 3:** $g - $ **Aggressive Inner Loop:** $\|S_i\|_{\mathcal{H}^\infty} < 1 \iff \max\{g_i - $ $stability, 2g_o\} < g_i \iff \|T_o\|_{\mathcal{H}^\infty} < \|T_u\|_{\mathcal{H}^\infty} ; z_i \approx 0.$

5. **Comment: Peak Sensitivities:** $\|S\|_{\mathcal{H}^\infty} , \|T\|_{\mathcal{H}^\infty}$ increase with increasing g_o and decreasing g_i.

79

Similar to example 1, we get the stability of the inner loop as:

Figure 2.5.2: Nominal System: Open Loop Transfer Function

1. If $g_i \approx stability condition$; the system has poles near the imaginary axis which leads to higher peak sensitivities t for the loop at output and for the loop at controls.
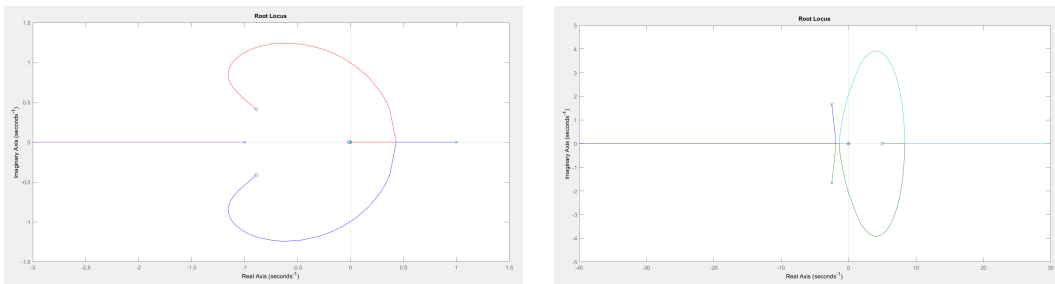
Figure 2.5.3: Nominal System: Root locus of Loop Transfer Function at Controls (left) and at Output (right)

The above plots explain that the $\uparrow GM$ at controls is infinite, $\downarrow GM$ at controls is finite. On the other hand, $\downarrow GM$ at output is zero, $\uparrow GM$ is finite because of the right hand plane zero. Thus, this cause the system to be bandwidth limited.

80

Varying Outer Loop Gain g_o with g_i fixed at $g_i = 5$

Loop at Input:
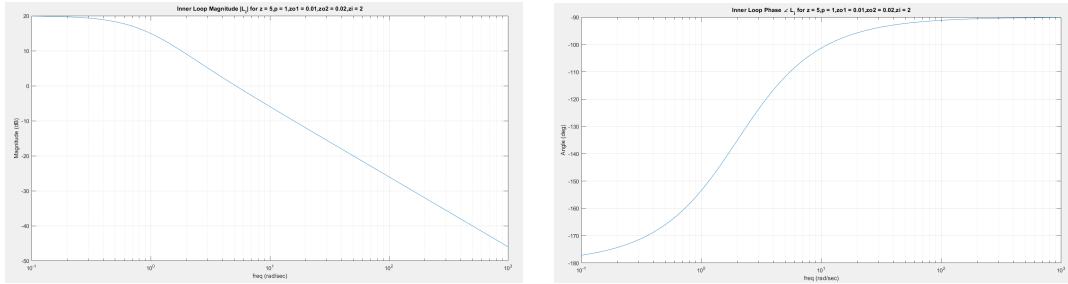
Figure 2.5.4: Inner Loop Magnitude $|L_i|$ and Phase $\angle L_i$

Loop at Output and Loop at Controls:
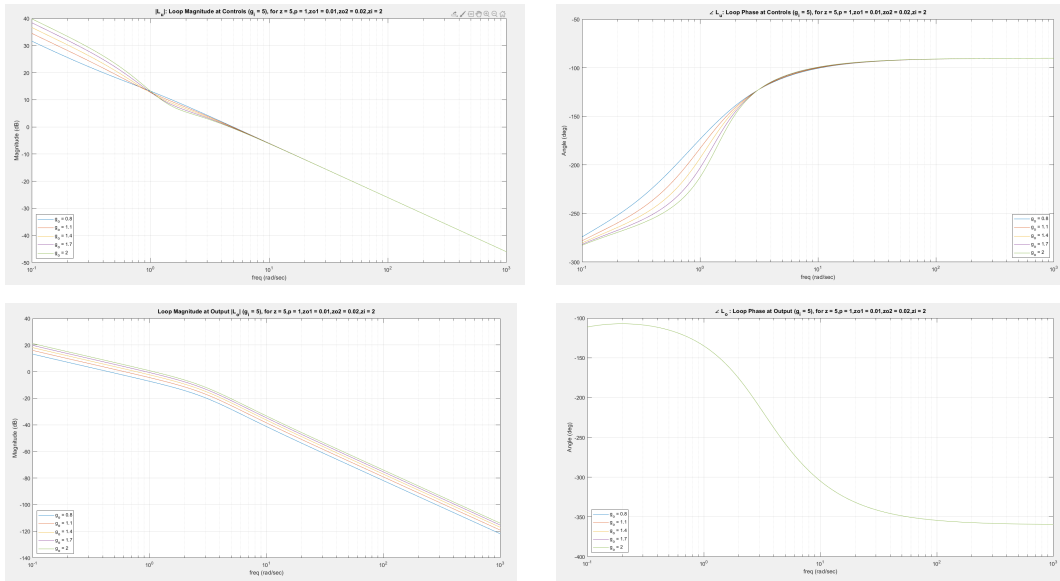
Figure 2.5.5: L_u and L_o: Loop Magnitude and Loop Phase ($g_i = 5$) for $z = 5$

- $|L_o|$ increases with increasing g_o, ω_{go} decreases with increasing g_o, $\angle L_o$ is independent of g_o, PM is independent of g_o. PM decreases with increasing g_o

Varying Outer Loop Gain g_i with g_o fixed at $g_o = 2$

Loop at Input:
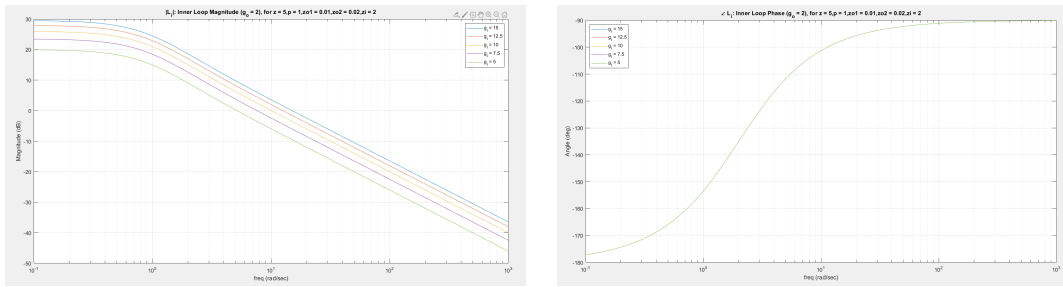
Figure 2.5.6: Inner Loop Magnitude $|L_i|$ and Phase $\angle L_i$

Loop at Output and Loop at Controls:
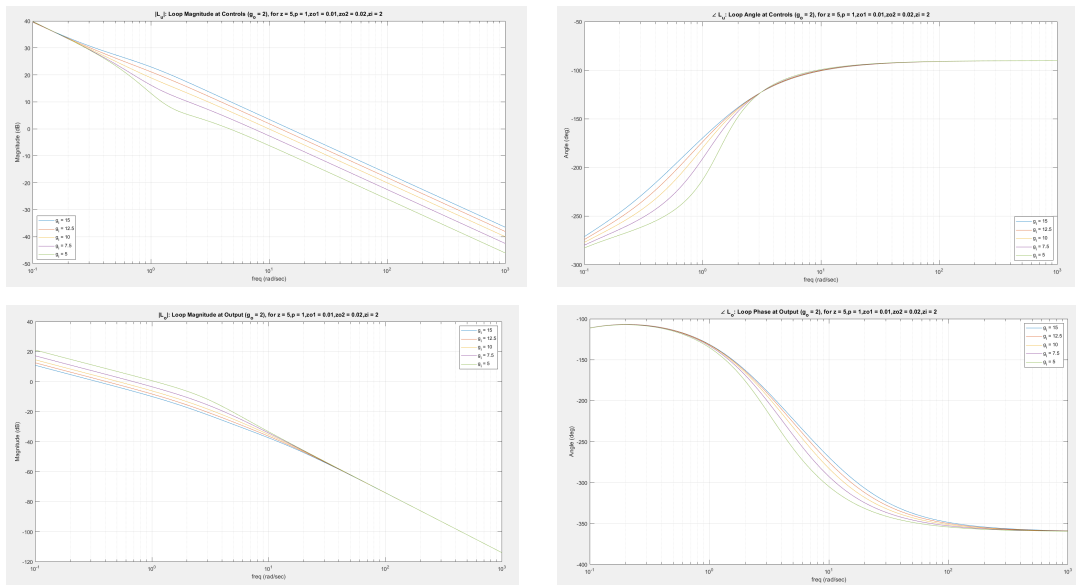
Figure 2.5.7: L_u and L_o: Loop Magnitude and Loop Phase ($g_o = 2$)

- $|L_o|$ increases with increasing g_o, ω_{go} decreases with increasing g_o, $\angle L_o$ is independent of g_o, PM is independent of g_o.

Varying Outer Loop Gain g_o with Inner Loop Gain g_i Fixed at $g_i = 5$

Inner Loop Sensitivity and Complementary Sensitivity
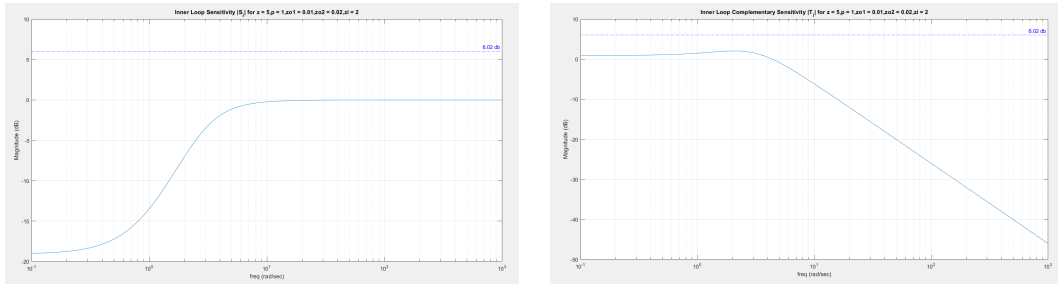
Figure 2.5.8: Inner Loop Sensitivity and Complementary Sensitivity $|S_i|$ and $|T_i|$

Loop Sensitivity at Output and at Controls
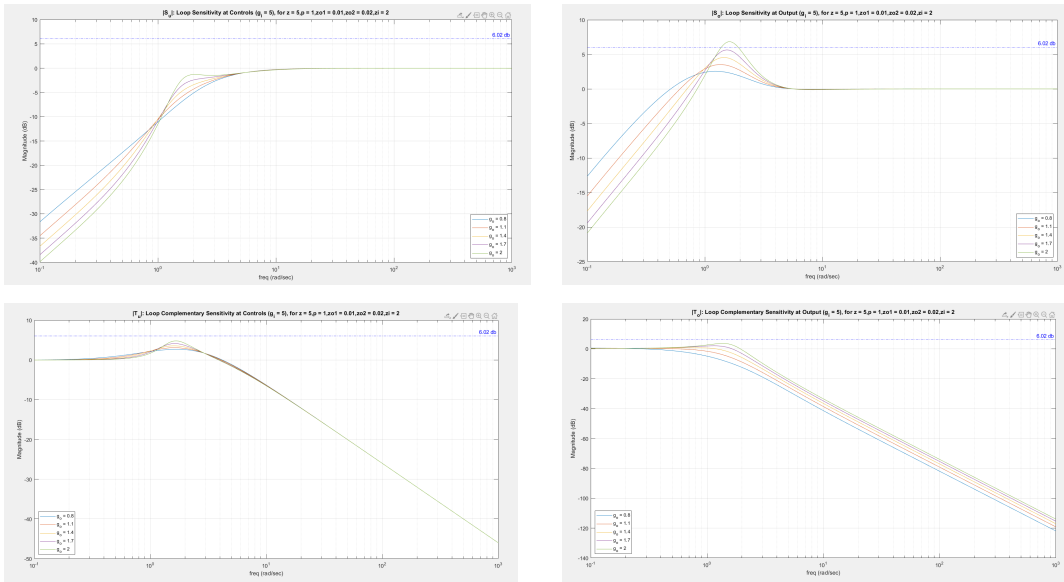
Figure 2.5.9: $|S_o|$ and $|S_u|$: Loop Sensitivity at Output and at Controls ($g_i = 5$)

- Peak $|S_o|$, $|S_u|$, $|T_o|$, $|T_u|$ increases with increasing g_o or decreasing g_i.

83

Inner Outer Loop Complementary Sensitivity Ratio $\left|\frac{T_u}{T_o}\right|$ vs g_i (g_o - parameter)
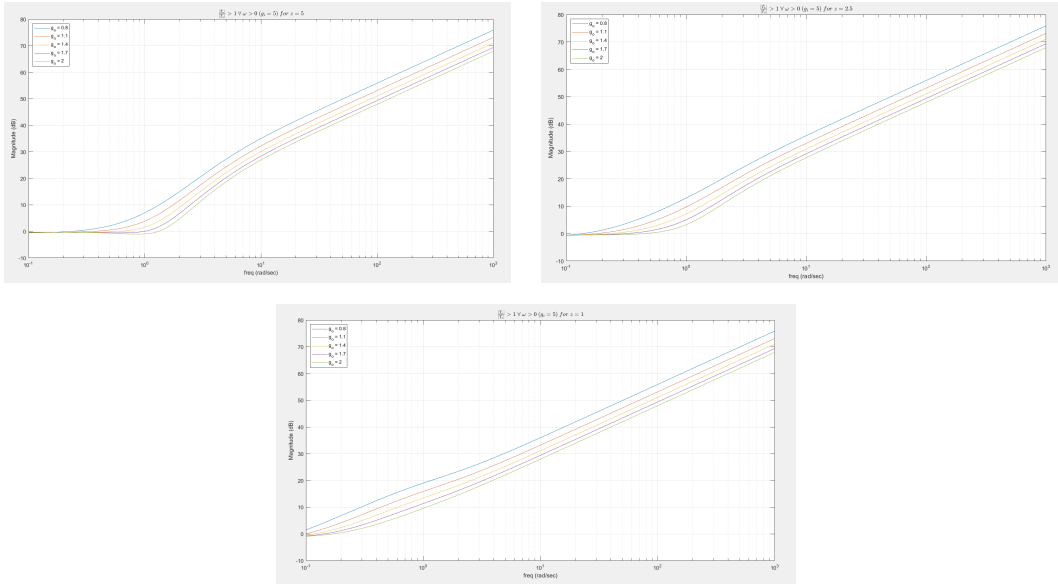
Figure 2.5.10: $\left|\frac{T_u}{T_o}\right| > 1 \;\; \forall \;\; \omega \;\; > 0 \; (g_i = 5)$

Peak Sensitivity and Complimentary Sensitivity
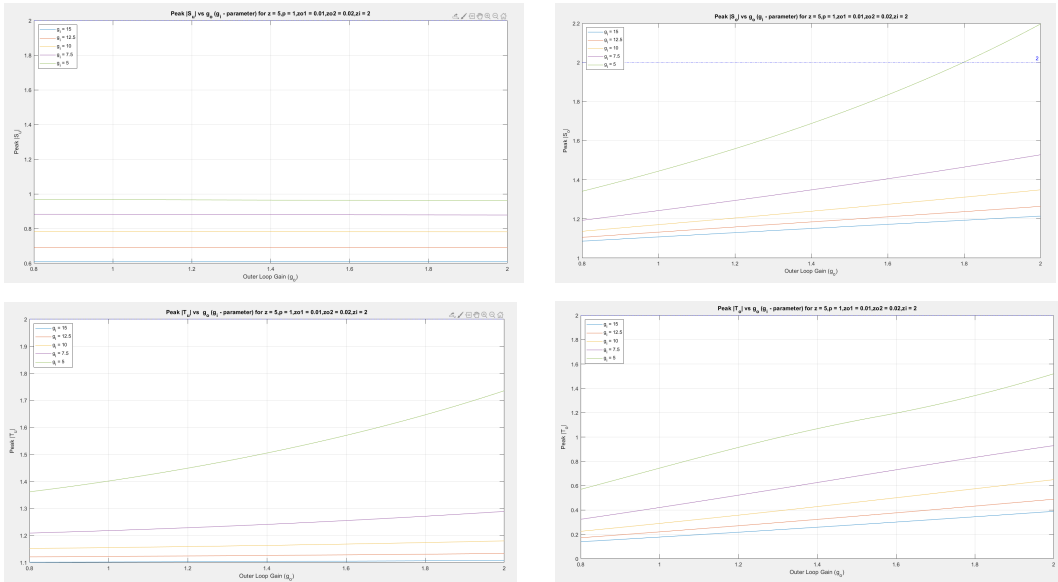
Figure 2.5.11: Peak $|S_u|$, $|S_o|$, $|T_u|$, $|T_o|$ vs g_o (g_i-parameter)

Varying Outer Loop Gain g_i with g_o fixed at $g_o = 2$
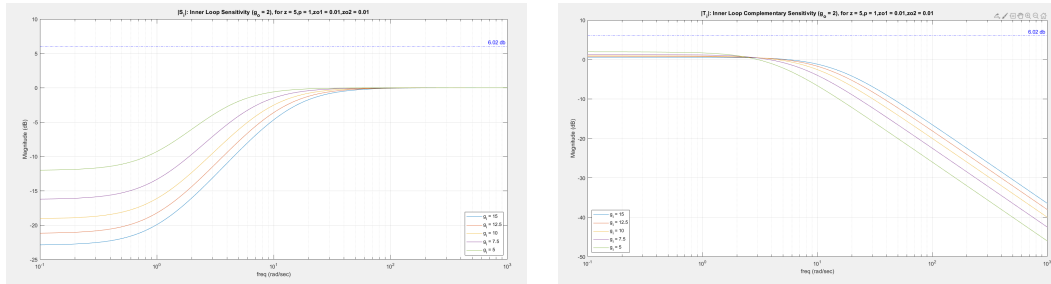
Inner Loop Sensitivity and Complementary Sensitivity
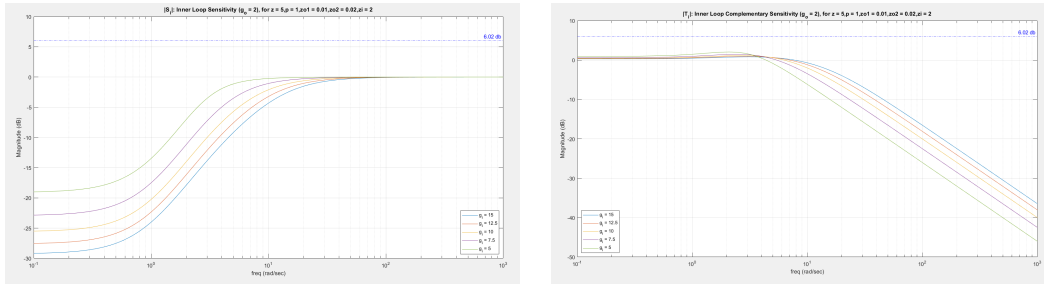
Figure 2.5.12: $|S_i|$ and $|T_i|$: Inner Loop Sensitivity and Complementary Sensitivity

$(g_o = 2)$

Loop Sensitivity and Complimentary sensitivity at Output and at Controls
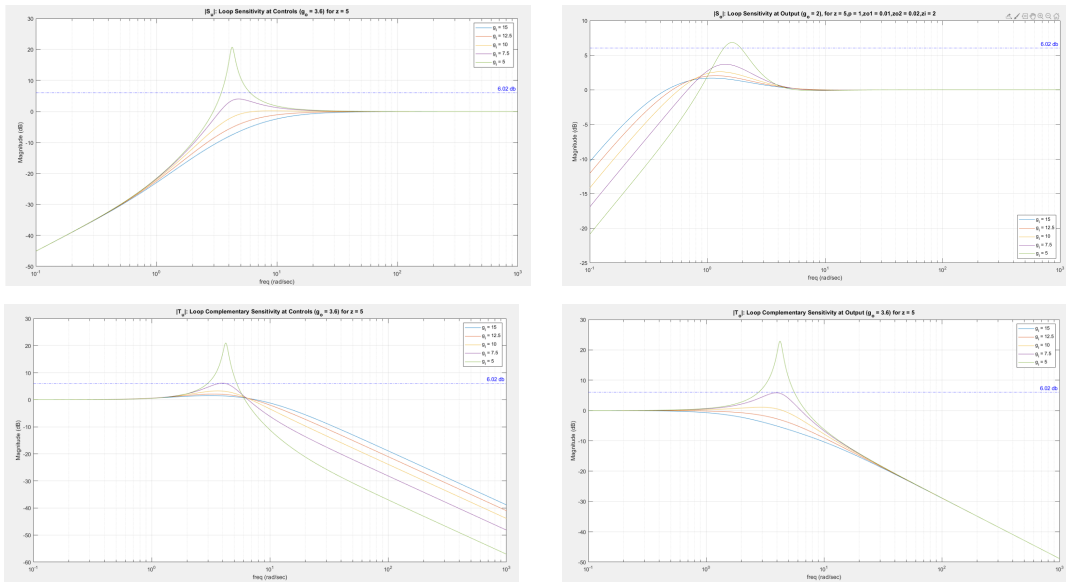
Figure 2.5.13: $|S_u|$, $|S_o|$, $|T_u|$, $|T_o|$: Loop Sensitivity at Output and at Controls

$(g_o = 2)$

- Peak $|S_o|$, $|S_u|$, $|T_o|$, $|T_u|$ increases with increasing g_o or decreasing g_i.

85

$|\frac{T_u}{T_o}|$ **vs** g_i **(**g_o **- parameter)**
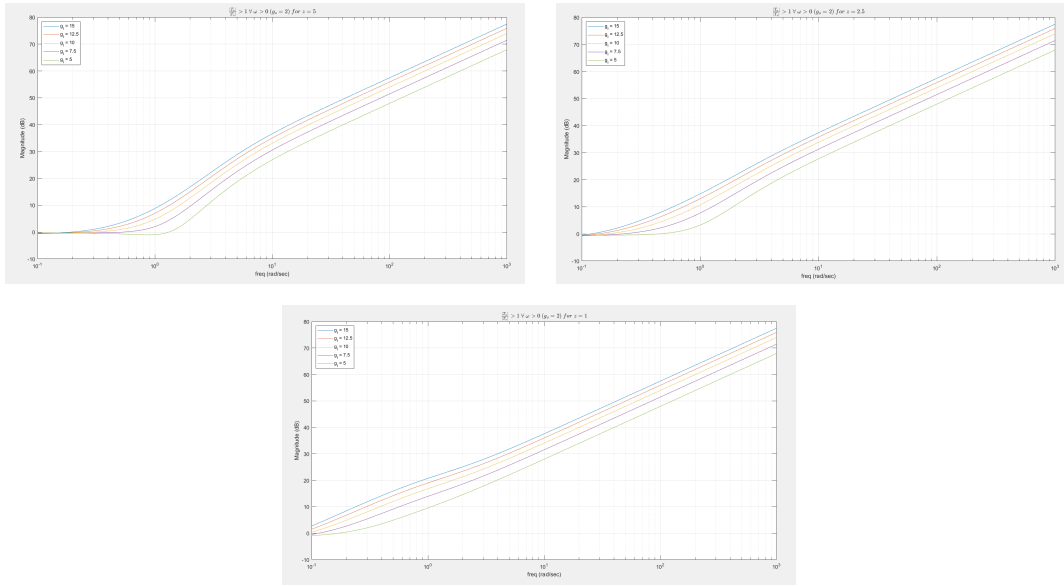
Figure 2.5.14: $|\frac{T_u}{T_o}| > 1 \ \forall \ \omega \ > 0 \ (g_o = 2)$

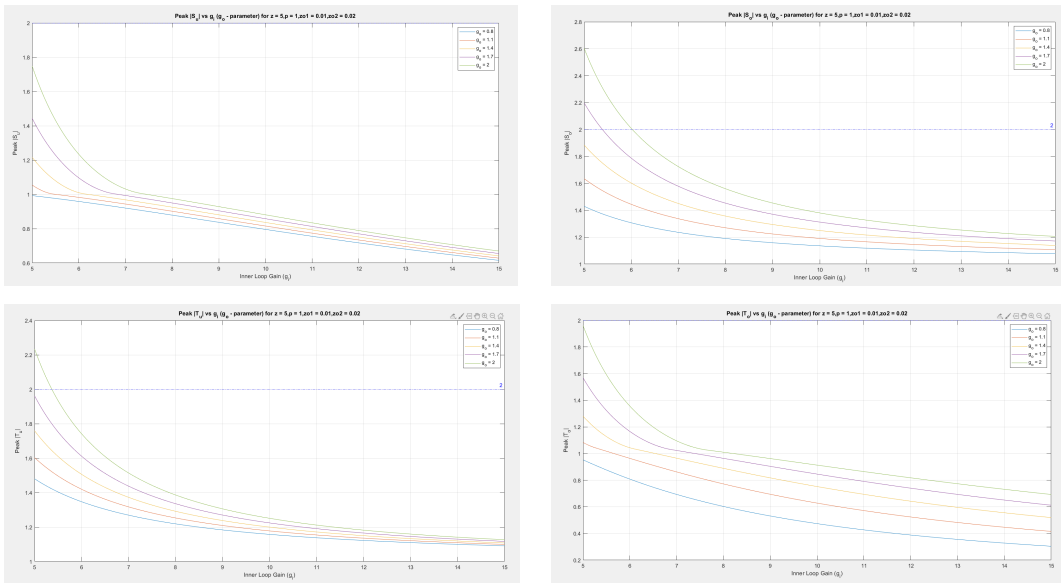**Peak** $|S_o|$ **,** $|S_u|$**,** $|T_o|$**,** $|T_u|$
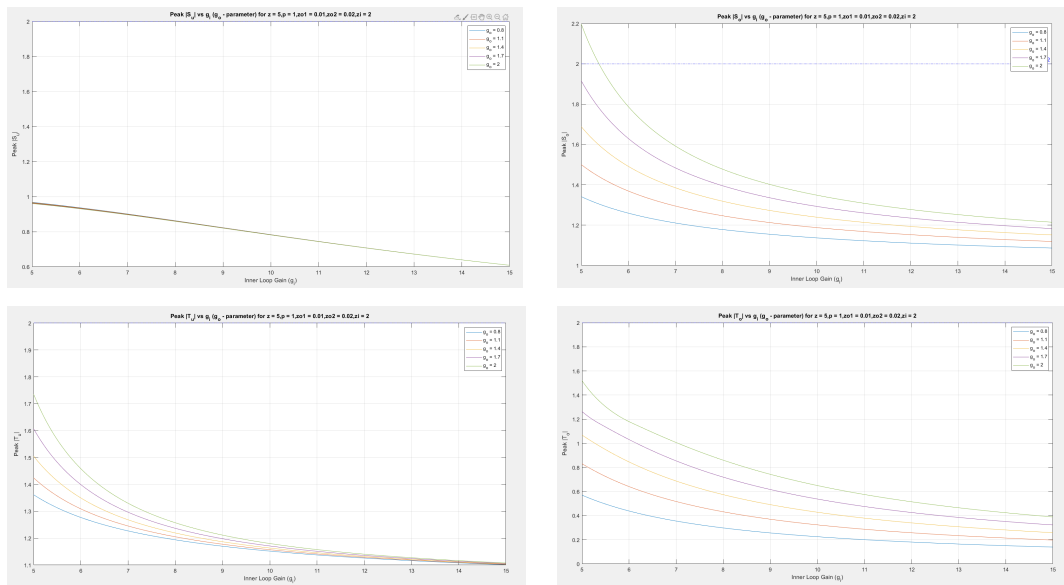
Figure 2.5.15: Peak $|S_o|$, $|S_u|$, $|T_o|$, $|T_u|$ vs g_o (g_i-parameter)

- Peak $|S_o|$, $|S_u|$, $|T_o|$, $|T_u|$ increases with increasing g_o or decreasing g_i.

86

Varying Outer Loop Gain g_o with Inner Loop Gain g_i Fixed at $g_i = 5$

Time Domain Analysis: Impulse S_o, S_u, T_o, T_u
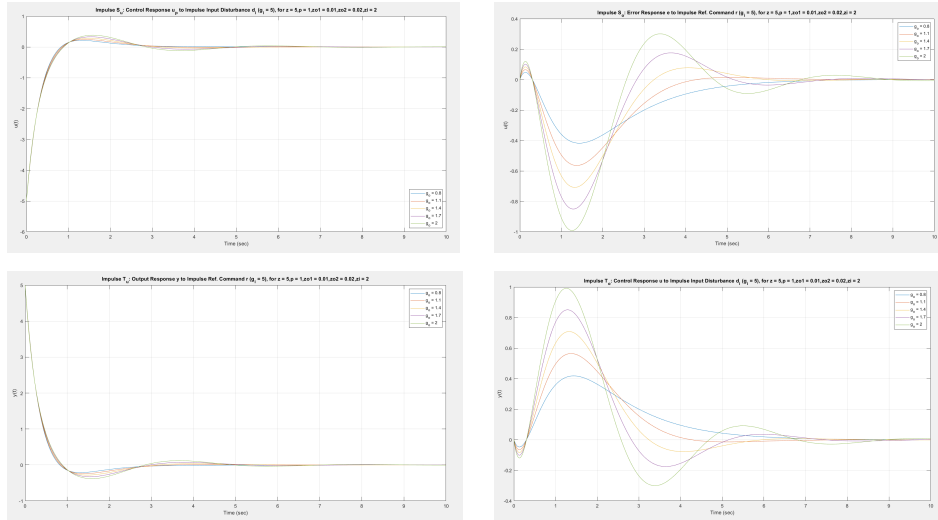
Figure 2.5.16: Impulse S_o, S_u, T_o, T_u $(g_i = 5)$

Time Domain Analysis: Step S_o, S_u, T_o, T_u
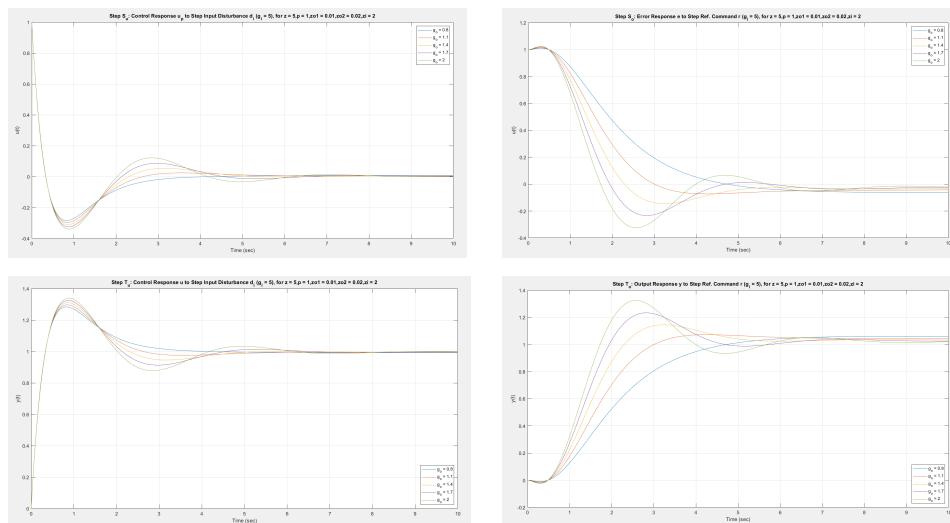
Figure 2.5.17: Step S_o, S_u, T_o, T_u $(g_i = 5)$

Varying Outer Loop Gain g_o with Inner Loop Gain g_i Fixed at $g_o = 2$

Time Domain Analysis: Impulse S_o, S_u, T_o, T_u
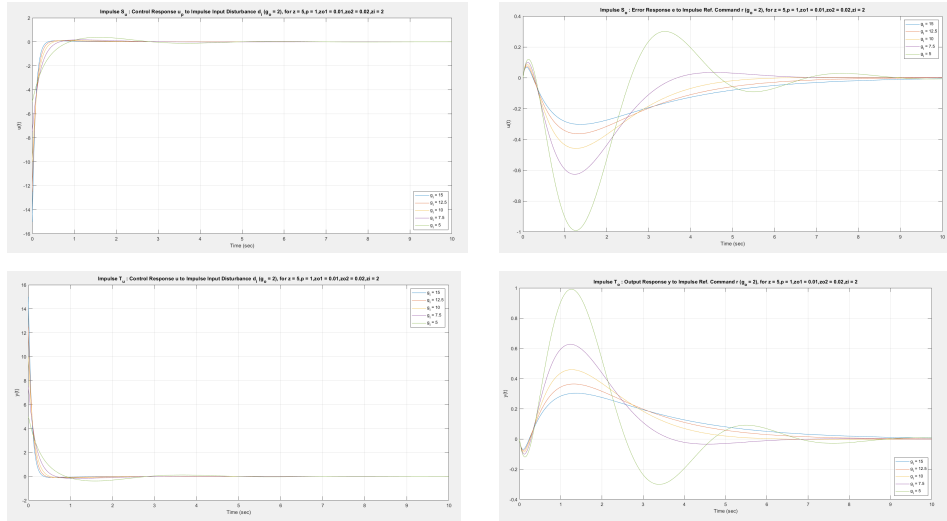
Figure 2.5.18: Impulse S_o, S_u, T_o, T_u ($g_o = 2$)

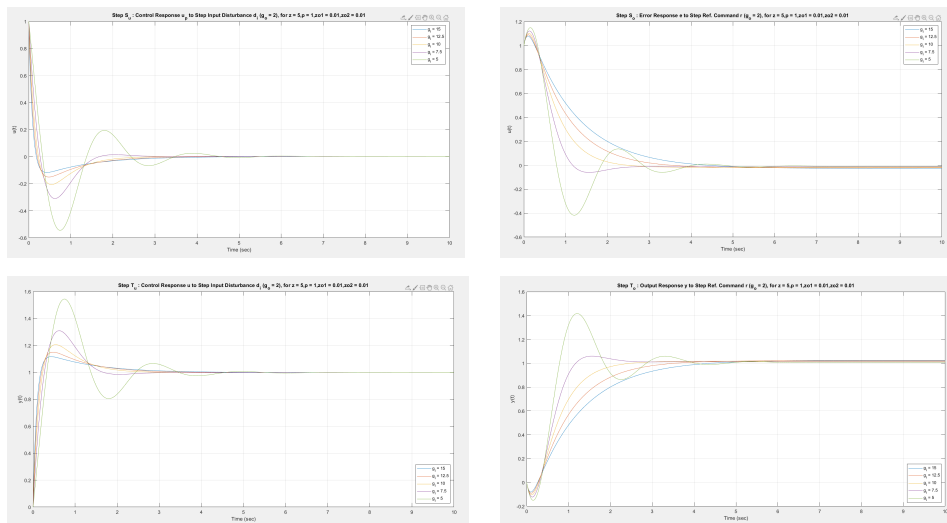**Time Domain Analysis: Step S_o, S_u, T_o, T_u**
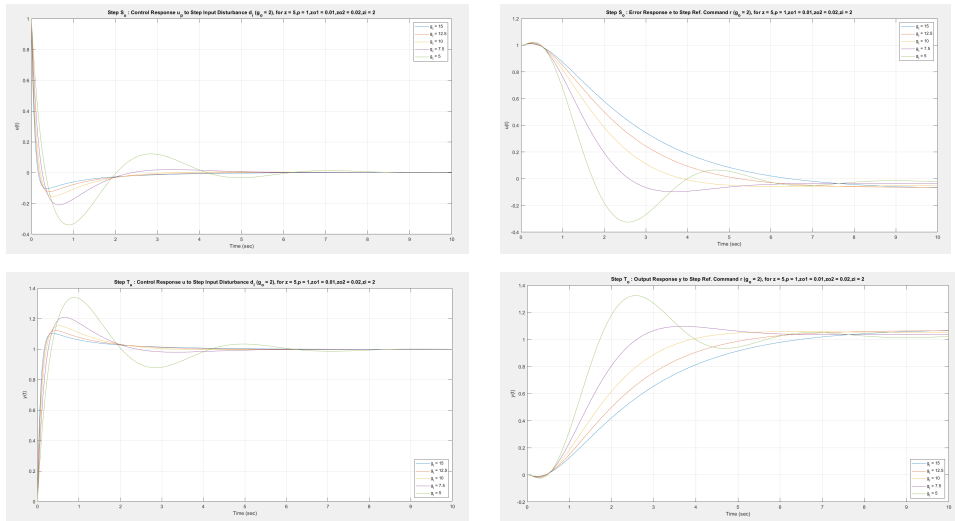
Figure 2.5.19: Step S_o, S_u, T_o, T_u ($g_o = 2$)

- Initial transient undershoot increases with increasing g_o.

2.5.6   Summary and Conclusions

	$g_o \uparrow$	$g_i \downarrow$	$z \downarrow$
$\lvert L_u \rvert$	$\uparrow$	$\downarrow$	$\downarrow$
$\angle L_u$	$\downarrow$	$\downarrow$	$\uparrow$
$\lvert L_o \rvert$	$\uparrow$	$\uparrow$	$\downarrow$
$\angle L_o$	$-$	$\downarrow$	$\downarrow$
$\downarrow GM$	$\uparrow$	$\uparrow$	$-$
$\omega_{\downarrow GM}$	$\uparrow$	$\uparrow$	$\downarrow$
$\uparrow GM$	$\downarrow$	$\downarrow$	$-$
$\omega_{\uparrow GM}$	$-$	$\downarrow$	$\downarrow$
PM_u	$\downarrow$	$\downarrow$	$\uparrow$
PM_o	$\downarrow$	$\downarrow$	$\downarrow$
DM_u	$\uparrow$	$\uparrow$	$\uparrow$
DM_o	$\downarrow$	$\downarrow$	$\uparrow$
ω_{gu}	$\downarrow$	$\downarrow$	$\uparrow$
ω_{go}	$\uparrow$	$\uparrow$	$\downarrow$
$PeakS_u$	$\uparrow$	$\uparrow$	$\downarrow$
$PeakS_o$	$\uparrow$	$\uparrow$	$\downarrow$
$PeakT_u$	$\uparrow$	$\uparrow$	$\downarrow$
$PeakT_o$	$\uparrow$	$\uparrow$	$\downarrow$
$\lvert S_i \rvert$	$-$	$\uparrow$	$-$
$\lvert T_i \rvert$	$-$	$\downarrow$	$-$

Table 2.5.2: Summary: Example 5

2.6 Example 6 Set Up and Assumptions

Block Diagram. The relevant block diagram can be found in Figure 2.6.1.
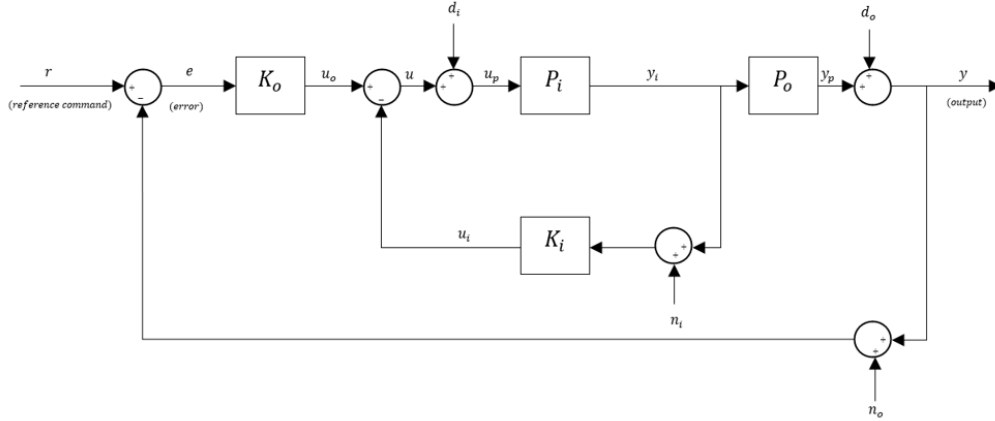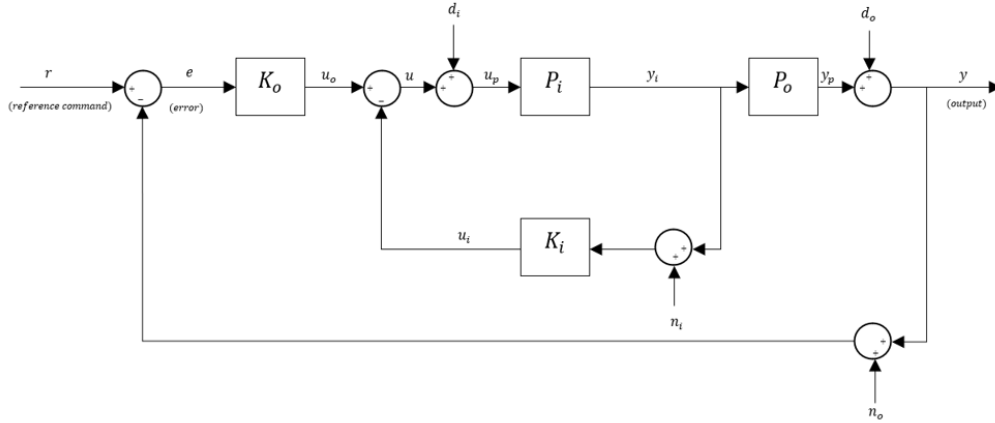

Figure 2.6.1: Block Diagram for Simple System

Nominal Parameters:

Parameter	Nominal Value	Description
p	1	Pole of P_i
z	5	Zero of P_o
g_o	(0.8 - 2)	Outer Loop Controller K_o Gain (will be varied)
g_i	(5-15)	Inner Loop Controller K_i Gain (will be varied)
z_{i1}	(1)	Inner Loop Controller K_i Zero (will be varied)
z_{i2}	(1.1)	Inner Loop Controller K_i Zero (will be varied)
z_{o1}	(0.01)	Outer Loop Controller K_o Zero (will be varied)
z_{o2}	(0.02)	Outer Loop Controller K_o Zero (will be varied)

Table 2.6.1: Nominal Parameter Values

This example introduces two zeros in the outer loop and two zeros in the inner loop.

90

Nominal System:

$$P_i = \frac{1}{(s-p)(s+p)}, \quad K_i = \frac{g_i(s+z_{i1})(s+z_{i2}))}{s}, \quad g_i > 0, z_{i1} > 0. z_{i2} > 0 \quad (2.6.1)$$

$$P_o = \frac{z-s}{s^2}, \quad K_o = \frac{g_o(s+z_{o1})(s+z_{o2}))}{s}, \quad g_o > 0, \quad z_{o1} > 0, \quad z_{o2} > 0 \quad (2.6.2)$$

Inner Loop Transfer Function:

$$L_i = P_i K_i = \frac{g_i(s+z_{i1})(s+z_{i2})}{(s-p)(s+p)} \quad (2.6.3)$$

Inner Loop Sensitivity:

$$S_i = \frac{1}{1+L_i} = \frac{s(s-p)(s+p)}{s^3 + g_i s^2 + (g_i z_{i1} + g_i z_{i2} - p^2)s + g_i z_{i1} z_{i2}} \quad (2.6.4)$$

Inner Loop Complementary Sensitivity:

$$T_i = \frac{L_i}{1+L_i} = \frac{g_i(s+z_{i1})(s+z_{i2})}{s^3 + g_i s^2 + (g_i z_{i1} + g_i z_{i2} - p^2)s + g_i z_{i1} z_{i2}} \quad (2.6.5)$$

Loop Transfer Function at Output:

$$L_o = P_o \left(\frac{P_i}{1+P_i K_i}\right) K_o = \frac{g_o(z-s)(s+z_{o1})(s+z_{o2})}{s^2(s^3 + g_i s^2 + (g_i z_{i1} + g_i z_{i2} - p^2)s + g_i z_{i1} z_{i2}} \quad (2.6.6)$$

Loop Transfer Function at Controls:

$$L_u = K_i P_i + K_o P_o P_i = \frac{g_i s^4 + c4 s^3 + c2 s^2 + c3 s + g_o z z_{o1} z_{o2}}{s^3(s-p)(s+p)} \quad (2.6.7)$$

Sensitivity at Output:

$$S_o = S_e = \frac{1}{1 + L_o} = \frac{s^2(s^3 + g_i s^2 + (g_i z_{i1} + g_i z_{i2} - p^2)s + g_i z_{i1} z_{i2})}{s^5 + g_i s^4 + c1 s^3 + c2 s^2 + c3 s + g_o z z_{o1} z_{o2}} \tag{2.6.8}$$

Sensitivity at Controls:

$$S_u = \frac{1}{1 + L_u} = \frac{s^3(s - p)(s + p)}{s^5 + g_i s^4 + c1 s^3 + c2 s^2 + c3 s + g_o z z_{o1} z_{o2}} \tag{2.6.9}$$

Inner Outer Loop Identity:

$$S_i = \frac{S_u}{S_o} \tag{2.6.10}$$

Complementary Sensitivity at Output:

$$T_o = T_e = \frac{g_o(z - s)(s + z_{o1})(s + z_{o2})}{s^5 + g_i s^4 + c1 s^3 + c2 s^2 + c3 s + g_o z z_{o1} z_{o2}} \tag{2.6.11}$$

Complementary Sensitivity at Controls:

$$T_u = \frac{L_u}{1 + L_u} = \frac{g_i s^4 + c4 s^3 + c2 s^2 + c3 s + g_o z z_{o1} z_{o2}}{s^5 + g_i s^4 + c1 s^3 + c2 s^2 + c3 s + g_o z z_{o1} z_{o2}} \tag{2.6.12}$$

$$c1 = (g_i z_{i1} + g_i z_{i2} - g_o - p^2); c2 = (g_o z - g_o z_{o1} - g_o z_{o2} + g_i z_{i1} z_{i2} + g_i z_{i1} z_{i2});$$
$$\tag{2.6.13}$$

$$c3 = (g_o z z_{o1} + g_o z z_{o2} - g_o z_{o1} z_{o2}); c4 = (g_i z_{i1} + g_i z_{i2} - g_o);$$
$$\tag{2.6.14}$$

2.6.1 Fundamental Sensitivity Relationships

Similar to example 1. The tradeoff between the properties of the loop at controls and the loop at output is given by the inner-outer loop identity:

- **Identity:** $S_u = S_o S_i$

From this identity it is observed that:

- **lemma:** $\|S_i\|_{\mathcal{H}^\infty} \geq 1, \|S_o\|_{\mathcal{H}^\infty} \leq \|S_u\|_{\mathcal{H}^\infty} : \|S_i\|_{\mathcal{H}^\infty} \geq 1, \|T_u\|_{\mathcal{H}^\infty} \leq \|T_o\|_{\mathcal{H}^\infty}$
  (since $S_o = 1 - T_o, S_u = 1 - T_u$).

- **lemma:** $\|S_i\|_{\mathcal{H}^\infty} < 1, \|S_u\|_{\mathcal{H}^\infty} < \|S_o\|_{\mathcal{H}^\infty} : \|S_i\|_{\mathcal{H}^\infty} < 1, \|T_o\|_{\mathcal{H}^\infty} < \|T_u\|_{\mathcal{H}^\infty}$
  (since $S_o = 1 - T_o, S_u = 1 - T_u$).

1. g_i - stability is very complicated to compute analytically, since the system has more parameters. So it is computed numerically. The code for computing it numerically is given in the appendix.

2. **Corollary 1:** $p -$ **Non-Aggressive Inner Loop:** $\|S_i\|_{\mathcal{H}^\infty} \geq 1 \iff g_i -$ $stability < g_i \leq 2p \iff \|S_o\|_{\mathcal{H}^\infty} \leq \|S_u\|_{\mathcal{H}^\infty} ; z_i \approx 0:$

3. **Corollary 2:** $p -$ **Aggressive Inner Loop:** $\|S_i\|_{\mathcal{H}^\infty} < 1 \iff \max\{g_i -$ $stability, 2p\} < g_i \iff \|S_o\|_{\mathcal{H}^\infty} > \|S_u\|_{\mathcal{H}^\infty} ; z_i \approx 0.$

4. **Corollary 3:** $g -$ **Aggressive Inner Loop:** $\|S_i\|_{\mathcal{H}^\infty} < 1 \iff \max\{g_i -$ $stability, 2g_o\} < g_i \iff \|T_o\|_{\mathcal{H}^\infty} < \|T_u\|_{\mathcal{H}^\infty} ; z_i \approx 0.$

5. **Comment: Peak Sensitivities:** $\|S\|_{\mathcal{H}^\infty}, \|T\|_{\mathcal{H}^\infty}$ increase with increasing g_o and decreasing g_i.

2.6.2   Closed Loop Stability

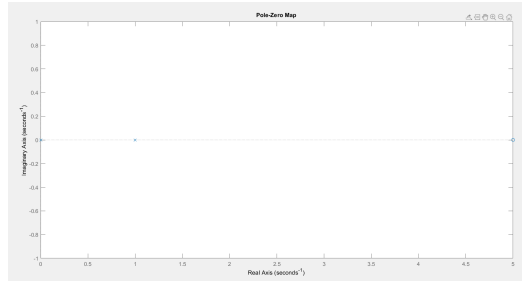Similar to example 1, we get the stability of the inner loop as:
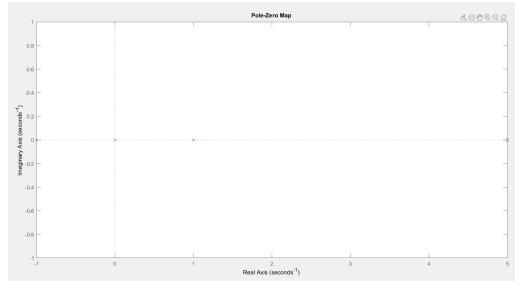
Figure 2.6.2: Nominal System: Open Loop Transfer Function

1. If $g_i \approx stability condition$; the system has poles near the imaginary axis which leads to higher peak sensitivities t for the loop at output and for the loop at controls.
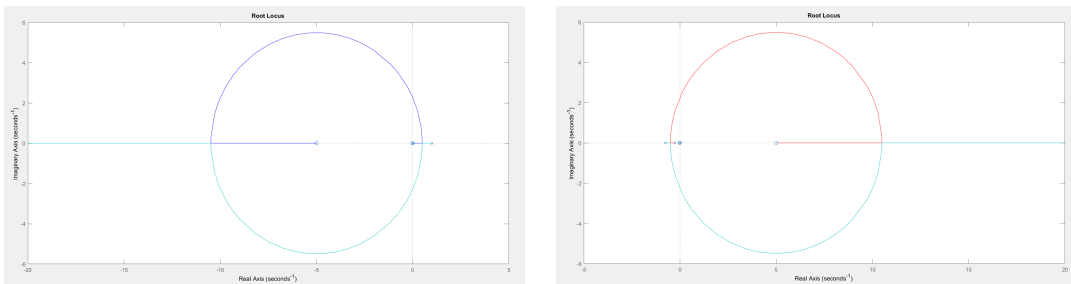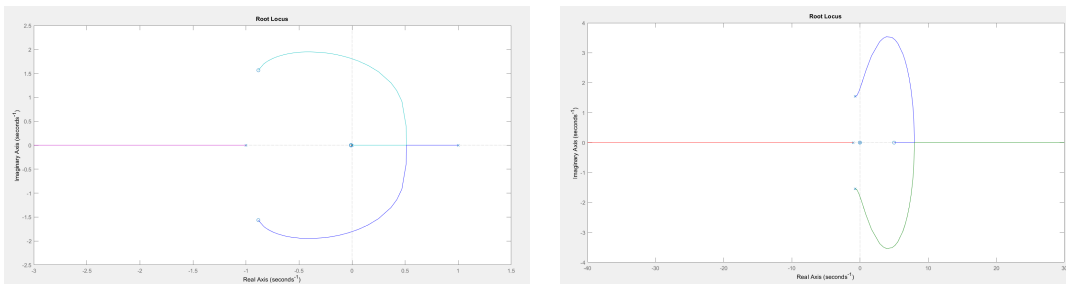

Figure 2.6.3: Nominal System: Root locus of Loop Transfer Function at Controls (left) and at Output (right)

The above plots explain that the $\uparrow GM$ at controls is infinite, $\downarrow GM$ at controls is finite. On the other hand, $\downarrow GM$ at output is zero, $\uparrow GM$ is finite because of the right hand plane zero. Thus, this cause the system to be bandwidth limited.

94

2.6.3   Open Loop Trade Studies

Varying Outer Loop Gain g_o with g_i fixed at $g_i = 5$

Loop at Input:
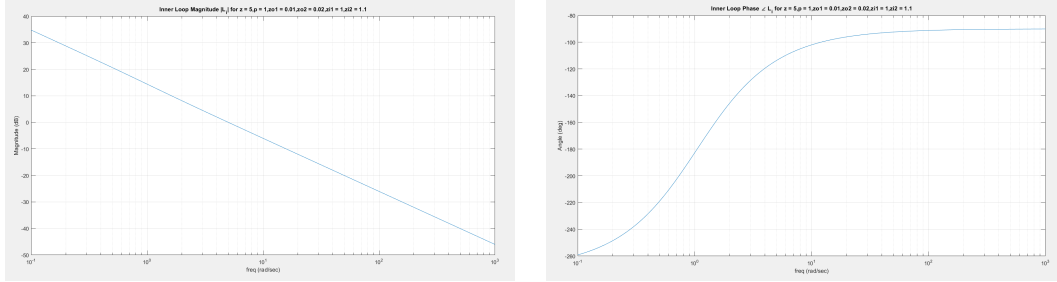
Figure 2.6.4: Inner Loop Magnitude $|L_i|$ and Phase $\angle L_i$

Loop at Output and Loop at Controls:
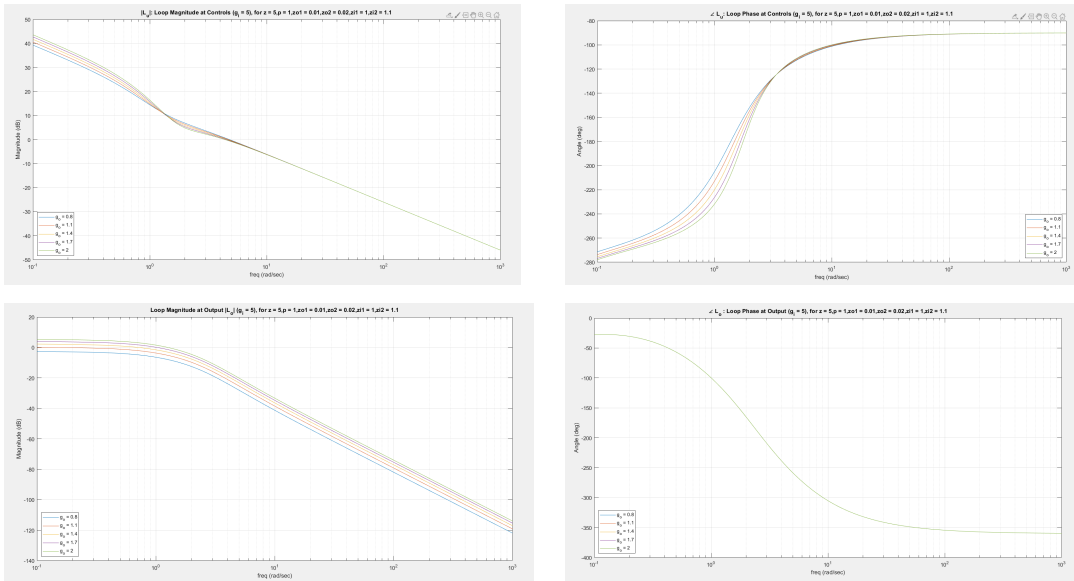
Figure 2.6.5: L_u and L_o: Loop Magnitude and Loop Phase ($g_i = 5$) for $z = 5$

- $|L_o|$ increases with increasing g_o, ω_{go} decreases with increasing g_o, $\angle L_o$ is independent of g_o, PM is independent of g_o, PM decreases with increasing g_o.

95

Varying Outer Loop Gain g_i with g_o fixed at $g_o = 2$
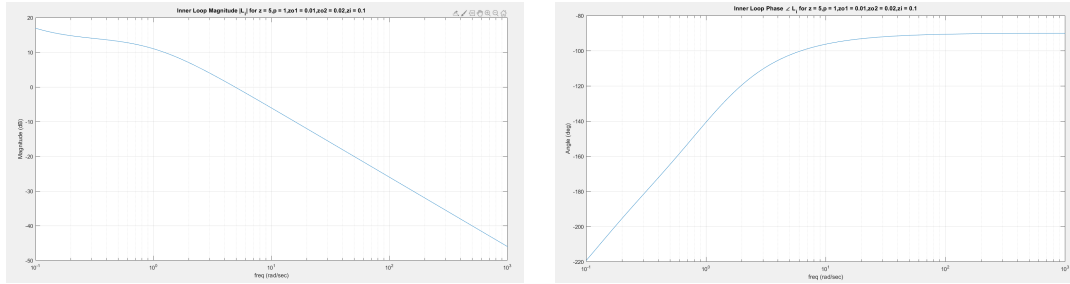
Loop at Input:
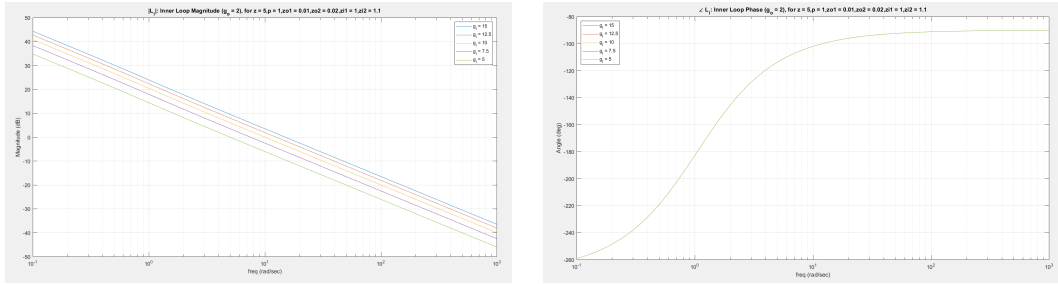
Figure 2.6.6: Inner Loop Magnitude $|L_i|$ and Phase $\angle L_i$
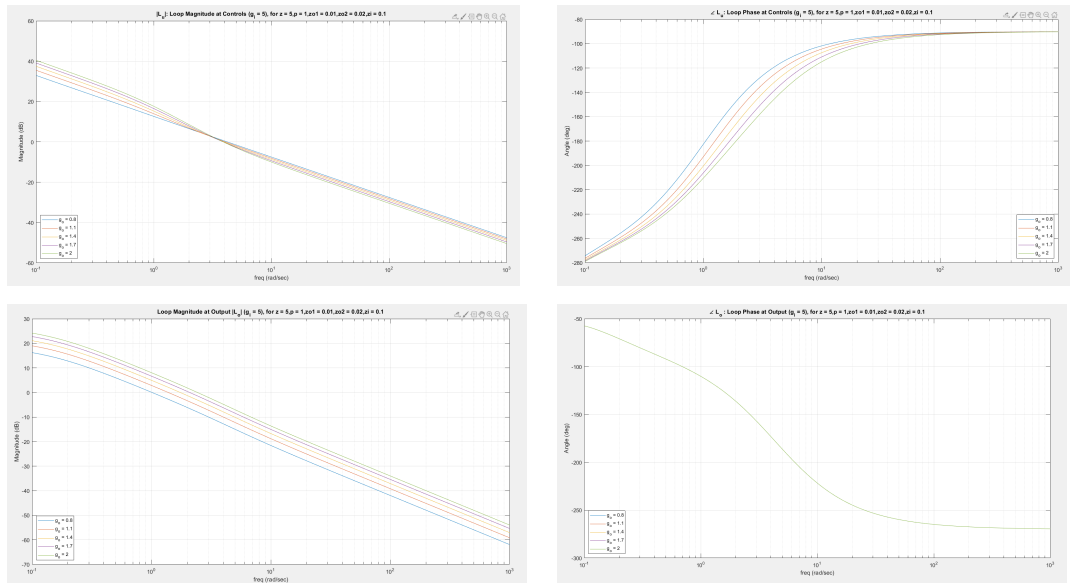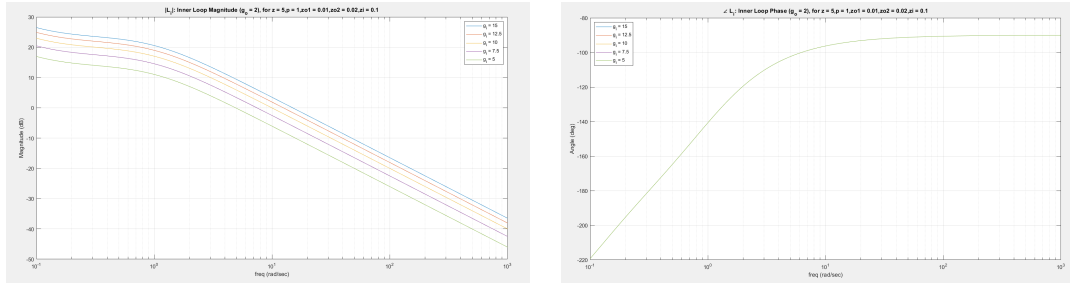
Loop at Output and Loop at Controls:
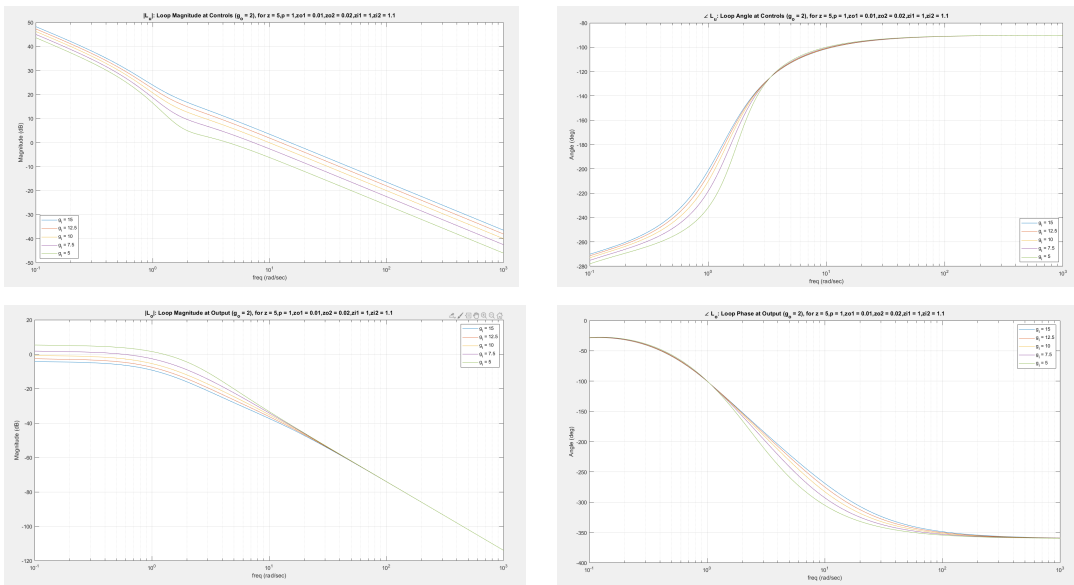
Figure 2.6.7: L_u and L_o: Loop Magnitude and Loop Phase ($g_o = 2$)

- $|L_o|$ increases with increasing g_o, ω_{go} decreases with increasing g_o, $\angle L_o$ is independent of g_o, PM is independent of g_o, PM decreases with increasing g_o.

Varying Outer Loop Gain g_o with Inner Loop Gain g_i Fixed at $g_i = 5$

Inner Loop Sensitivity and Complementary Sensitivity
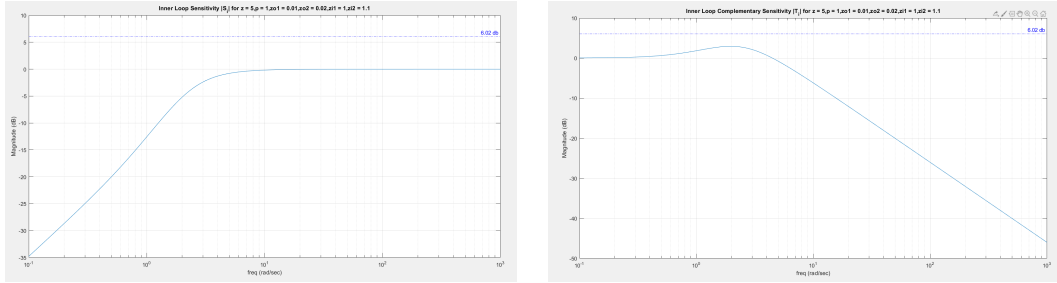
Figure 2.6.8: Inner Loop Sensitivity and Complementary Sensitivity $|S_i|$ and $|T_i|$

Loop Sensitivity at Output and at Controls
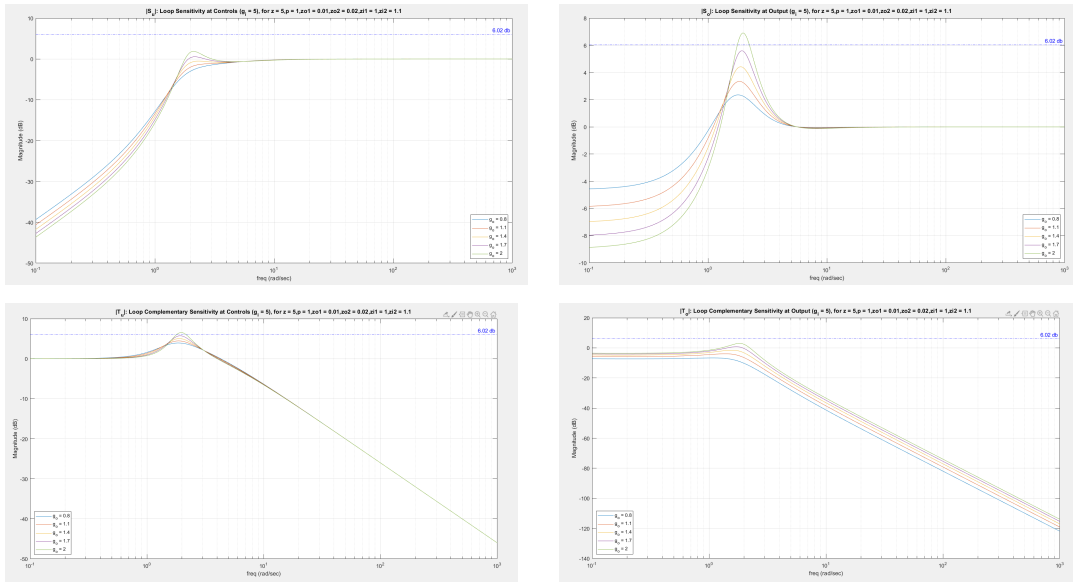
Figure 2.6.9: $|S_o|$ and $|S_u|$: Loop Sensitivity at Output and at Controls ($g_i = 5$)

- Peak $|S_o|$, $|S_u|$, $|T_o|$, $|T_u|$ increases with increasing g_o or decreasing g_i.

Inner Outer Loop Complementary Sensitivity Ratio $\left|\frac{T_u}{T_o}\right|$ vs g_i (g_o - parameter)
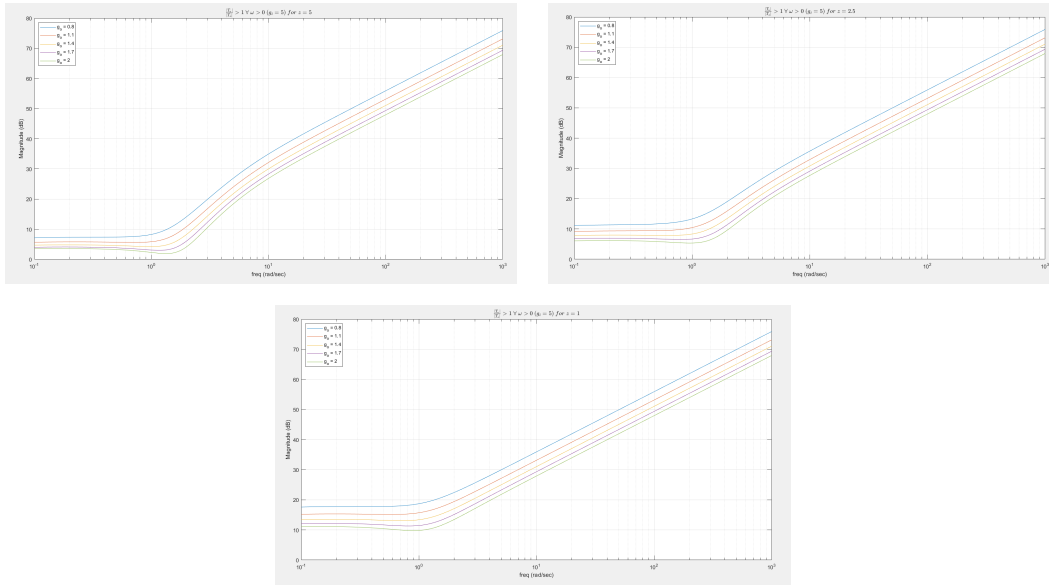
Figure 2.6.10: $\left|\frac{T_u}{T_o}\right| > 1 \ \forall \ \omega \ > 0 \ (g_i = 5)$

Peak Sensitivity and Complimentary Sensitivity
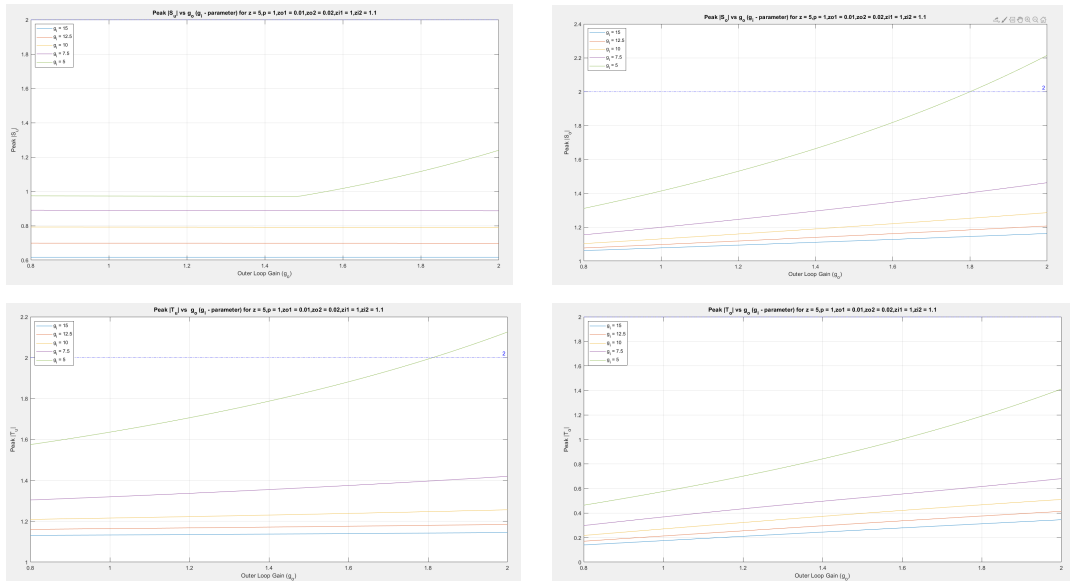
Figure 2.6.11: Peak $|S_u|$, $|S_o|$, $|T_u|$, $|T_o|$ vs g_o (g_i-parameter)

Varying Outer Loop Gain g_i with g_o fixed at $g_o = 2$

Inner Loop Sensitivity and Complementary Sensitivity
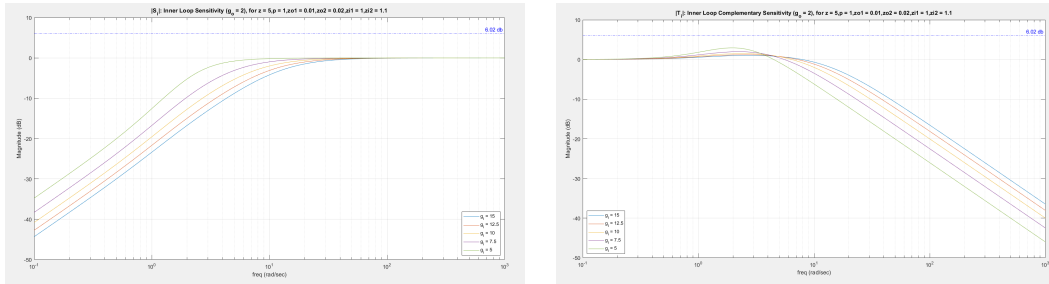
Figure 2.6.12: $|S_i|$ and $|T_i|$: Inner Loop Sensitivity and Complementary Sensitivity

$(g_o = 2)$

Loop Sensitivity and Complimentary sensitivity at Output and at Controls

Figure 2.6.13: $|S_u|$, $|S_o|$, $|T_u|$, $|T_o|$: Loop Sensitivity at Output and at Controls

$(g_o = 2)$

- $|S_i| < 1 \Rightarrow |S_o| > |S_u|, \ T_u - T_i = S_i T_o, \ T_u - T_o = S_o T_i$

99

$\left|\frac{T_u}{T_o}\right|$ **vs** g_i **(**g_o **- parameter)**
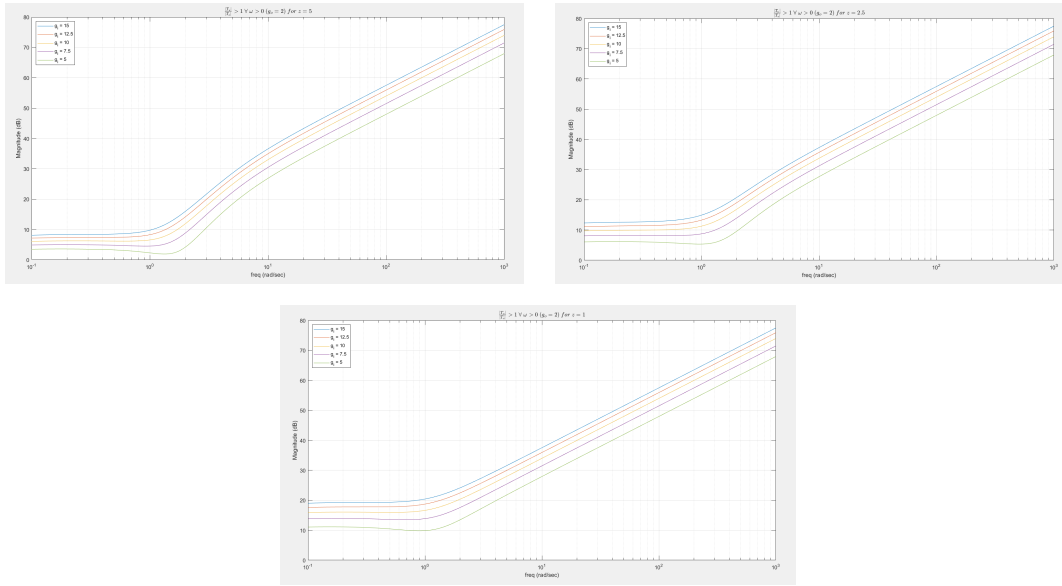
Figure 2.6.14: $\left|\frac{T_u}{T_o}\right| > 1 \ \forall \ \omega \ > 0 \ (g_o = 2)$

Peak $|S_o|$ **,** $|S_u|$**,** $|T_o|$**,** $|T_u|$
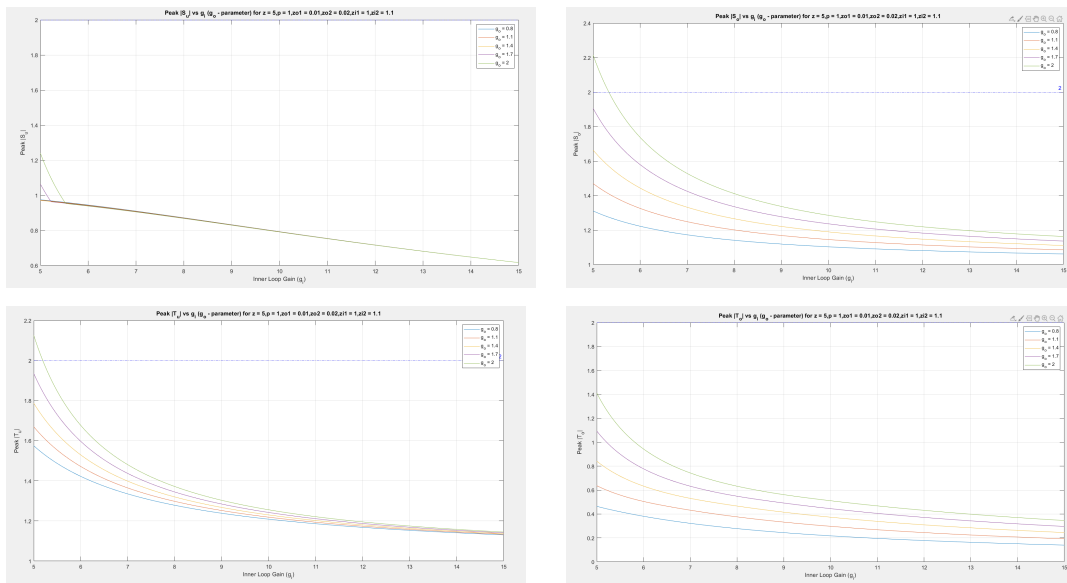
Figure 2.6.15: Peak $|S_o|$, $|S_u|$, $|T_o|$, $|T_u|$ vs g_o (g_i-parameter)

- Peak $|S_o|$, $|S_u|$, $|T_o|$, $|T_u|$ increases with increasing g_o or decreasing g_i.

2.6.5   Closed Loop Time Response Trade Studies

Varying Outer Loop Gain g_o with Inner Loop Gain g_i Fixed at $g_i = 5$

Time Domain Analysis: Impulse S_o, S_u, T_o, T_u
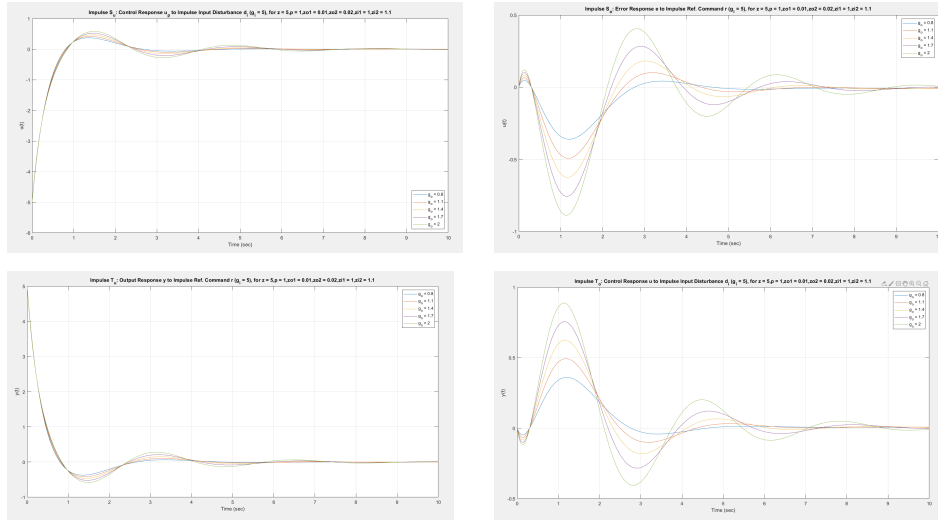
Figure 2.6.16: Impulse S_o, S_u, T_o, T_u ($g_i = 5$)

Time Domain Analysis: Step S_o, S_u, T_o, T_u
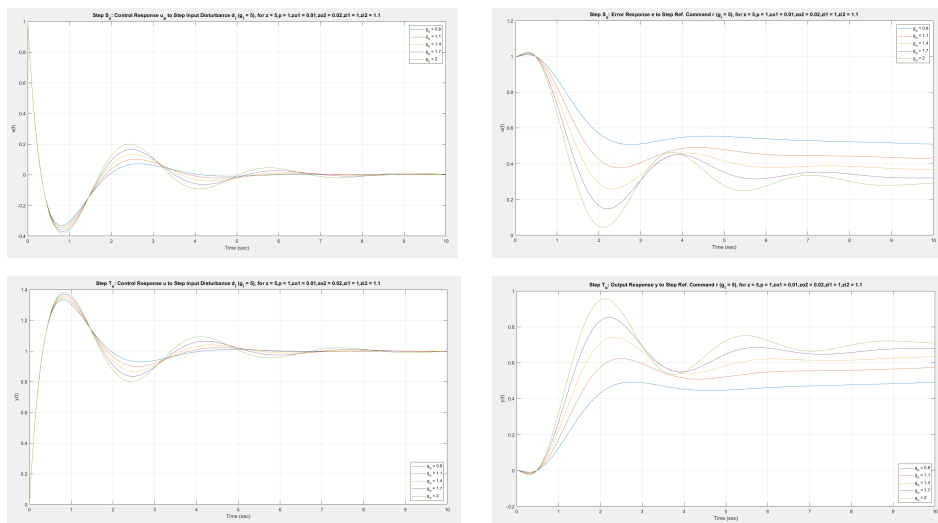
Figure 2.6.17: Step S_o, S_u, T_o, T_u ($g_i = 5$)

101

Varying Outer Loop Gain g_o with Inner Loop Gain g_i Fixed at $g_o = 2$

Time Domain Analysis: Impulse S_o, S_u, T_o, T_u
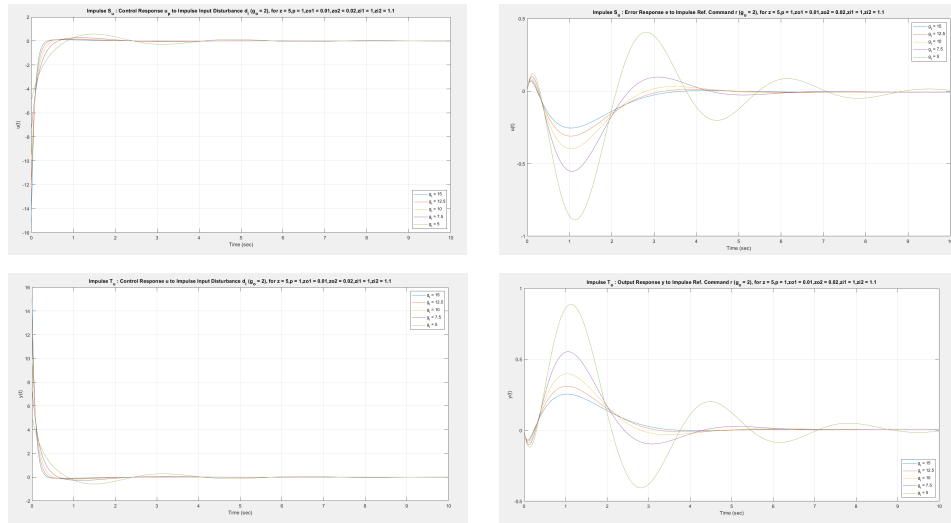
Figure 2.6.18: Impulse S_o, S_u, T_o, T_u $(g_o = 2)$

Time Domain Analysis: Step S_o, S_u, T_o, T_u
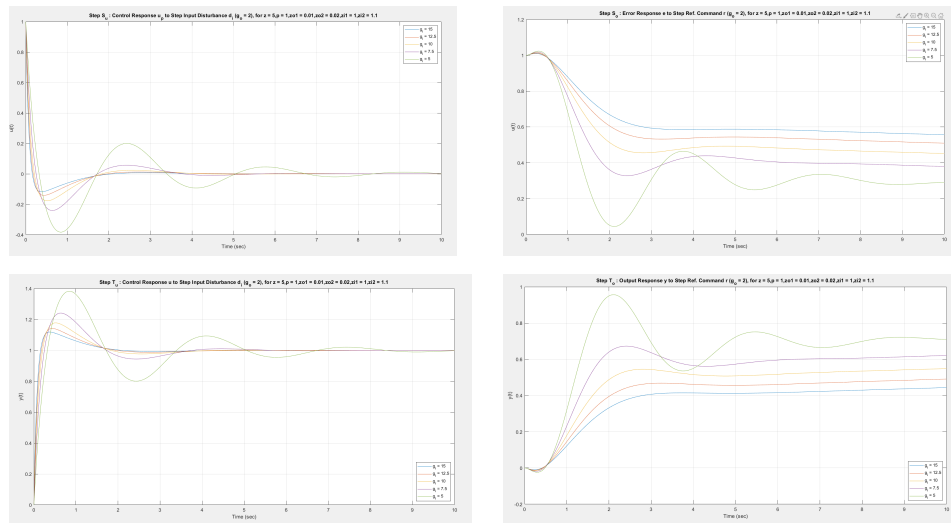
Figure 2.6.19: Step S_o, S_u, T_o, T_u $(g_o = 2)$

- Zero steady state error due to integrator in K_o.

2.6.6   Summary and Conclusions

	$g_o \uparrow$	$g_i \downarrow$	$z \downarrow$
$\|L_u\|$	$\uparrow$	$\downarrow$	$\downarrow$
$\angle L_u$	$\downarrow$	$\downarrow$	$\uparrow$
$\|L_o\|$	$\uparrow$	$\uparrow$	$\downarrow$
$\angle L_o$	$-$	$\downarrow$	$\downarrow$
$\downarrow GM$	$\uparrow$	$\uparrow$	$-$
$\omega_{\downarrow GM}$	$\uparrow$	$\uparrow$	$\downarrow$
$\uparrow GM$	$\downarrow$	$\downarrow$	$-$
$\omega_{\uparrow GM}$	$-$	$\downarrow$	$\downarrow$
PM_u	$\downarrow$	$\downarrow$	$\uparrow$
PM_o	$\downarrow$	$\downarrow$	$\downarrow$
DM_u	$\uparrow$	$\uparrow$	$\uparrow$
DM_o	$\downarrow$	$\downarrow$	$\uparrow$
ω_{gu}	$\downarrow$	$\downarrow$	$\uparrow$
ω_{go}	$\uparrow$	$\uparrow$	$\downarrow$
$PeakS_u$	$\uparrow$	$\uparrow$	$\downarrow$
$PeakS_o$	$\uparrow$	$\uparrow$	$\downarrow$
$PeakT_u$	$\uparrow$	$\uparrow$	$\downarrow$
$PeakT_o$	$\uparrow$	$\uparrow$	$\downarrow$
$\|S_i\|$	$-$	$\uparrow$	$-$
$\|T_i\|$	$-$	$\downarrow$	$-$

Table 2.6.2: Summary: Example 6

2.7 Example 7 Set Up and Assumptions

Block Diagram. The relevant block diagram can be found in Figure 2.7.1.

Figure 2.7.1: Block Diagram for Simple System

Nominal Parameters:

Parameter	Nominal Value	Description
p	1	Pole of P_i
z	5	Zero of P_o
g_o	(0.8-2)	Outer Loop Controller K_o Gain (will be varied)
g_i	(5-15)	Inner Loop Controller K_i Gain (will be varied)
z_{i1}	(0.1)	Inner Loop Controller K_i Zero (will be varied)
z_{i2}	(2)	Inner Loop Controller K_i Zero (will be varied)
z_{o1}	(0.05)	Outer Loop Controller K_o Zero (will be varied)
z_{o2}	(0.05)	Outer Loop Controller K_o Zero (will be varied)

Table 2.7.1: Nominal Parameter Values

Nominal System:

$$P_i = \frac{g_{pi}}{(s-p)(s+p)}, \quad K_i = \frac{g_i(s+z_{i1})(s+z_{i2}))}{s}, \quad g_i > 0, z_{i1} > 0.z_{i2} > 0 \quad (2.7.1)$$

$$P_o = \frac{(z-s)(z+s)}{s^2}, \quad K_o = \frac{g_o(s+z_{o1})(s+z_{o2}))}{s}, \quad g_o > 0, \quad z_{o1} > 0, \quad z_{o2} > 0$$

$$(2.7.2)$$

Inner Loop Transfer Function:

$$L_i = P_i K_i = \frac{g_i(s+z_{i1})(s+z_{i2})}{s^2(s-p)(s+p)} \qquad (2.7.3)$$

Inner Loop Sensitivity:

$$S_i = \frac{1}{1+L_i} = \frac{s(s-p)(s+p)}{s^3 + g_i s^2 + (g_i z_{i1} + g_i z_{i2} - p^2)s + g_i z_{i1} z_{i2}} \qquad (2.7.4)$$

Inner Loop Complementary Sensitivity:

$$T_i = \frac{L_i}{1+L_i} = \frac{g_i(s+z_{i1})(s+z_{i2})}{s^3 + g_i s^2 + (g_i z_{i1} + g_i z_{i2} - p^2)s + g_i z_{i1} z_{i2}} \qquad (2.7.5)$$

Loop Transfer Function at Output:

$$L_o = P_o \left(\frac{P_i}{1+P_i K_i} \right) K_o = \frac{g_o(z-s)(z+s)(s+z_{o1})(s+z_{o2})}{s^2(s^3 + g_i s^2 + (g_i z_{i1} + g_i z_{i2} - p^2)s + g_i z_{i1} z_{i2})} \qquad (2.7.6)$$

Loop Transfer Function at Controls:

$$L_u = K_i P_i + K_o P_o P_i = \frac{(g_i - g_o)s^4 + c4 s^3 + c2 s^2 + c3 s + g_o z^2 z_{o1} z_{o2}}{s^3(s-p)(s+p)} \qquad (2.7.7)$$

105

Sensitivity at Output:

$$S_o = S_e = \frac{1}{1 + L_o} = \frac{s^2(s^3 + g_i s + (g_i z_{i1} + g_i z_{i2} - p^2)s + g_i z_{i1} z_{i2})}{s^5 + (g_i - g_o)s^4 + c1 s^3 + c2 s^2 + c3 s + g_o z^2 z_{o1} z_{o2}} \qquad (2.7.8)$$

Sensitivity at Controls:

$$S_u = \frac{1}{1 + L_u} = \frac{s^3(s - p)(s + p)}{s^5 + (g_i - g_o)s^4 + c1 s^3 + c2 s^2 + c3 s + g_o z^2 z_{o1} z_{o2}} \qquad (2.7.9)$$

Inner Outer Loop Identity:

$$S_i = \frac{S_u}{S_o} \qquad (2.7.10)$$

Complementary Sensitivity at Output:

$$T_o = T_e = \frac{g_o(z - s)(z + s)(s + z_{o1})(s + z_{o2})}{s^5 + (g_i - g_o)s^4 + c1 s^3 + c2 s^2 + c3 s + g_o z^2 z_{o1} z_{o2}} \qquad (2.7.11)$$

Complementary Sensitivity at Controls:

$$T_u = \frac{L_u}{1 + L_u} = \frac{(g_i - g_o)s^4 + c4 s^3 + c2 s^2 + c3 s + g_o z^2 z_{o1} z_{o2}}{s^5 + (g_i - g_o)s^4 + c1 s^3 + c2 s^2 + c3 s + g_o z^2 z_{o1} z_{o2}} \qquad (2.7.12)$$

Complementary Sensitivity Identities:

$$T_u - T_o = T_i S_u \qquad (2.7.13)$$

$$T_u - T_i = T_o S_i \qquad (2.7.14)$$

$$c1 = (g_i z_{i1} + g_i z_{i2} - g_o z_{o1} - g_o z_{o2} - p^2); c2 = (g_o z^2 + g_i z_{i1} z_{i2} - g_o z_{o1} z_{o2}); \qquad (2.7.15)$$

$$c3 = (g_o z^2 z_{o1} + g_o z^2 z_{o2}); c4 = (g_i z_{i1} + g_i z_{i2} - g_o z_{o1} - g_o z_{o2}); \qquad (2.7.16)$$

2.7.1 Fundamental Sensitivity Relationships

Similar to example 1. The tradeoff between the properties of the loop at controls and the loop at output is given by the inner-outer loop identity:

- **Identity:** $S_u = S_o S_i$

From this identity it is observed that:

- **lemma:** $\|S_i\|_{\mathcal{H}^\infty} \geq 1, \|S_o\|_{\mathcal{H}^\infty} \leq \|S_u\|_{\mathcal{H}^\infty} : \|S_i\|_{\mathcal{H}^\infty} \geq 1, \|T_u\|_{\mathcal{H}^\infty} \leq \|T_o\|_{\mathcal{H}^\infty}$ (since $S_o = 1 - T_o, S_u = 1 - T_u$).

- **lemma:** $\|S_i\|_{\mathcal{H}^\infty} < 1, \|S_u\|_{\mathcal{H}^\infty} < \|S_o\|_{\mathcal{H}^\infty} : \|S_i\|_{\mathcal{H}^\infty} < 1, \|T_o\|_{\mathcal{H}^\infty} < \|T_u\|_{\mathcal{H}^\infty}$ (since $S_o = 1 - T_o, S_u = 1 - T_u$).

1. g_i - stability is very complicated to compute analytically, since the system has more parameters. So it is computed numerically. The code for computing it numerically is given in the appendix.

2. **Corollary 1:** $p -$ **Non-Aggressive Inner Loop:** $\|S_i\|_{\mathcal{H}^\infty} \geq 1 \iff g_i -$ $stability < g_i \leq 2p \iff \|S_o\|_{\mathcal{H}^\infty} \leq \|S_u\|_{\mathcal{H}^\infty} ; z_i \approx 0:$

3. **Corollary 2:** $p -$ **Aggressive Inner Loop:** $\|S_i\|_{\mathcal{H}^\infty} < 1 \iff \max\{g_i -$ $stability, 2p\} < g_i \iff \|S_o\|_{\mathcal{H}^\infty} > \|S_u\|_{\mathcal{H}^\infty} ; z_i \approx 0.$

4. **Corollary 3:** $g -$ **Aggressive Inner Loop:** $\|S_i\|_{\mathcal{H}^\infty} < 1 \iff \max\{g_i -$ $stability, 2g_o\} < g_i \iff \|T_o\|_{\mathcal{H}^\infty} < \|T_u\|_{\mathcal{H}^\infty} ; z_i \approx 0.$

5. **Comment: Peak Sensitivities:** $\|S\|_{\mathcal{H}^\infty}, \|T\|_{\mathcal{H}^\infty}$ increase with increasing g_o and decreasing g_i.

Similar to example 1, we get the stability of the inner loop as:

Figure 2.7.2: Nominal System: Open Loop Transfer Function

1. If $g_i \approx stabilitycondition$; the system has poles near the imaginary axis which leads to higher peak sensitivities t for the loop at output and for the loop at controls.
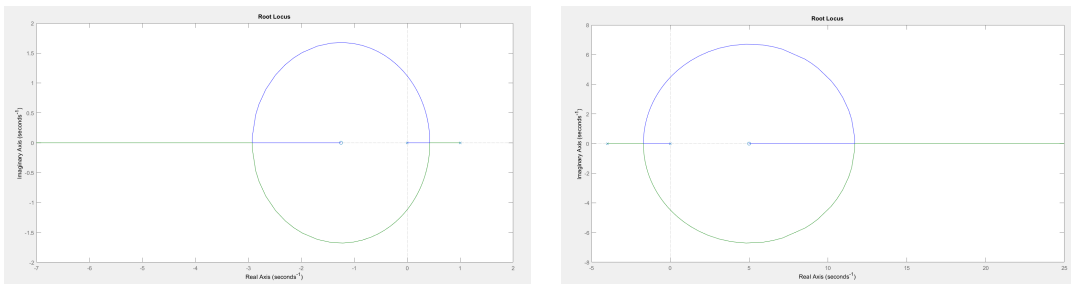
Figure 2.7.3: Nominal System: Root locus of Loop Transfer Function at Controls (left) and at Output (right)

The above plots explain that the $\uparrow GM$ at controls is infinite, $\downarrow GM$ at controls is finite. On the other hand, $\downarrow GM$ at output is zero, $\uparrow GM$ is finite because of the right hand plane zero. Thus, this cause the system to be bandwidth limited.

2.7.3   Open Loop Trade Studies

Varying Outer Loop Gain g_o with g_i fixed at $g_i = 5$

Loop at Input:
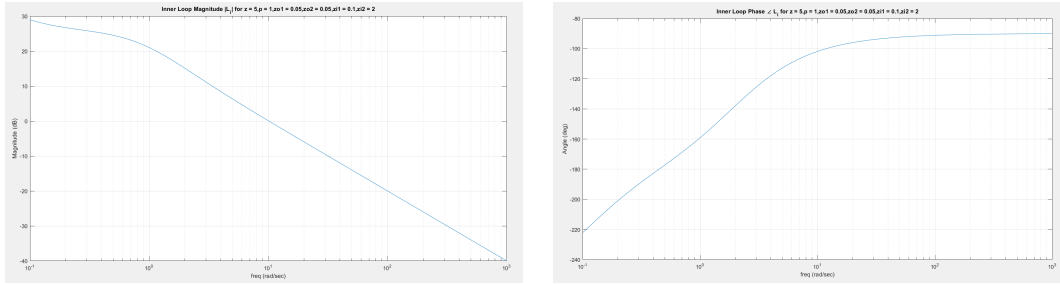
Figure 2.7.4: Inner Loop Magnitude $|L_i|$ and Phase $\angle L_i$

Loop at Output and Loop at Controls:
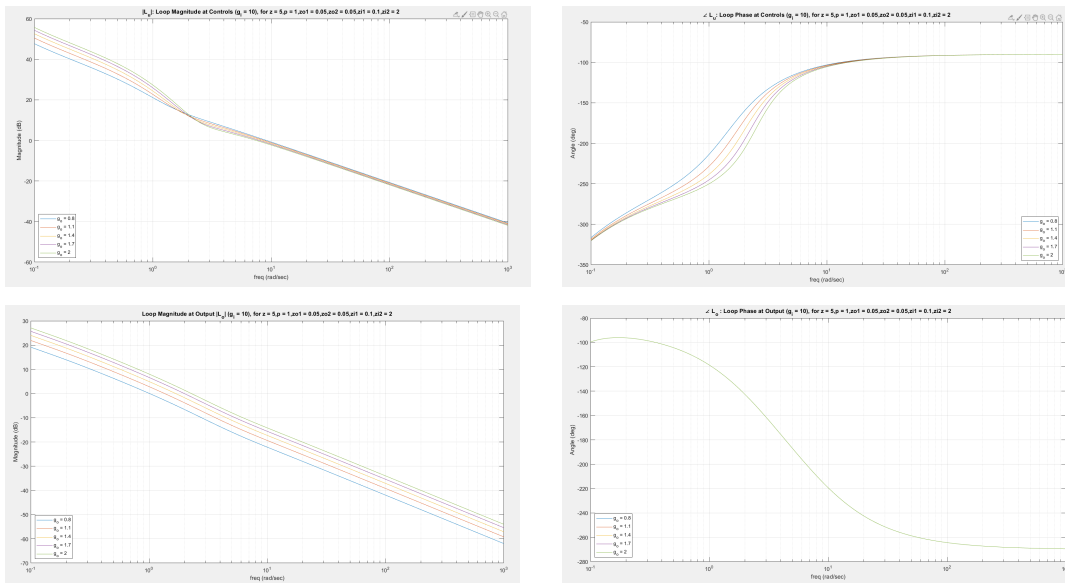
Figure 2.7.5: L_u and L_o: Loop Magnitude and Loop Phase $(g_i = 5)$ for $z = 5$

- $|L_o|$ increases with increasing g_o, ω_{go} decreases with increasing g_o, $\angle L_o$ is independent of g_o, PM is independent of g_o, PM decreases with increasing g_o

109

Varying Outer Loop Gain g_i with g_o fixed at $g_o = 2$

Loop at Input:
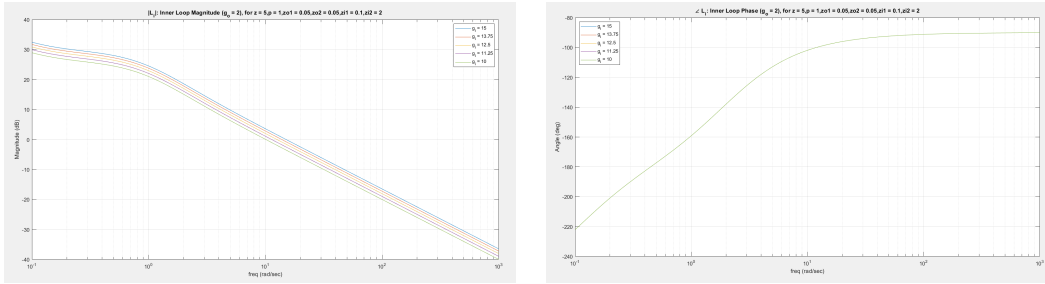
Figure 2.7.6: Inner Loop Magnitude $|L_i|$ and Phase $\angle L_i$

Loop at Output and Loop at Controls:
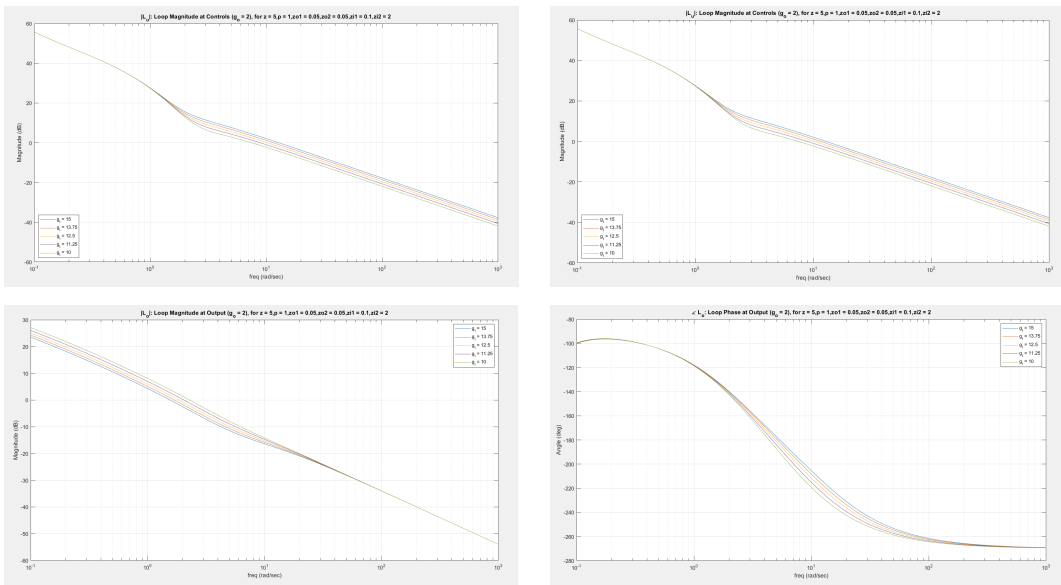
Figure 2.7.7: L_u and L_o: Loop Magnitude and Loop Phase ($g_o = 2$)

- $|L_o|$ increases with increasing g_o, ω_{go} decreases with increasing g_o, $\angle L_o$ is independent of g_o, PM is independent of g_o, PM decreases with increasing g_o.

Varying Outer Loop Gain g_o with Inner Loop Gain g_i Fixed at $g_i = 5$

Inner Loop Sensitivity and Complementary Sensitivity
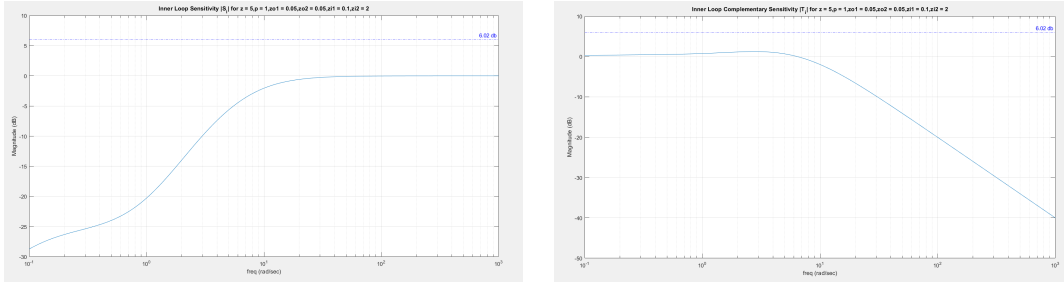
Figure 2.7.8: Inner Loop Sensitivity and Complementary Sensitivity $|S_i|$ and $|T_i|$

Loop Sensitivity at Output and at Controls
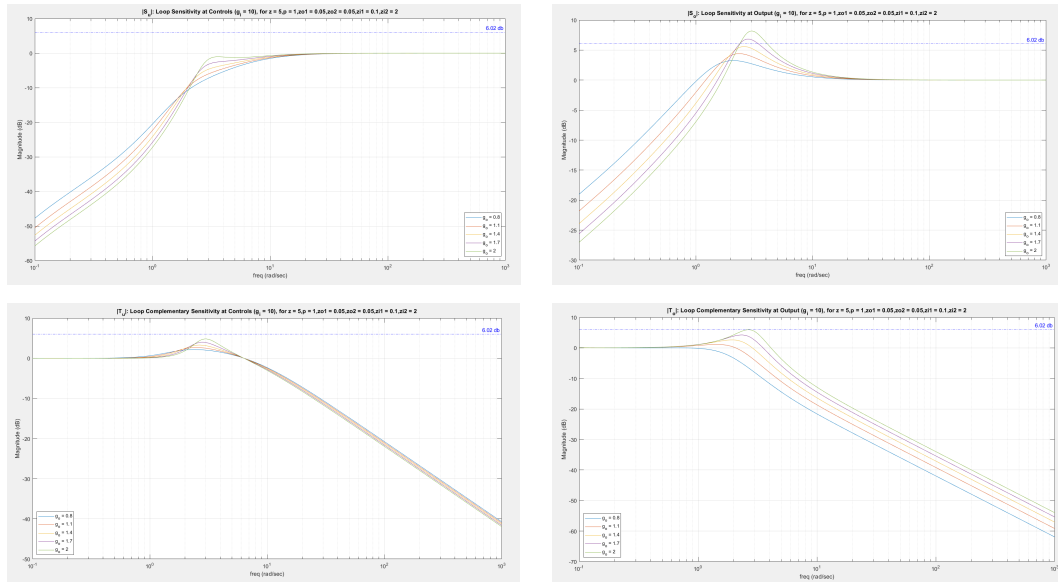
Figure 2.7.9: $|S_o|$ and $|S_u|$: Loop Sensitivity at Output and at Controls ($g_i = 5$)

- Peak $|S_o|$, $|S_u|$, $|T_o|$, $|T_u|$ increases with increasing g_o or decreasing g_i.

Inner Outer Loop Complementary Sensitivity Ratio $\left|\frac{T_u}{T_o}\right|$ vs g_i (g_o - parameter)
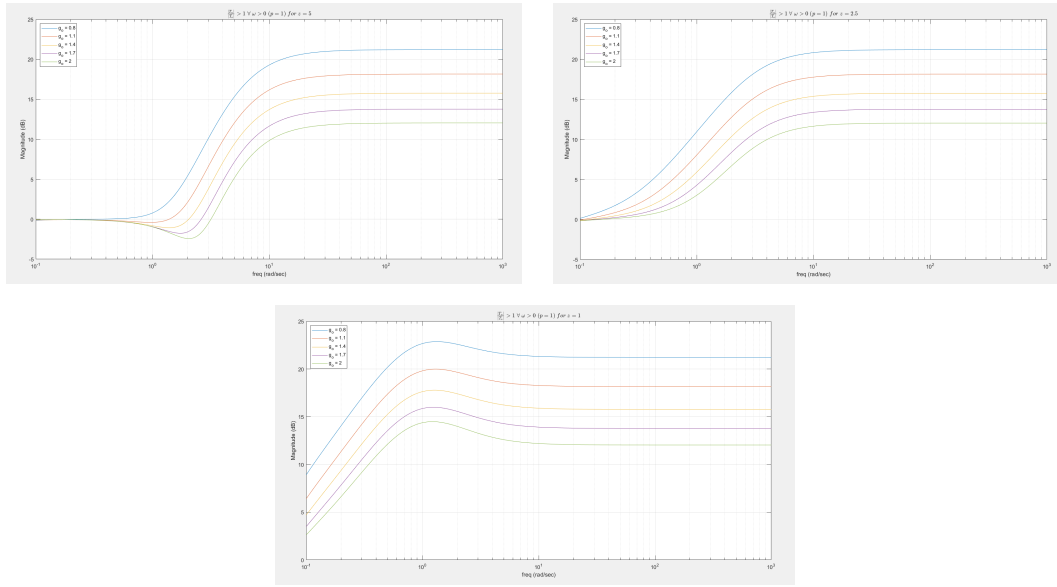
Figure 2.7.10: $\left|\frac{T_u}{T_o}\right| > 1 \ \forall \ \omega \ > 0 \ (g_i = 5)$

Peak Sensitivity and Complimentary Sensitivity
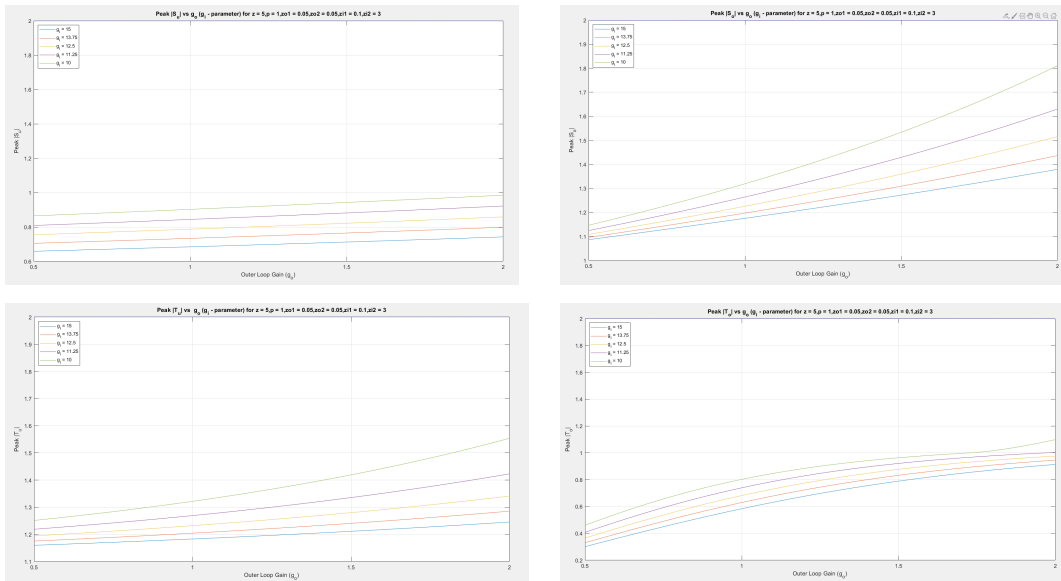
Figure 2.7.11: Peak $|S_u|$, $|S_o|$, $|T_u|$, $|T_o|$ vs g_o (g_i-parameter)

Varying Outer Loop Gain g_i **with** g_o **fixed at** $g_o = 2$

Inner Loop Sensitivity and Complementary Sensitivity
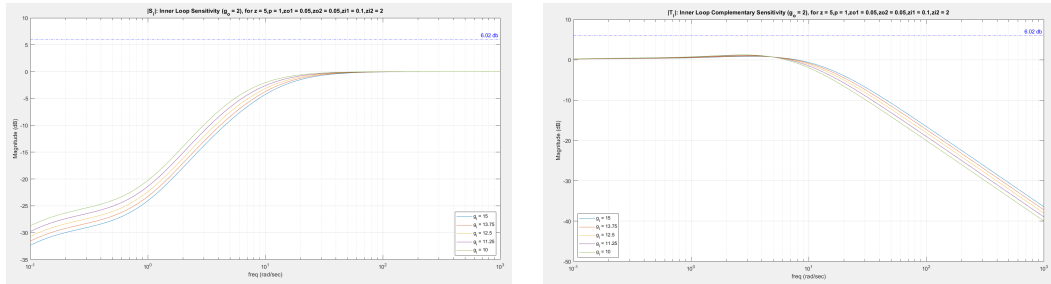
Figure 2.7.12: $|S_i|$ and $|T_i|$: Inner Loop Sensitivity and Complementary Sensitivity
$(g_o = 2)$

Loop Sensitivity and Complimentary sensitivity at Output and at Controls
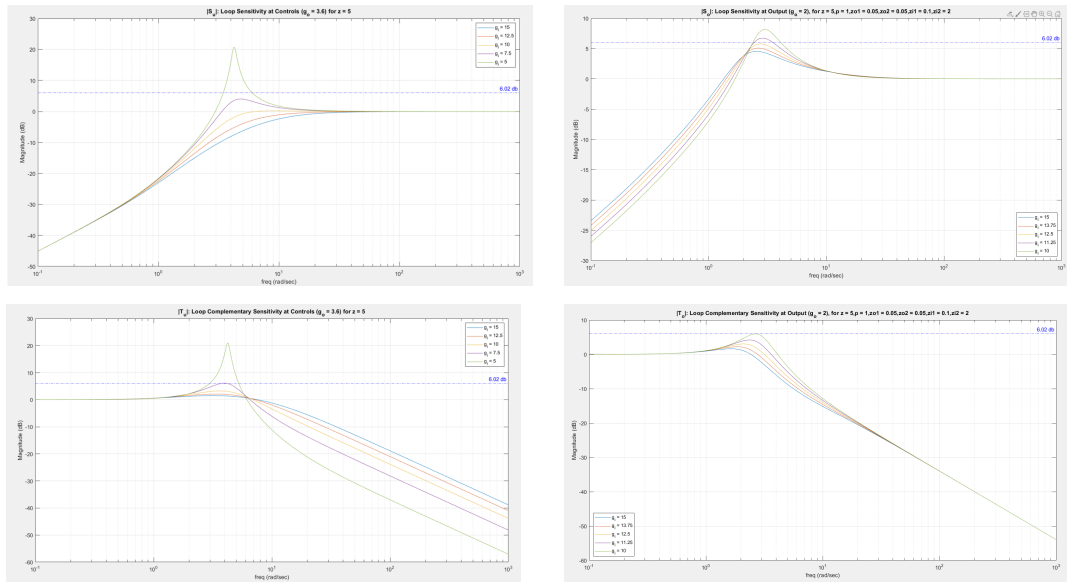
Figure 2.7.13: $|S_u|$, $|S_o|$, $|T_u|$, $|T_o|$: Loop Sensitivity at Output and at Controls
$(g_o = 2)$

- Peak $|S_o|$, $|S_u|$, $|T_o|$, $|T_u|$ increases with increasing g_o or decreasing g_i.

$\left|\frac{T_u}{T_o}\right|$ **vs** g_i $(g_o$ **- parameter)**
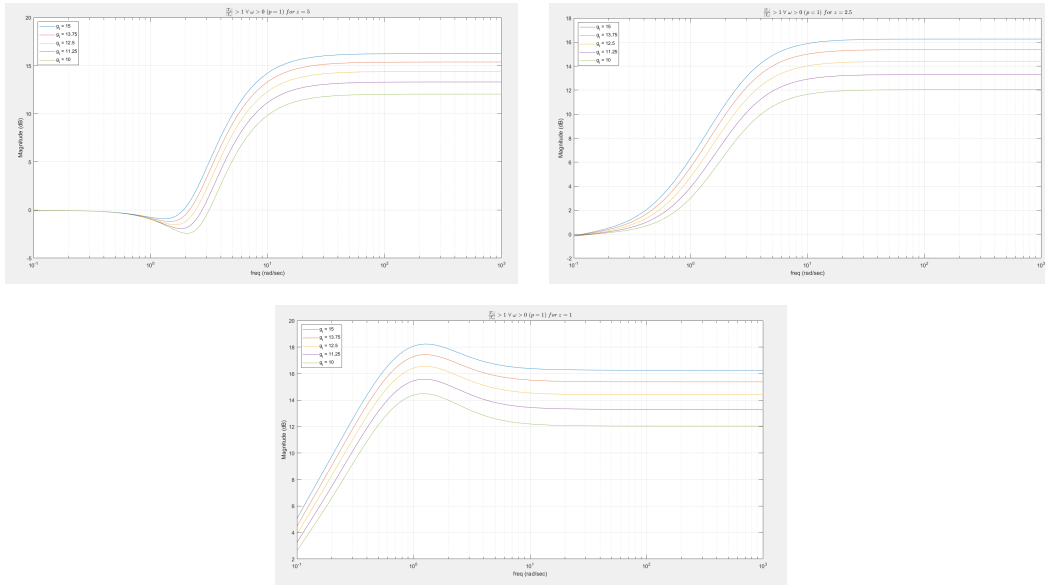
Figure 2.7.14: $\left|\frac{T_u}{T_o}\right| > 1 \ \forall \ \omega \ > 0 \ (g_o = 2)$

Peak $|S_o|$ **,** $|S_u|$**,** $|T_o|$**,** $|T_u|$
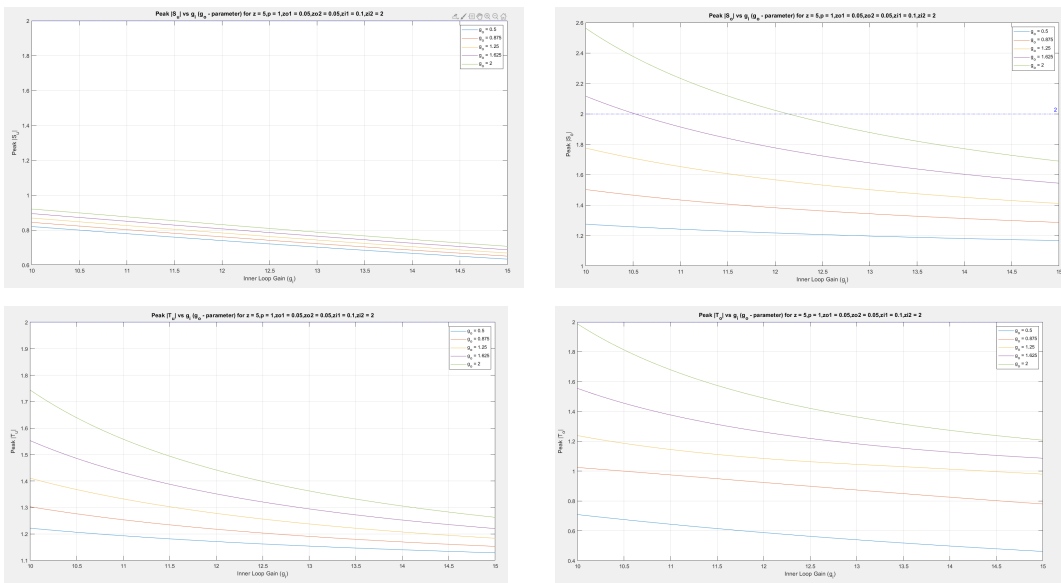
Figure 2.7.15: Peak $|S_o|$, $|S_u|$, $|T_o|$, $|T_u|$ vs g_o (g_i-parameter)

- Peak $|S_o|$, $|S_u|$, $|T_o|$, $|T_u|$ increases with increasing g_o or decreasing g_i.

114

2.7.5   Closed Loop Time Response Trade Studies

Varying Outer Loop Gain g_o with Inner Loop Gain g_i Fixed at $g_i = 5$

Time Domain Analysis: Impulse S_o, S_u, T_o, T_u
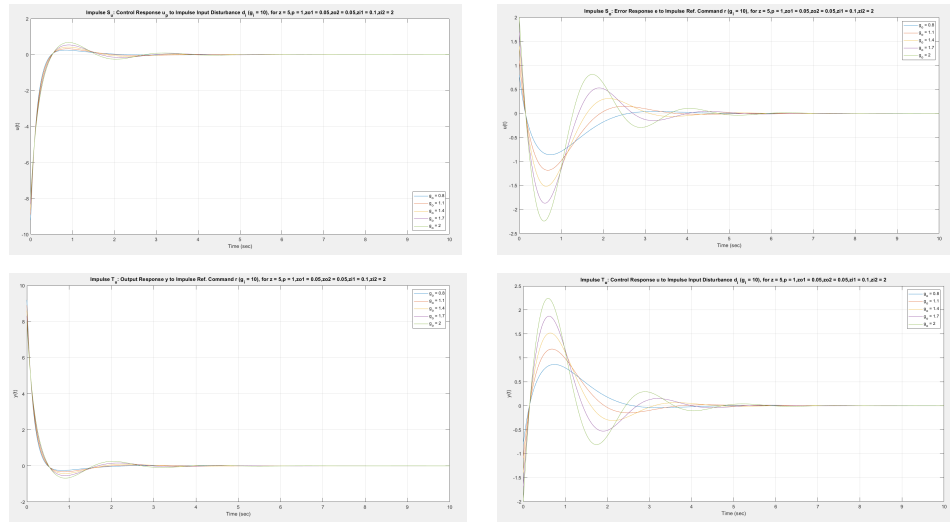
Figure 2.7.16: Impulse S_o, S_u, T_o, T_u $(g_i = 5)$

Time Domain Analysis: Step S_o, S_u, T_o, T_u
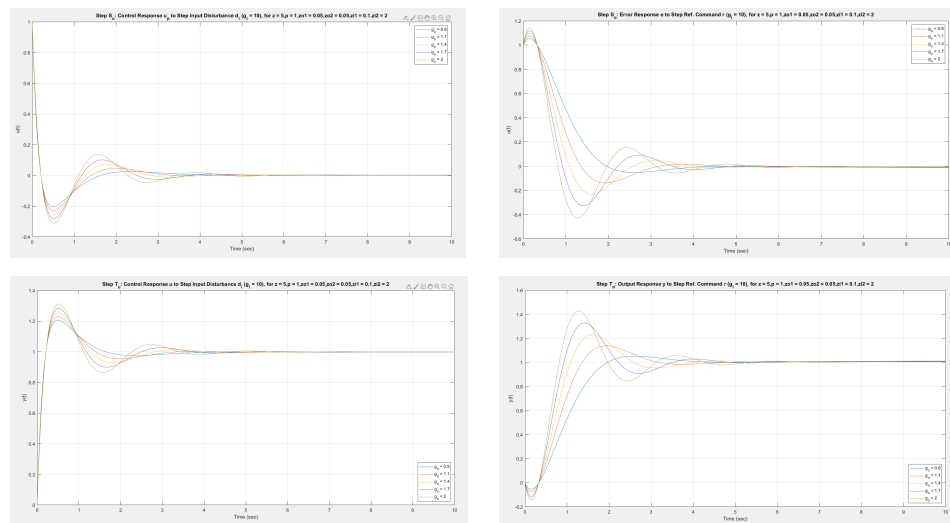
Figure 2.7.17: Step S_o, S_u, T_o, T_u $(g_i = 5)$

Varying Outer Loop Gain g_o with Inner Loop Gain g_i Fixed at $g_o = 2$

Time Domain Analysis: Impulse S_o, S_u, T_o, T_u
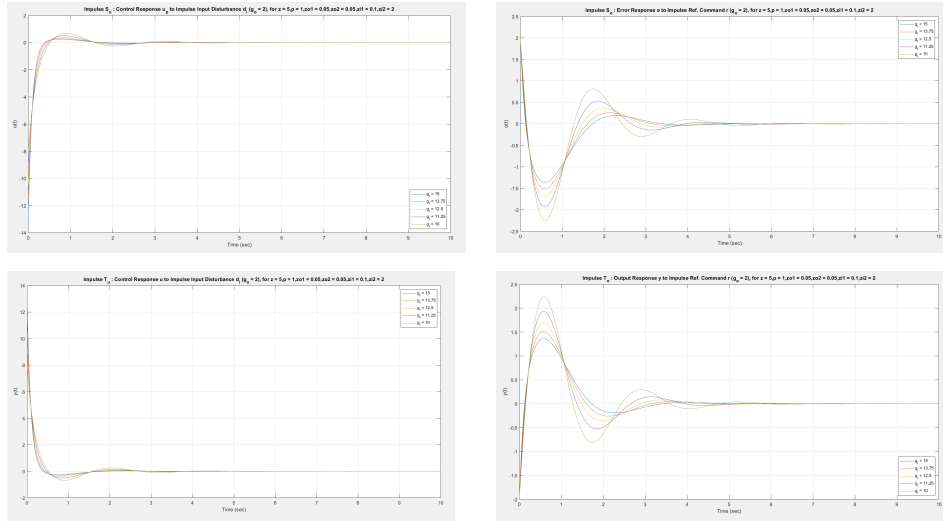
Figure 2.7.18: Impulse S_o, S_u, T_o, T_u ($g_o = 2$)

Time Domain Analysis: Step S_o, S_u, T_o, T_u
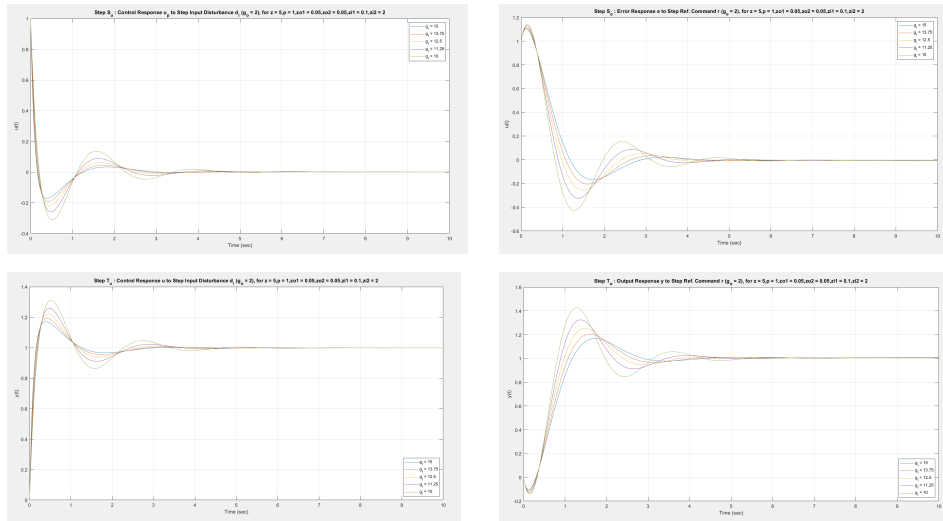
Figure 2.7.19: Step S_o, S_u, T_o, T_u ($g_o = 2$)

- Initial transient undershoot increases with increasing g_o.

116

2.7.6 Summary and Conclusions

	$g_o \uparrow$	$g_i \downarrow$	$z \downarrow$
$|L_u|$	$\uparrow$	$\downarrow$	$\downarrow$
$\angle L_u$	$\downarrow$	$\downarrow$	$\uparrow$
$|L_o|$	$\uparrow$	$\uparrow$	$\downarrow$
$\angle L_o$	$-$	$\downarrow$	$\downarrow$
$\downarrow GM$	$\uparrow$	$\uparrow$	$-$
$\omega_{\downarrow GM}$	$\uparrow$	$\uparrow$	$\downarrow$
$\uparrow GM$	$\downarrow$	$\downarrow$	$-$
$\omega_{\uparrow GM}$	$-$	$\downarrow$	$\downarrow$
PM_u	$\downarrow$	$\downarrow$	$\uparrow$
PM_o	$\downarrow$	$\downarrow$	$\downarrow$
DM_u	$\uparrow$	$\uparrow$	$\uparrow$
DM_o	$\downarrow$	$\downarrow$	$\uparrow$
ω_{gu}	$\downarrow$	$\downarrow$	$\uparrow$
ω_{go}	$\uparrow$	$\uparrow$	$\downarrow$
$PeakS_u$	$\uparrow$	$\uparrow$	$\downarrow$
$PeakS_o$	$\uparrow$	$\uparrow$	$\downarrow$
$PeakT_u$	$\uparrow$	$\uparrow$	$\downarrow$
$PeakT_o$	$\uparrow$	$\uparrow$	$\downarrow$
$|S_i|$	$-$	$\uparrow$	$-$
$|T_i|$	$-$	$\downarrow$	$-$

Table 2.7.2: Summary: Example 7

Chapter 3

NON-LINEAR CART-INVERTED PENDULUM DYNAMIC MODEL

All systems are non-linear. However, studies have been conducted with non-linear systems, involving linearization of the system about its equilibria. This gives insightful information for linear control designs that are applicable to non-linear models. To understand the nonlinearities associated with a model, it is essential to have a major understanding of its equilibria(Persson, 2013). The equilibria pertaining to a non-linear model has been used rigorously for linearization. As we proceed through this chapter, these techniques addressed will involve major linear concepts. These concepts however may or may not be successful to make a non-linear model stable. Hence a comparison between linear control design on non-linear models would be helpful and highly recommended(Prasad et al., 2014).

3.1   Motivation: What can a Cart-Inverted Pendulum Be Used to Approximate.

This thesis motivates the idea of systems in nature which have the necessity to balance themselves about their vertical position. Systems that need balance are heavily inspired by the cart-pole model. To understand the control action needed for restricting any instability, one needs to understand this model and its working. Hence, the cart-inverted problem is an excellent example to approximate the necessary requirements to satisfy the conditions of locomotion as stated above(Yi et al., 2016). This idea is illuminated from the fact that the analogies related to the examples related to the previous chapter are useful in approximation of the cart-pole system. Hence analogies from the cart-inverted problem can be applicable to a more complicated system.

3.2 Non-Linear Model: Variables, Parameters and Nominal Values

The model that has been extensively studied in this thesis (Ogata, 2001) and (Ponce et al., 2014). The system parameters are selected with respect to the references as stated above and are defined in the table below.

Variable	Nominal Value	Description
m	0.21	mass of the pendulum
L	0.61	length of the pendulum
l	0.305	position of center of gravity of pendulum from pivot
b	0	coefficient of friction
g	9.81	acceleration due to center of gravity
I	0.006	moment of inertia
M	0.455	Mass of the cart

Table 3.2.1: System Variables.

Nominal Values:

Variable	Nominal Value	(Units)
m	0.21	kg
L	0.61	m
l	0.305	m
b	0	kgm^2/s
g	9.81	m/s^2
I	0.006	$Nm/rad^2/sec^2$
M	0.455	kg

Table 3.2.2: Nominal Parameter Values.

The non-linear model pertaining to this system can be derived using euler-lagrange equations. This methodology has been very useful for deriving the states with respect to the control inputs. The code for linearizing the system about its equilibria is provided in the appendix. The equations of representing the non-linear model can be given by:

$$(M + m)\ddot{x} + ml\ddot{\theta}\cos(\theta) + b\dot{\theta} - m\dot{\theta}^2 l \sin(\theta) = F \tag{3.2.1}$$

$$(I + ml^2)\ddot{\theta} + ml\cos(\theta)\ddot{x} - mgl\sin(\theta) = 0 \tag{3.2.2}$$

Cart Inverted Pendulum: The relevant block diagram can be found in Figure 4.6.1.
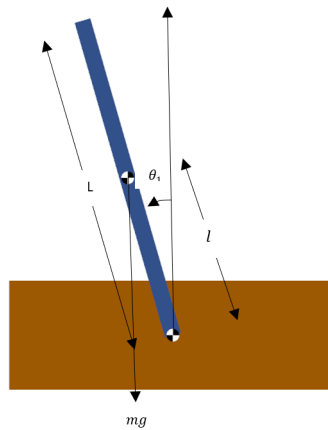
Figure 3.2.1: Block Diagram for cart-pole system

Assumptions:

The following assumptions are made for this model to get an idea of a system related to our examples:

1. The coefficient of damping b is negligible.

2. The coefficient of rotational damping associated with the pendulum is negligible.

3.3 Properties of Non-Linear Model

There are two degrees of freedom pertaining to the cart-pole system namely: one translational degree of freedom and a rotational degree of freedom x, θ and their derivatives constitute the state variables of the plant which are respectively: the cart position and the angle of the pendulum. The links are rigid with a lumped mass at the center of gravity. A single input F provides the control force. A disturbance d may also be present at the plant input and it should be similar to tapping the cart to verify that the control system is stabilizing the pendulum.

The model may be written as follows:

$$\dot{x} = F(x, u), y = Cx_p \tag{3.3.1}$$

$$u_p = \begin{bmatrix} u - force \quad on \quad the \quad cart \end{bmatrix} \tag{3.3.2}$$

$$x_p = \begin{bmatrix} x - cart & position(m) \\ \dot{x} - cart & velocity(m/sec) \\ \theta - pendulum & angle(rad) \\ \dot{\theta} - angular & velocity(rad/sec) \end{bmatrix} \tag{3.3.3}$$

$$y_p = \begin{bmatrix} x - cart & position(m) \\ \theta - pendulum & angle(rad) \end{bmatrix} \tag{3.3.4}$$

3.4 Equilibrium Analysis

The cart-pendulum equilibrium can be shown by the computing the equations
of motion and using ordinary differential equations to simulate: the position, the
velocity, the angle, the angular velocity (Persson, 2013).

To view the equilibrium, we plot : the angular velocity of the pendulum about
angular position.The figure is given in 3.4.1. Hence we get the following phase por-
traits.
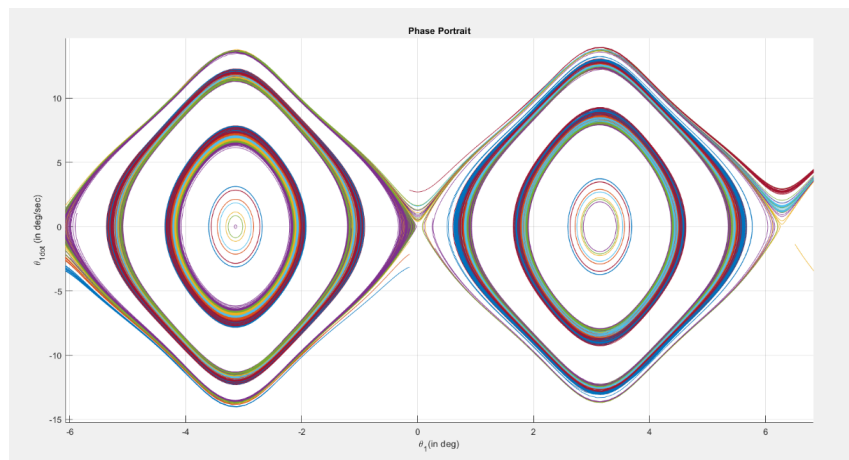
Figure 3.4.1: Angular Velocity vs Angular Position

From this figure, we get the following equilibrium points for our states.

$$x_p = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{3.4.1}$$

This CIP system has two types of equilibria associated with the pendulum - sta-
ble (pendulum resting at bottom) and unstable equilibria (pendulum about vertical
position).

3.5   Linear Model

The non-linear model is linearized about its equilbria x_p 3.4.1 and $u_p = 0$ using by using Jacobian method ($I \approx 0$, $b = 0$). The linear equations derived are as below:

$$(M + m)\ddot{x} + ml\ddot{\theta} = u \tag{3.5.1}$$

$$ml^2\ddot{\theta} + ml\ddot{x} - mgl\theta = 0 \tag{3.5.2}$$

Linearizing the model about its equilibrium results in the following state space representation for the system is given by:

$$\dot{x}_p = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{-gm}{M} & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & \frac{g(m+M)}{l} & 0 \end{bmatrix} x_p + \begin{bmatrix} 0 \\ \frac{1}{M} \\ 0 \\ \frac{-1}{Ml} \end{bmatrix} u \tag{3.5.3}$$

$$\dot{x}_p = Ax_p + Bu; y_p = Cx_p \tag{3.5.4}$$

$$\frac{\theta}{u} = \begin{bmatrix} \dfrac{\frac{-1}{Ml}}{s^2 - \frac{g(m+M)}{Ml}} \end{bmatrix} \tag{3.5.5}$$

$$\frac{x}{u} = \frac{\frac{1}{M}}{s^2} \begin{bmatrix} \dfrac{s^2 - \frac{g}{l}}{s^2 - \frac{g(m+M)}{Ml}} \end{bmatrix} \tag{3.5.6}$$

123

Poles: Pole Zero Map
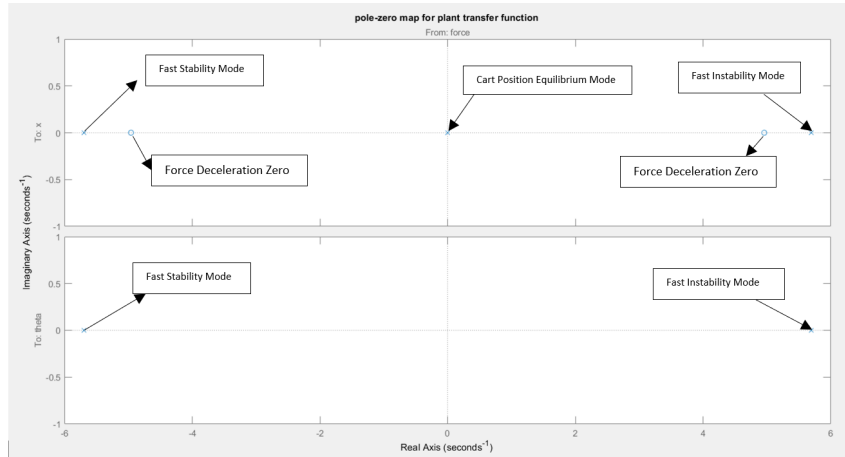
Figure 3.6.1: Pole-Zero Map

Poles:

Pole Description	Poles
Pendulum Toppling Instability	$\lambda_1 = 5.693$
Pendulum Damping	$\lambda_2 = $ -5.693
Cart Position Equlibrium	$\lambda_{3,4} = 0$

Table 3.6.1: System Poles

Zeros:

Zero Description	Zeros
Pendulum Force Acceleration	$z_1 = 4.958$
Pendulum Force Deceration	$z_2 = $ -4.958

Table 3.6.2: System Zeros

Modes:
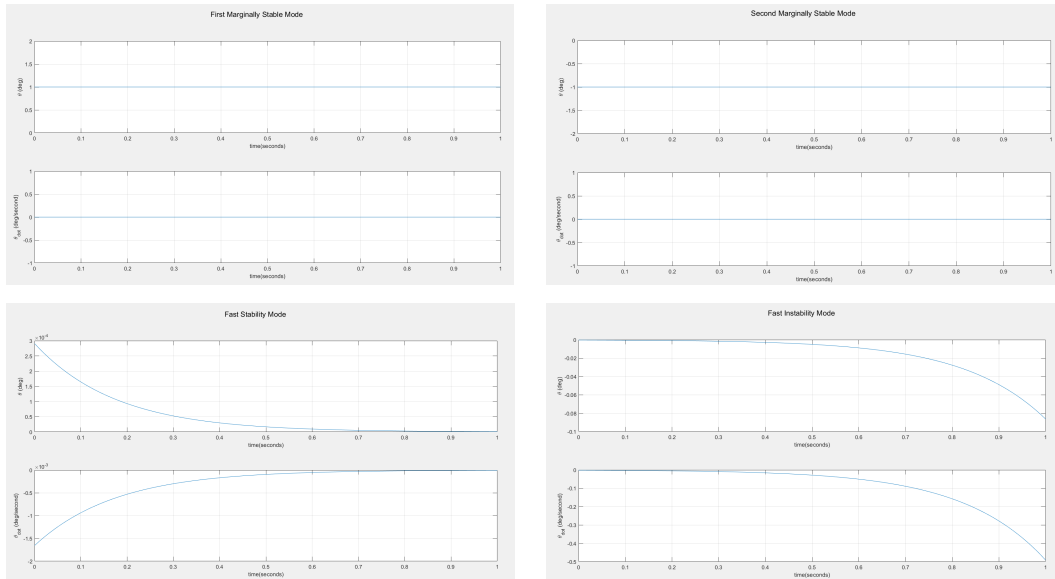
Figure 3.6.2: Modal Analysis

- *Comment: The first marginally stable mode and the second marginally stable mode corresponds to integrators in the plant responsible for cart position equilibrium. Since the poles are at the origin, the system remains balanced. Ofcourse, this is theoritically applicable and is ideal. It is understood that this mode is independent of any changes to the parameters of the system.*

- *Comment: The fast instability mode corresponds to right hand pole in the plant responsible for pendulum instability.*

- *Comment: The fast stability mode corresponds to left hand pole in the plant responsible for pendulum stability.*

The system is controllable and observable at x - position of the cart. Hence, PBH tests help prove no loss of controllability and observability.

125

3.6.1 Stability Trade Studies

The CIP systems used for the study comprises of three models:

- **System 1: Hernandez Model (Hernandez, 2000)**

- **System 2: Ogata Model(Ogata, 2001), (Ponce et al., 2014)**

- **System 3: Vignesh Model (Namasivayam, 2020)**

The poles and zeros obtained from the aforementioned models are:

Systems	Poles	Zeros
$System1$	$p = $ 0,0,-3.6,3.602	$z = $ 3.118,-3.118
$System2$	$p = $ 0,0,-5.693, 5.693	$z = $ 4.598,-4.598
$System3$	$p = $ 0,0,-9.456,9.456	$z = $ 8.821, - 8.821

Table 3.6.3: Poles And Zeros Of Respective Systems

The pole-zero map for visualizing these designs can be very insightful on the amount of bandwidth that can be used to control each of the systems. It also explains how fast the instability affects the system.
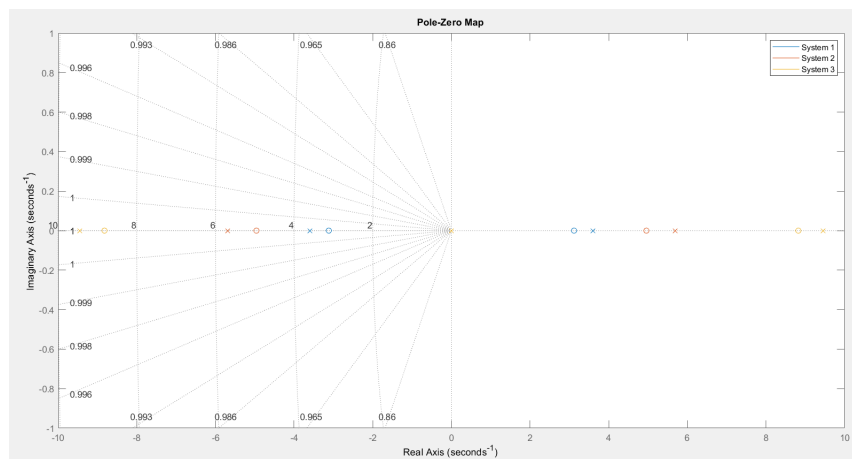
Figure 3.6.3: Pole Zero Map For Systems Discussed

- *Comment: The inner outer loop assists with stabilizing each of the systems as discussed above. The system 1 has the longest pendulum. It has poles and zeros closest to the imaginary axis compared to the other systems. Hence, it will be easiest to stabilize. The system 3 has the smallest pendulum. It has poles farthest from the imaginary axis. Consequently, the system will be hardest to stabilize. The system 2 has the pendulum of intermediate length in comparison to the others. It will be easier to stabilize in comparison to the other systems.*
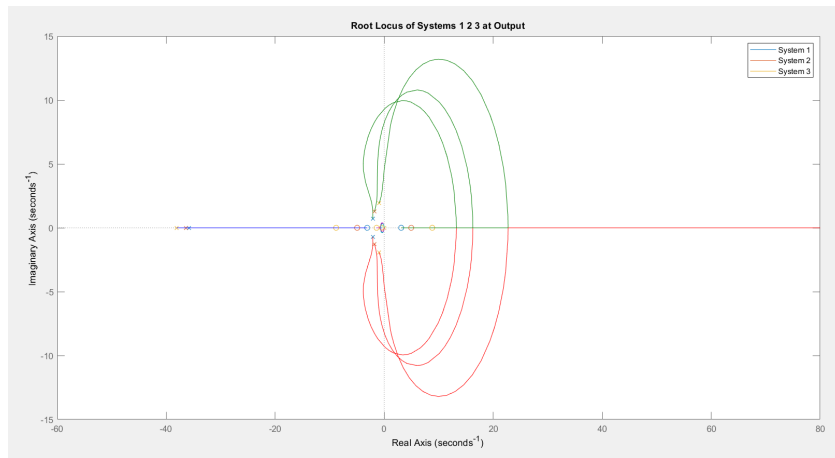
Figure 3.6.4: Root Locus At Output
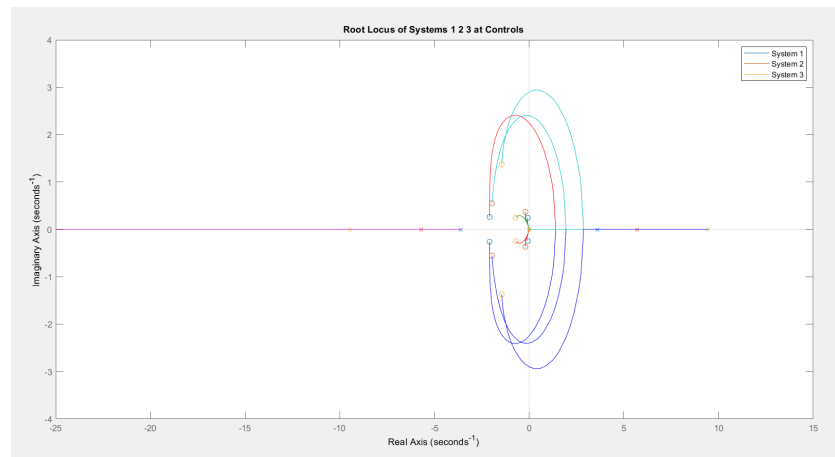
Figure 3.6.5: Root Locus At Controls

Varying Outer Loop Gain g_o with Inner Loop Gain g_i Fixed at $g_i = 20$

Time Domain Analysis: Impulse S_o, S_u, T_o, T_u
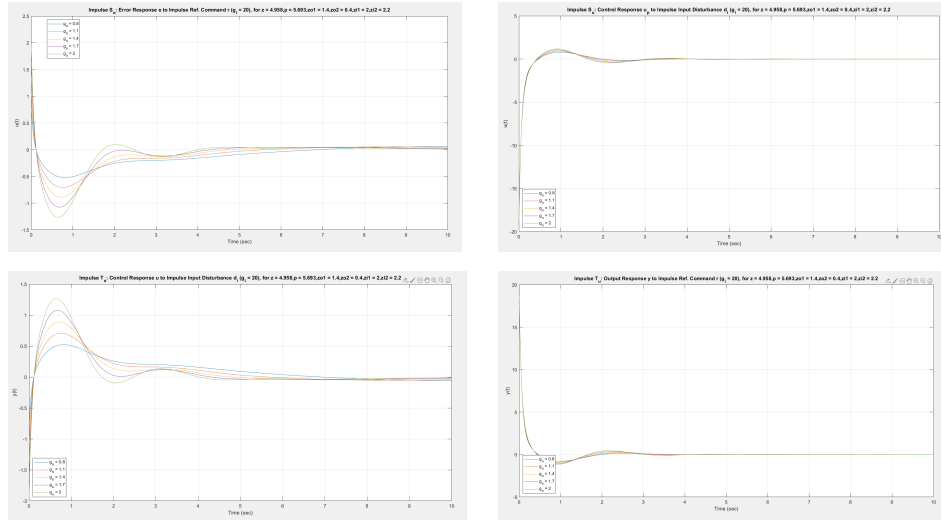
Figure 3.6.6: Impulse S_o, S_u, T_o, T_u $(g_i = 20)$

Time Domain Analysis: Step S_o, S_u, T_o, T_u
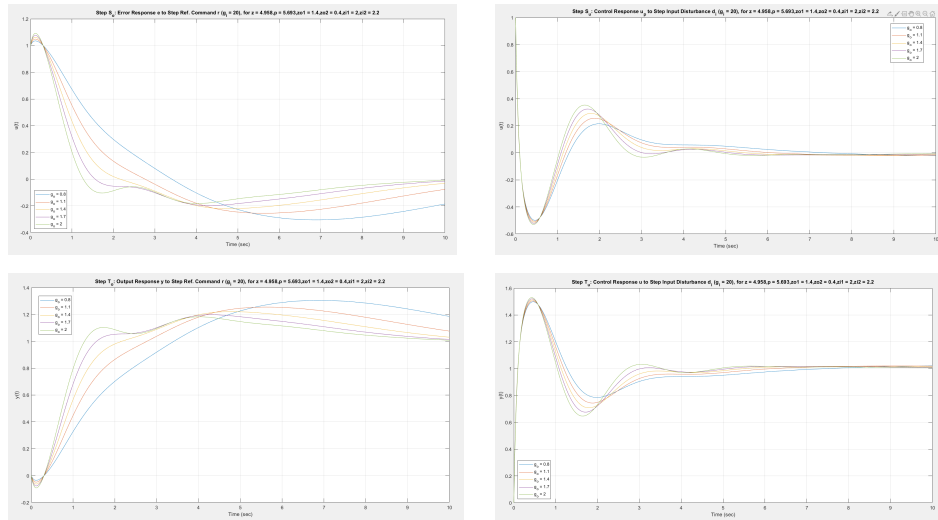
Figure 3.6.7: Step S_o, S_u, T_o, T_u $(g_i = 20)$

Varying Outer Loop Gain g_o **with Inner Loop Gain** g_i **Fixed at** $g_o = 2$

Time Domain Analysis: Impulse S_o, S_u, T_o, T_u
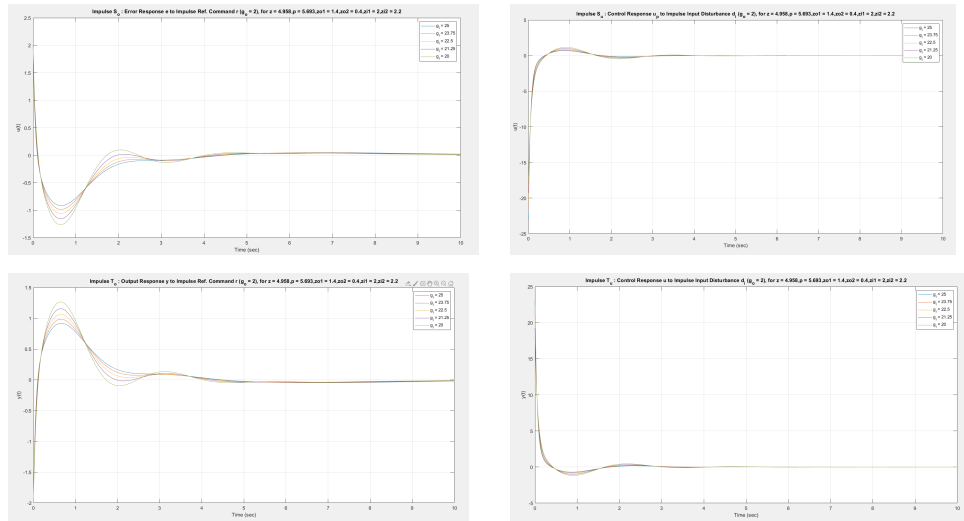
Figure 3.6.8: Impulse S_o, S_u, T_o, T_u $(g_o = 2)$

Time Domain Analysis: Step S_o, S_u, T_o, T_u
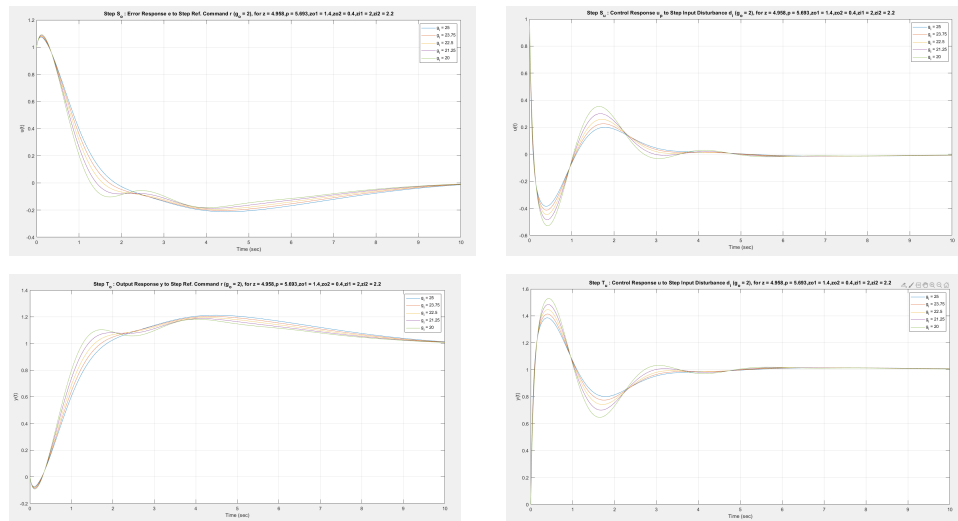
Figure 3.6.9: Step S_o, S_u, T_o, T_u $(g_o = 2)$

- Zero steady state error due to integrator in K_o.

Varying Outer Loop Gain g_o with g_i fixed at $g_i = 20$

Loop at Input:
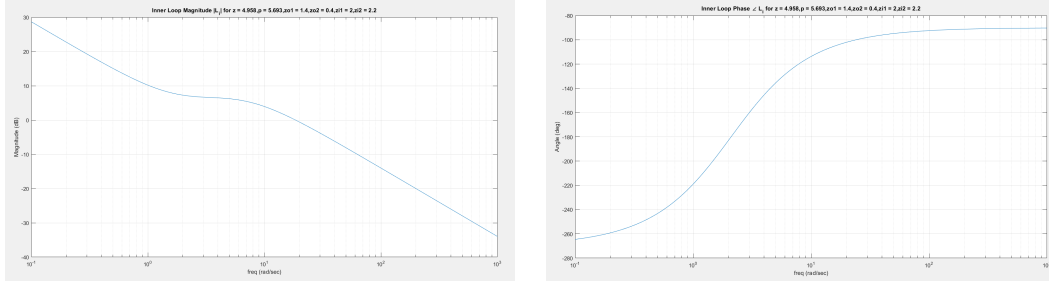
Figure 3.6.10: Inner Loop Magnitude $|L_i|$ and Phase $\angle L_i$

Loop at Output and Loop at Controls:
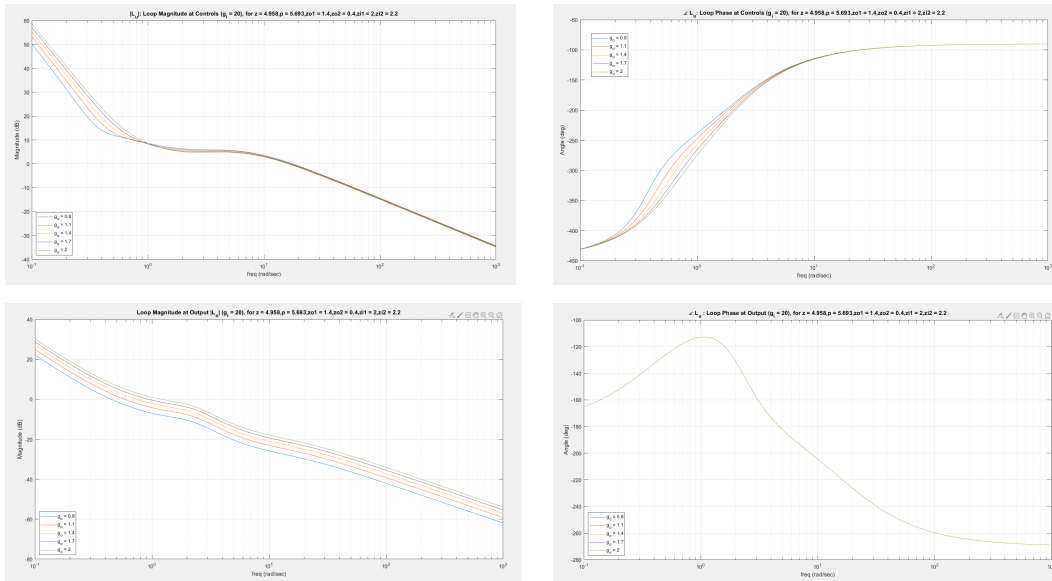
Figure 3.6.11: L_u and L_o: Loop Magnitude and Loop Phase ($g_i = 20$) for $z = 5$

- $|L_o|$ increases with increasing g_o, ω_{go} decreases with increasing g_o, $\angle L_o$ is independent of g_o, PM is independent of g_o, PM decreases with increasing g_o.

Varying Outer Loop Gain g_i with g_o fixed at $g_o = 2$

Loop at Input:
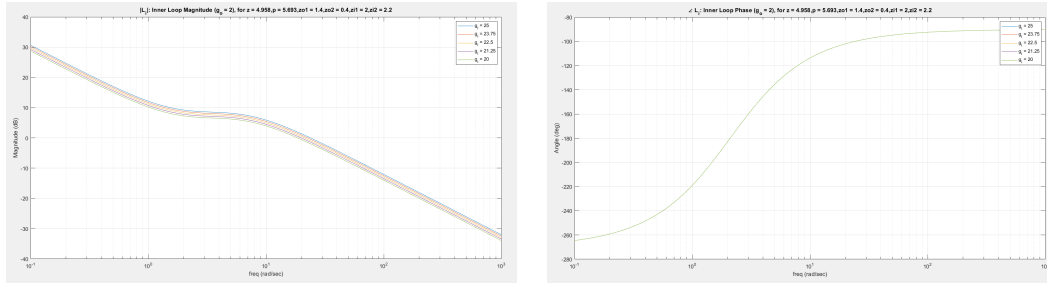
Figure 3.6.12: Inner Loop Magnitude $|L_i|$ and Phase $\angle L_i$

Loop at Output and Loop at Controls:
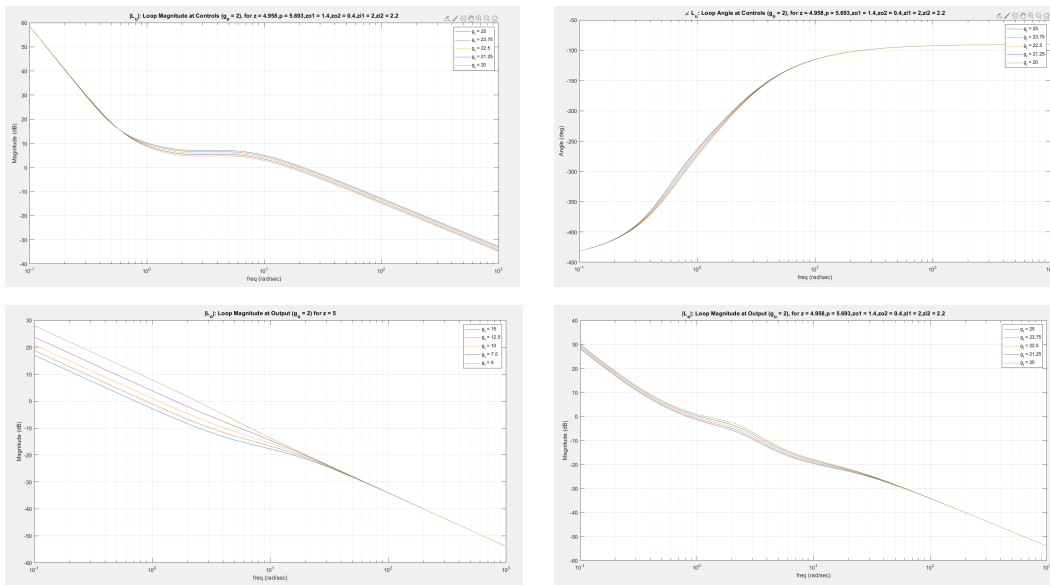
Figure 3.6.13: L_u and L_o: Loop Magnitude and Loop Phase $(g_o = 2)$ for $z = 5, 2.5, 1$

- $|L_o|$ increases with increasing g_o, ω_{go} decreases with increasing g_o, $\angle L_o$ is independent of g_o, PM is independent of g_o, PM decreases with increasing g_o.

131

Varying Outer Loop Gain g_o with Inner Loop Gain g_i Fixed at $g_i = 20$

Inner Loop Sensitivity and Complementary Sensitivity
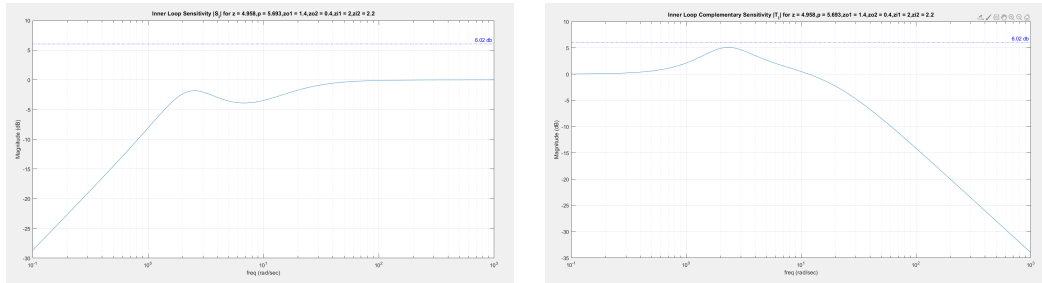
Figure 3.6.14: Inner Loop Sensitivity and Complementary Sensitivity $|S_i|$ and $|T_i|$

Loop Sensitivity at Output and at Controls
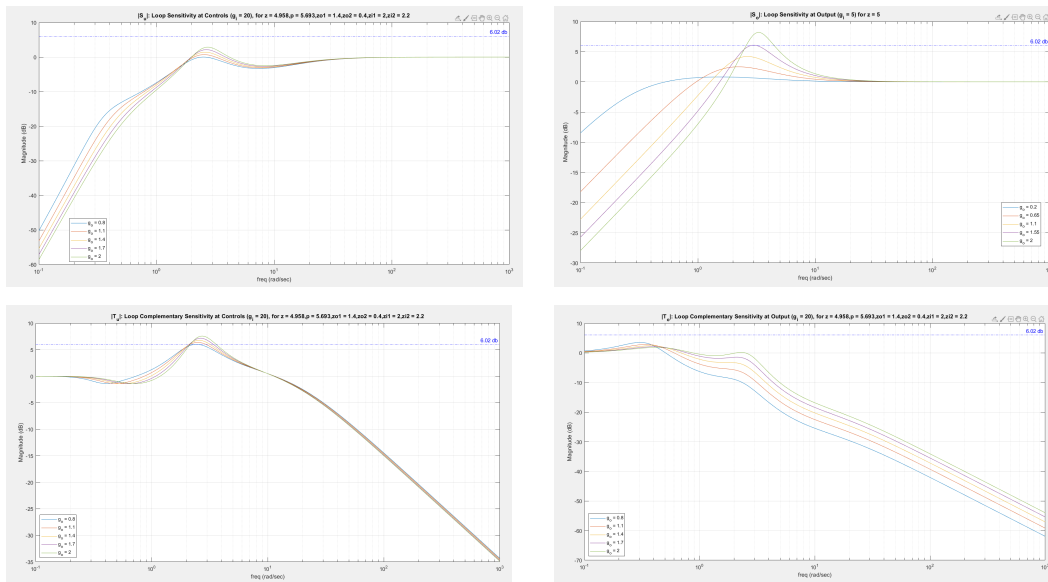
Figure 3.6.15: $|S_o|$ and $|S_u|$: Loop Sensitivity at Output and at Controls ($g_i = 20$)

- Peak $|S_o|$, $|S_u|$, $|T_o|$, $|T_u|$ increases with increasing g_o or decreasing g_i.

Varying Outer Loop Gain g_i with g_o fixed at $g_o = 2$

Inner Loop Sensitivity and Complementary Sensitivity
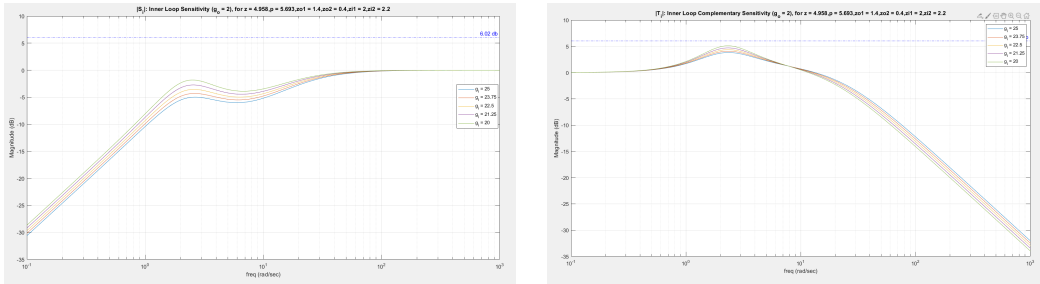
Figure 3.6.16: $|S_i|$ and $|T_i|$: Inner Loop Sensitivity and Complementary Sensitivity

$$(g_o = 2)$$

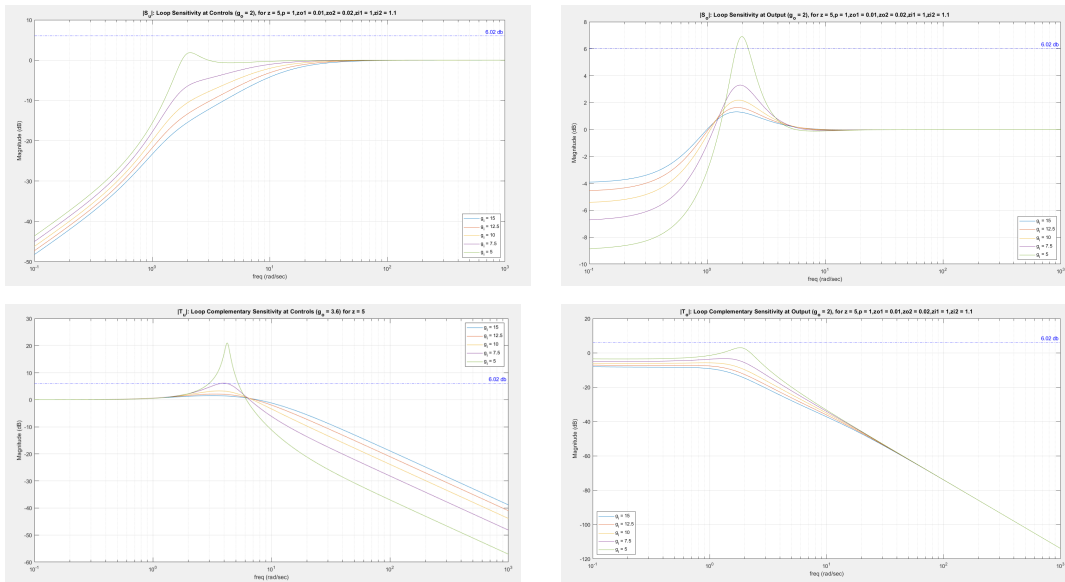**Loop Sensitivity and Complimentary Sensitivity at Output and at Controls**
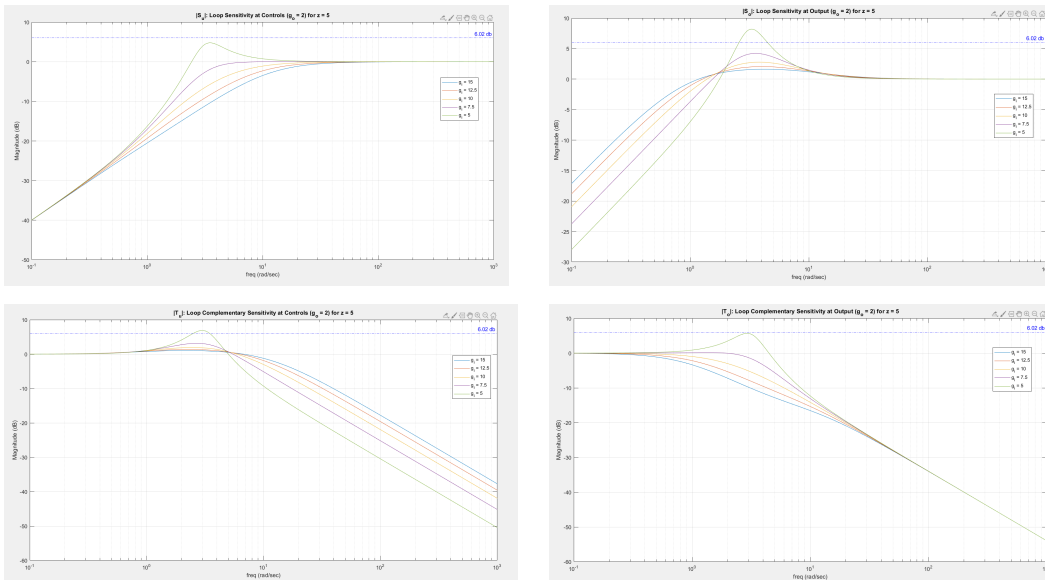
Figure 3.6.17: $|S_u|$, $|S_o|$, $|T_u|$, $|T_o|$: Loop Sensitivity at Output and at Controls

$$(g_o = 2)$$

- Peak $|S_o|$, $|S_u|$, $|T_o|$, $|T_u|$ increases with increasing g_o or decreasing g_i.

133

Varying Outer Loop Gain g_i with g_o fixed at $g_o = 2$

Peak $|S_o|$, $|S_u|$, $|T_o|$, $|T_u|$
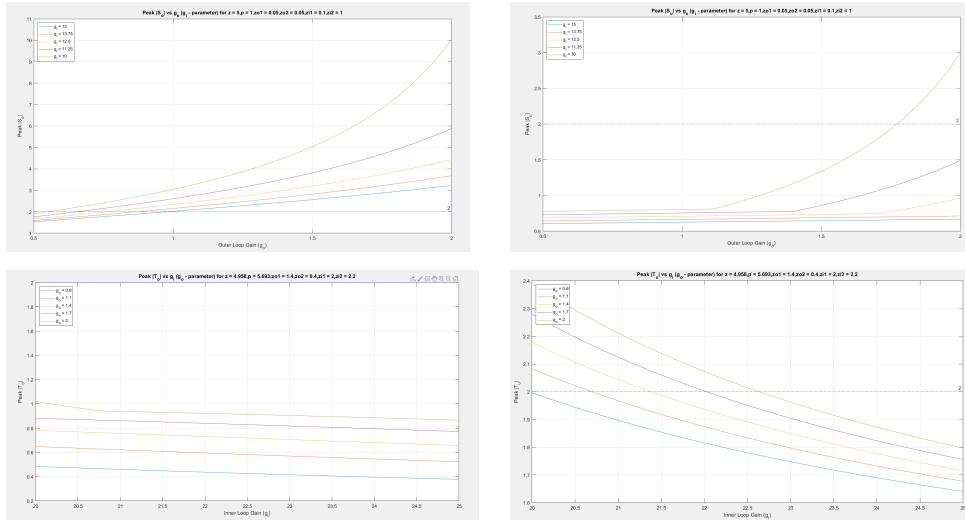
Figure 3.6.18: Peak $|S_o|$, $|S_u|$, $|T_o|$, $|T_u|$ vs g_o (g_i-parameter)

Varying Outer Loop Gain g_o with g_i fixed at $g_i = 20$
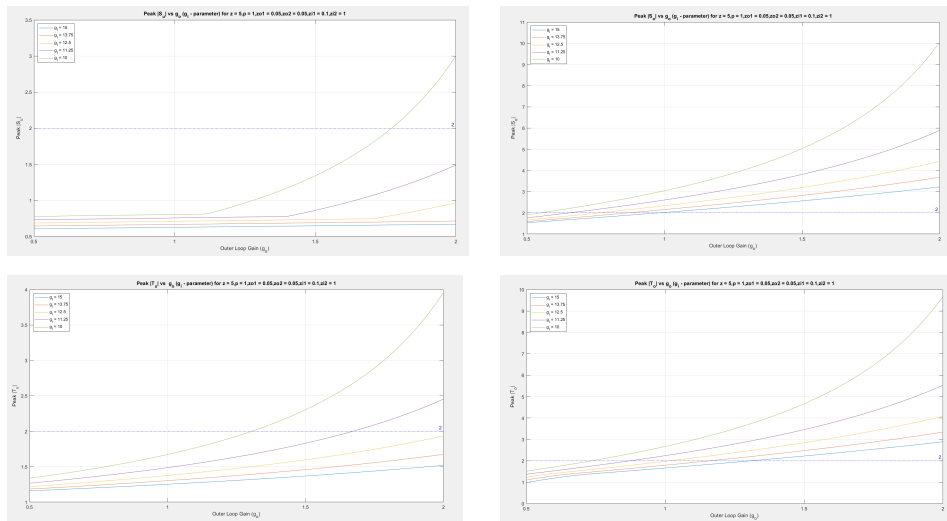
Figure 3.6.19: Peak $|S_u|$, $|S_o|$, $|T_u|$, $|T_o|$ vs g_o (g_i-parameter)

134

3.6.4   Fundamental Performance Limitations: A Prelude to Control Design

From the conclusions above, the parameters for proportional integral derviative design at input and at output can be implemented on the three systems which have been selected, namely,

- **System 1: Hernandez Model**

- **System 2: Ogata Model**

- **System 3: Vignesh Model**

The bandwidth of each of these systems are limited. However it needs to be computed for each of these systems to have intuitive information about each of the bandwidth at controls and at output. For comparing the bandwidth, it is necessary to fix the sensitivities at the input and the output. Hence, a sensitivity of 6 dB is chosen for sensitivity at controls and 0 dB at output.

Comparison of sensitivity at the Controls:
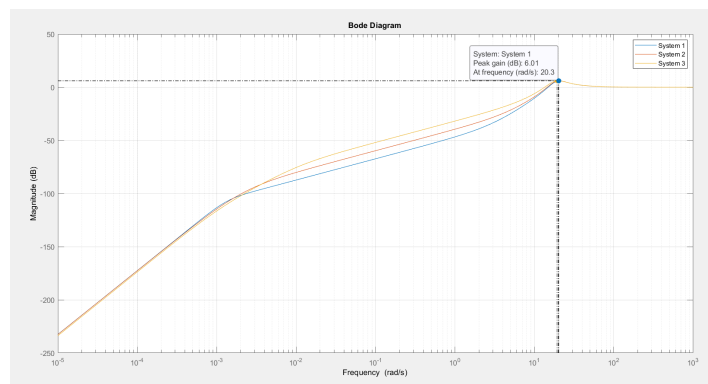
Figure 3.6.20: Comparison of sensitivity at the Controls

Comparison of sensitivity at the Output:
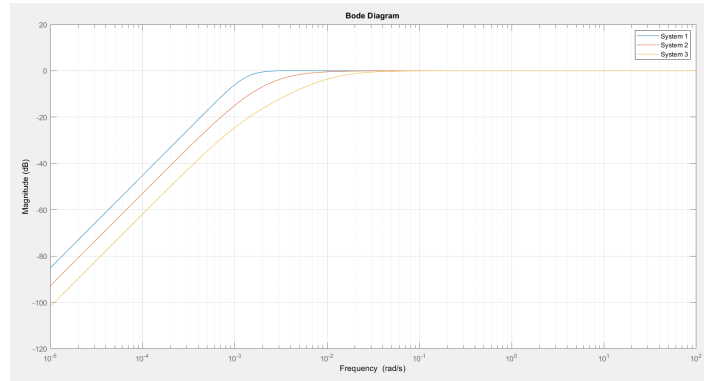
Figure 3.6.21: Comparison of sensitivity at the Output

The parameters for each of these systems for proportional integral derivative design at controls and at output are as below:

Systems	g_i	z_{i1}	z_{i2}	g_o	z_{o1}	z_{o2}
$System1$	17.65	14	12.2	0.02	0.001	28
$System2$	18.35	13.75	12.22	0.05	0.001	28.2
$System3$	20.65	13.5	12.3	0.08	0.001	29.2

Table 3.6.4: PID Design Parameters With Above Requirements

Using the above design parameters, the following ω_g at controls, ω_g at output, $\downarrow GM$ at controls are obtained.

Systems	ω_{gsu}	ω_{gso}	$\downarrow GM$
$System1$	14.9	0.003	0.3984
$System2$	14.4	0.2	0.4237
$System3$	13.24	0.51	0.4822

Table 3.6.5: System Insights

136

Comparison of Step Sensitivity at the Controls:
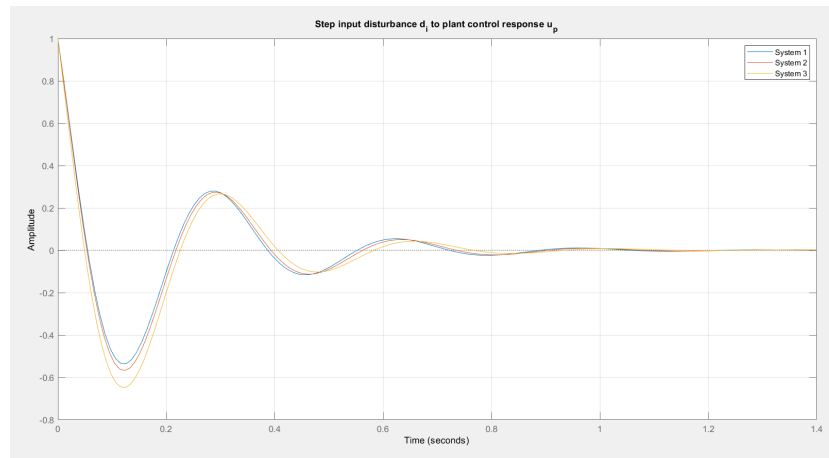
Figure 3.6.22: Comparison of Step sensitivity at the Controls

Comparison of Step sensitivity at the Output:
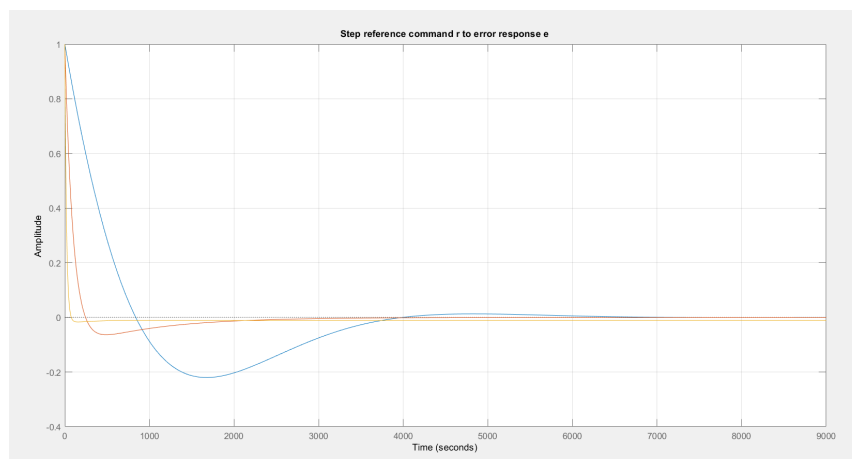
Figure 3.6.23: Comparison of sensitivity at the Output

- The peak step response for shortest pendulum (largest $\downarrow GM$) is the shortest. The peak sensitivity at the output is taken to be about 0 dB and the peak sensitivity at the controls is taken to be 6dB. Using the Zeigler Nichols initiated PID search, the following sensitivities and step responses are computed.

137

3.7   Summary and Conclusions

For comparing the systems at input and at controls, it is essential to compute the bandwidth at controls and at output while the systems have the same peak sensitivity at the inner loop and the outer loop. The peak sensitivity at the output is taken to be about 0 dB and the peak sensitivity at the controls is taken to be 6dB. Using the Zeigler Nichols initiated PID search, the following sensitivities and step responses are computed. The parameters involved with this search are given on table **??**. On computing this, the following results are found.

At Controls: The aforementioned results depict that the shortest pendulum (system 3) requires smallest bandwidth (ω_{gu}) at input to stabilize. It requires maximum u - bandwidth (ω_{gu}) for stabilization. Meanwhile, the longest pendulum (system 1) requires largest bandwidth (ω_{gu}) at controls to stabilize. It requires maximum u - bandwidth (ω_{gu}) for stabilization.

At output: Furthermore, system 3 has the largest x-bandwidth (ω_{go}). Therefore it is easiest to control. In contrast, the longest pendulum (system 1) requires smallest bandwidth (ω_{go}) at output for stabilization. As a consequence, it has the smallest x-bandwidth (ω_{go}). Therefore it is the hardest to control.

Using some non-linear concepts at play, it is found that the longest pendulum is easiest one to saturate since it has the smallest $\downarrow GM$. Comparing the control energies by showing step responses. The peak step response for shortest pendulum $\downarrow GM$ is the shortest. On saturating too much, the closed loop system is bound to go unstable. From a saturation point of view, the least likely to saturate is the longest pendulum. However, if the actuator saturation is small, then the system 3 (shortest pendulum) will be hardest to control, since there is only a short range of movement allowed which does not give much room for the shortest pendulum for stabilization.

Chapter 4

CONTROL DESIGN FOR NON-LINEAR CART INVERTED PENDULUM
SYSTEM

4.1 Motivation: Fundamental Performance Limitations Resolved- Issues, Control Architectures and Trade-offs

From the previous chapter it is seen that the shortest pendulum requires largest bandwidth at input to stabilize and has the largest x-bandwidth (hardest to control!!). In contrast, the longest pendulum requires smallest bandwidth at input to stabilize and has the smallest x-bandwidth (easiest to control!!). Using some non-linear thinking: the longest pendulum is easiest one to saturate since it has the smallest $\downarrow GM$. Comparing the control energies by showing step responses. The peak step response for shortest pendulum $\downarrow GM$. On saturating too much, the closed loop system is unstable. From a saturation point of view, the least likely to saturate is the longest pendulum(Hu and Lin, 2001).

The analogies from the previous examples is insightful to do a search using the PID parameters at the input and at the output for obtaining some useful designs with good properties at the output and at the controls. In the following chapter, the disadvantages of conventional PK structure for control loops will be explained. Also different designs such as Linear Quadratic Regulator and some computed H_∞ designs will be used to deduce some good properties at the input and some decent properties at the output. Furthermore the PID control parameters will be used to relate to the optimal control designs obtained.

4.2 Hierarchical Inner-Outer Loop Control Architecture

The systems with right hand plane zeros and right hand plane poles have some issues that need to be addressed such as bandwidth limitations, non-minimum phase. These issue restrict the system from a $\downarrow GM$ of zero and $\uparrow GM$ of infinite. Additionally, more contribution is made in making the system increasingly difficult to control. To deal with these issues, a conventional inner-outer loop control architecture is used. The hierarchical inner-outer control architecture also known as the cascade control design serves a purpose to assist with the right hand plane zero associated with the system.
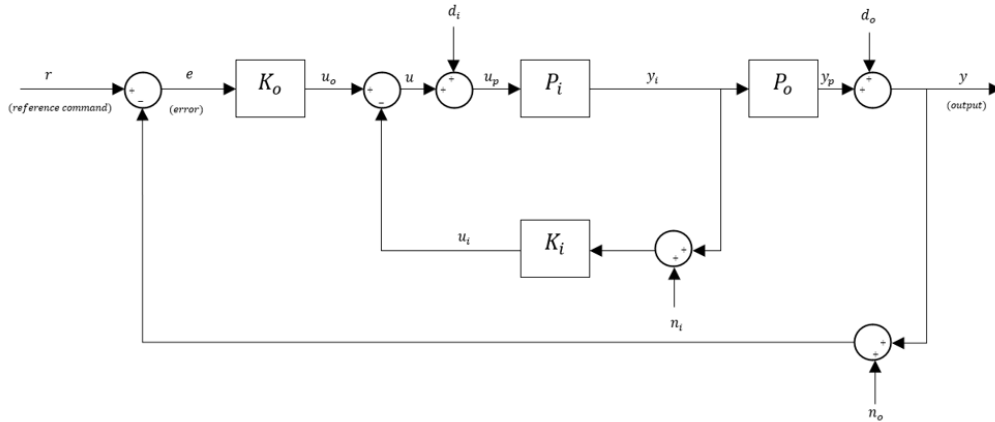
Figure 4.2.1: Block Diagram for hierarchical inner-outer loop control architecture

However, the need for this control architecture relies on the fact that the classical $P - K$ control architecture does not give us good properties. The system could be stabilized by using the conventional $P - K$ structure. However, the sensitivities at the output and at the input is not necessarily useful. It is observed that the sensitivities are very high at the input and at the controls. This design, therefore, cannot be used practically for implementation as it gives an immensely high overshoot which might harm the device and its actuators used.

4.3 Nominal Performance and Robustness Specifications

The nominal system has an instability associated with the system - the right hand plane pole. In addition to this the right hand plane zero does not let this instability to move to the left-half plane. A step response to the system depicts the system to be unstable. Hence, to address this issue, three control desigs will be used for comparison. To analyse the limitations of a design and relating the parameters to one another, this comparison is very useful.

- Design 1: A Traditional Single-Loop PK Architecture and Tradeoffs

- Design 2: Inner-Outer Loop Control Design Via LQR servo

- Design 3: Computed general H-infinity Mixed senstivity design for comparison

- Design 4: Inner-Outer Loop Control Design Via PID parameter search

The robustness specifications associated with each of these designs are:

Designs	S_u	S_o	T_u	T_o
$Design1$	25.7dB	25.7dB	25.8 dB	25.8dB
$Design2$	0dB	7 dB	0 dB	5.65dB
$Design3$	2dB	3dB	2.5 dB	1.5dB

Table 4.3.1: Robustness Of Aforementioned Designs

Using these designs and their respective sensitivities, the PID parameter search is computed. The following sensitivities at input and at output are related with the proportional integral derivative inner outer loop to match the sensitivities. On computing this, one can relate between the designs to get more insights about the PID inner-outer parameter search.

4.4   Design 1: A Traditional Single-Loop PK Architecture
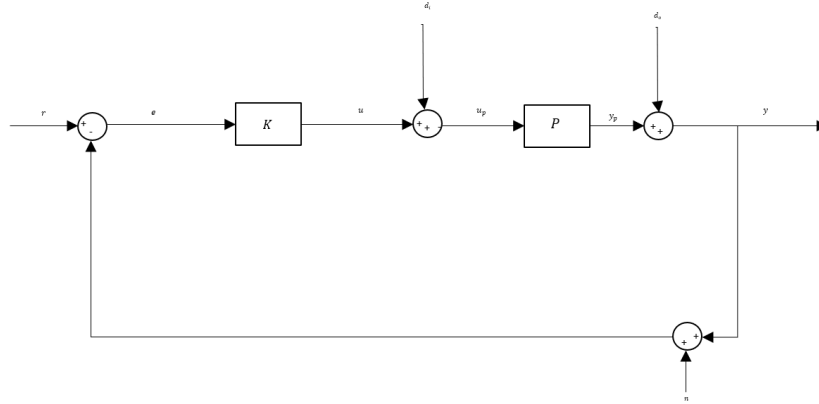
Figure 4.4.1: Block Diagram for P-K Structure

This tranditional $P - K$ structure uses a model based compensator using pole placement design. The plant is augmented to reject step disturbances at input and at the output. Using separation principle,the controller is designed. The model compensator poles corresponding to the contrller and the observer are chosen to make the system stable and detectable.

Separation Principle:

$$K = [A - BG - H(C - DG), H, G] \tag{4.4.1}$$

Loop Transfer at Input and Output:

$$L_i = PK; L_o = KP \tag{4.4.2}$$

Loop Senstivity and Complementary Sensitivity at Input

$$T_{d_i u_p} = S_i = [I + L_i]^{-1}; T_{d_i u} = T_i = L_i[I + L_i]^{-1} \tag{4.4.3}$$

Loop Senstivity and Complementary Sensitivity at Output

$$T_{re} = S_o = [I + L_o]^{-1}; T_{ry} = T_o = L_o[I + L_o]^{-1} \tag{4.4.4}$$

Sensitivity and Complimentary Sensitivity
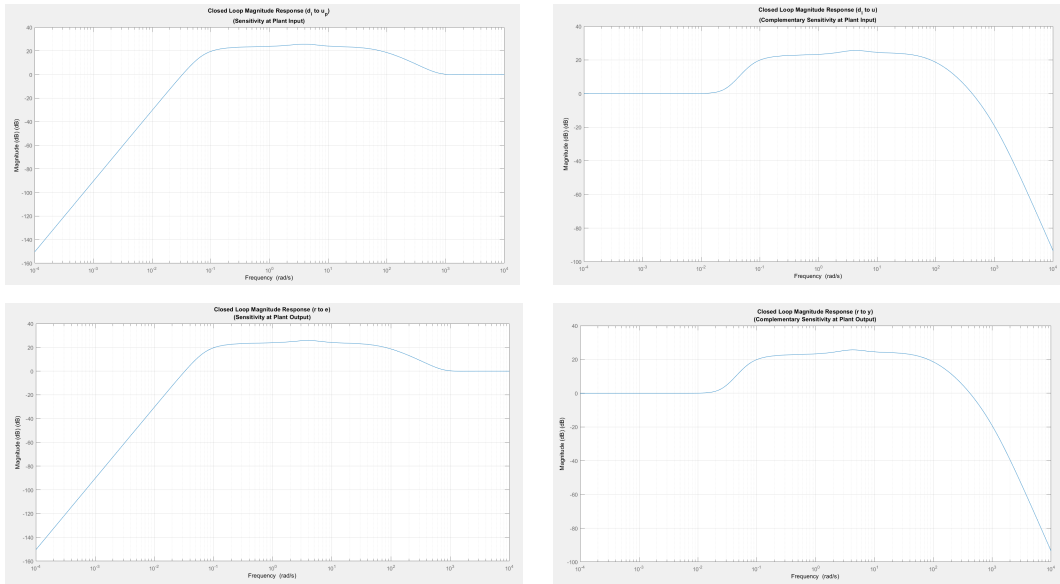
Figure 4.4.2: S_i, S_o, T_i, T_o

Step Responses
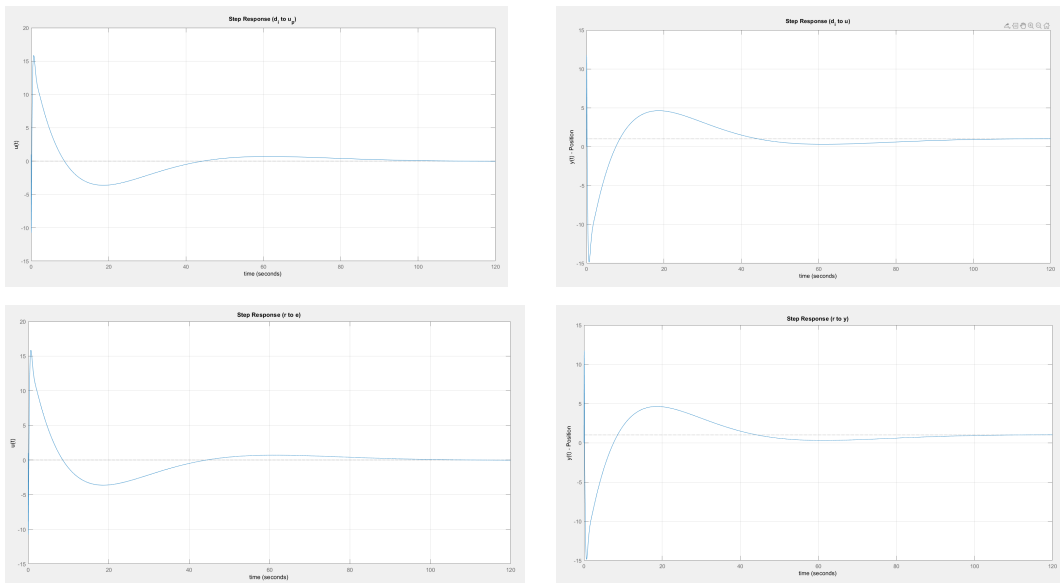
Figure 4.4.3: Step S_i, S_o, T_i, T_o

From these above figures 4.5.4, the following peak senstivites are obtained:

Designs	S_u	S_o	T_u	T_o
MBC	25.7dB	25.7dB	25.8 dB	25.8dB

Table 4.4.1: Model Based Compensator Peak Senstivities

- The peak sensitivites and step responses obtained depict that inspite of having good step responses the properties at the input and output are not useful. Considering the above results, the properties that the system depicts, are terrible.

- The conventional design does not give us good properties at the input and output because of the control issues associated withthe system (non-minimum phase, unstable).

- More precisely, using pole-placement also gives no information about the properties at the input and the output. It gives the position of poles with respect to the imaginary axis and real axis.

- Moreover, this type of design architecture has same properties at the input as well as on the output. Hence, tradeoffs are not involved with this control architecture. Additionally, the peak responses shows significant amount of saturation occurs on the input as well as output.

- To tackle with these issue, the control architecture demands more than one loop to address these control issues and help deal with them. As seen in the literature, this motivates the need of a new control architecture for getting useful properties at the input and output (inner-outer loop architecture). For this, it is necessary to use a design that gives good properties at the input and output.

144

4.5 Design 2: Inner-Outer Loop Control Design Via LQR servo
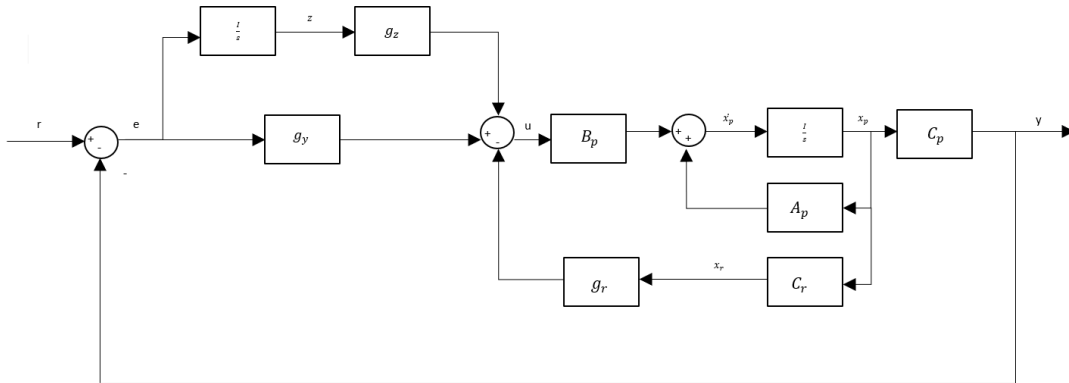
Figure 4.5.1: Block Diagram for hierarchial inner-outer loop LQR servo control
architecture

The LQR Servo uses an inner-outer loop control architecture as seen in figure
4.5.1. The cost function is minimized using:

LQR Problem Statement:

$$minJ(u) = \frac{1}{2} \int_0^\infty (x^T Q x + u^T R u) d\tau \qquad (4.5.1)$$

Solution:

$$u = -Gx \qquad (4.5.2)$$

Control Gain Matrix:

$$G = R^{-1} B^T K \qquad (4.5.3)$$

Control Algebraic Ricatti Equation:

$$0 = KA + A^T K + M^T M - KBR^{-1}B^T K \qquad (4.5.4)$$

Inner-Outer Loop Control Design Via LQR - PI

Sensitivity and Complimentary Sensitivity
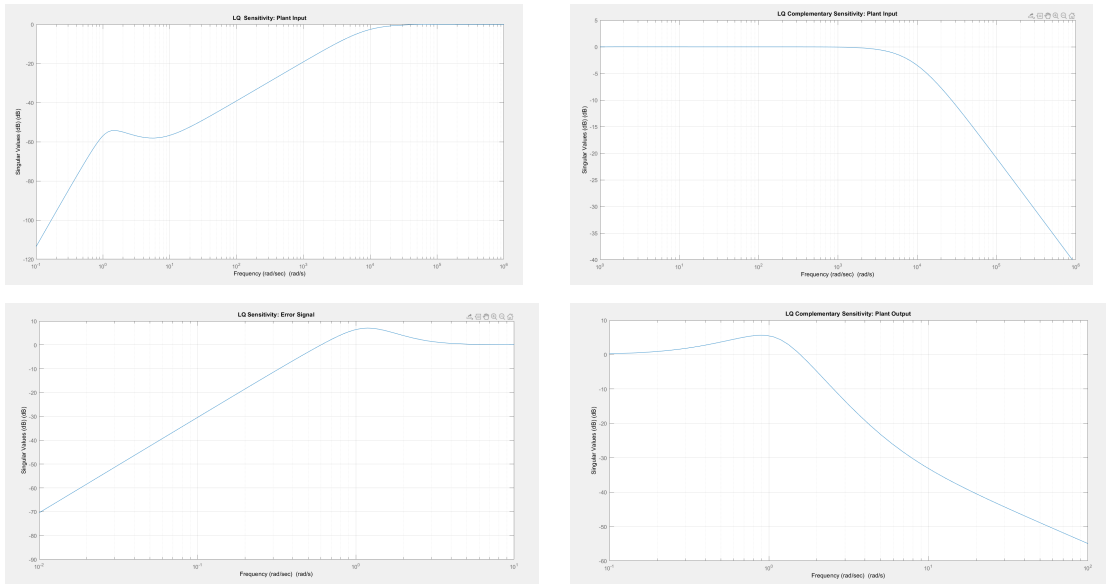
Figure 4.5.2: S_i, S_o, T_i, T_o

Step Responses
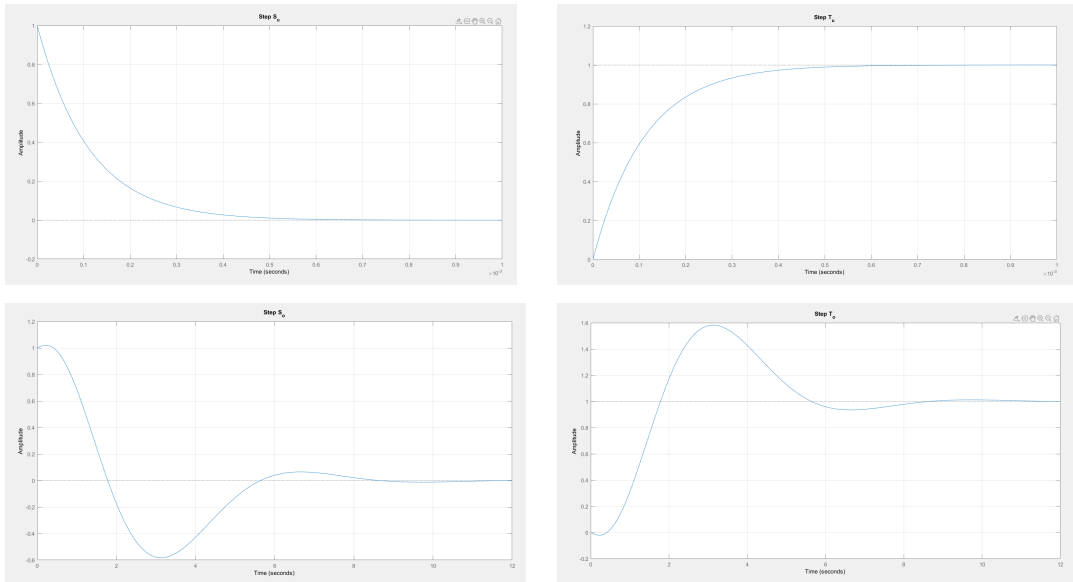
Figure 4.5.3: Step S_i, S_o, T_i, T_o

Inner-Outer Loop Control Design Via LQR - PID

Sensitivity and Complimentary Sensitivity
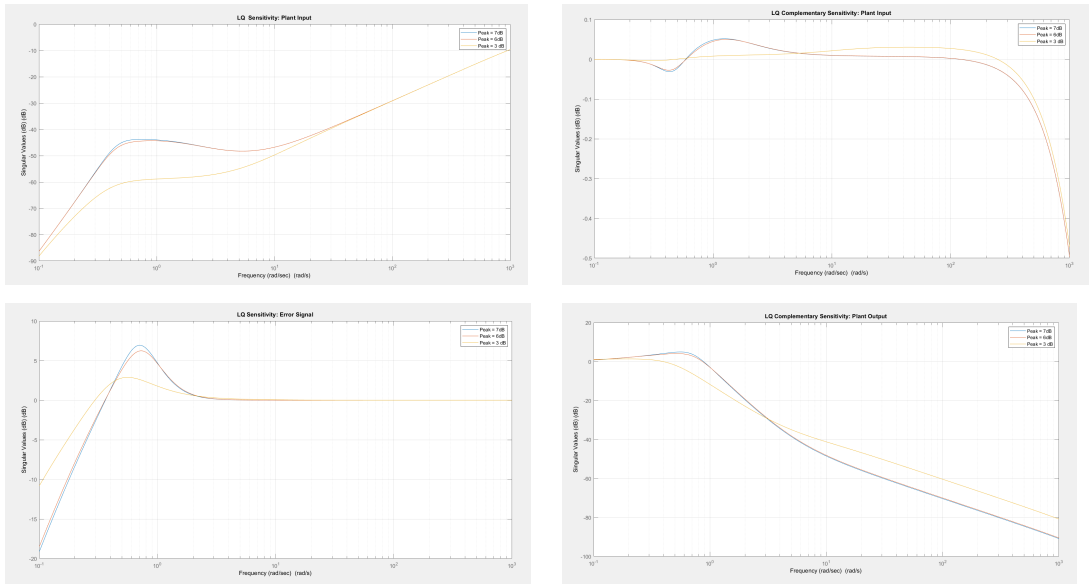
Figure 4.5.4: S_i, S_o, T_i, T_o

Step Responses
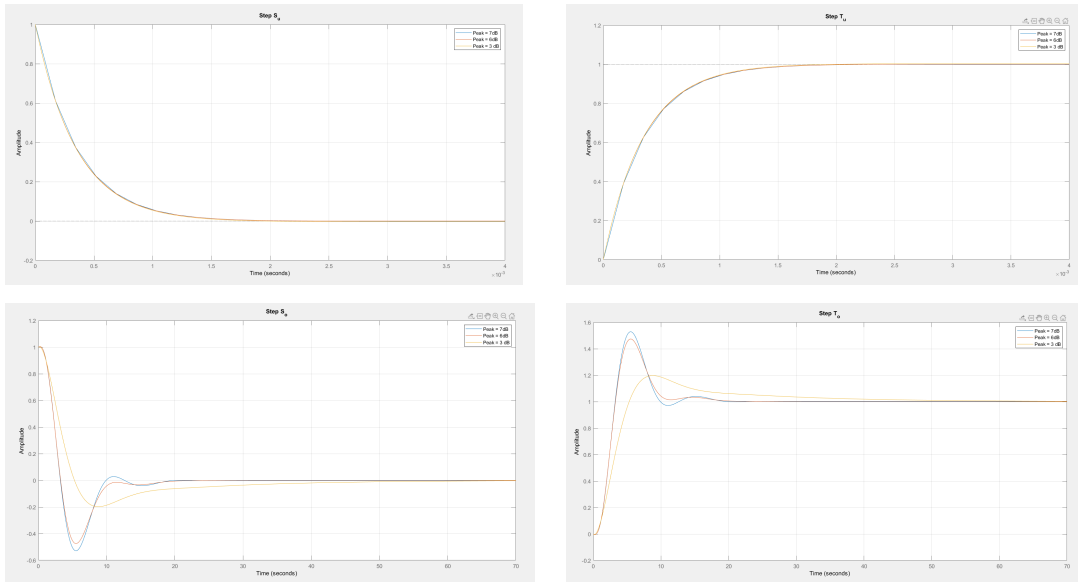
Figure 4.5.5: Step S_i, S_o, T_i, T_o

From these above figures, the following peak senstivites are obtained:

Designs	S_u	S_o
$LQR - PI$	0dB	7 dB
$LQR - PID$	0dB	7 dB
$LQR - PID$	0dB	6 dB
$LQR - PID$	0dB	3 dB

Table 4.5.1: LQR Peak Senstivities

- The peak sensitivites and step responses obtained depict that inspite of having good step responses the properties at the input the design has decent properties at output. However if more weights are placed on the state corresponding to the position, angle angular velocity, properties at output are improved.

- The LQR PI has been used to obtain good peak $|S_u|$ ($0dB$ as expected), unacceptable peak $|S_o|$ ($7dB$) which requires work (derivative action) to improve peak $|S_u|$ (J.-B. He et al., 2000).

- The LQR PID on the other hand, obtains good peak $|S_u|$ ($0dB$ as expected); acceptable peak $|S_o|$ ($3dB$) using diagonal Q and R.

- Hence, it is seen that the inner-outer loop structure contributes to lower the peak sensitivities of the inner loop and the outer loop. However, this structure does not have a PID in the inner loop as well as the outer loop (Prasad et al., 2014).

- Therefore, more parameters are not available for tradeoff. The Zeigler Nichols PID search involves a PID in the inner loop and the outer loop.

148

4.6 Design 3: Computed general H-infinity Mixed senstivity design for comparison
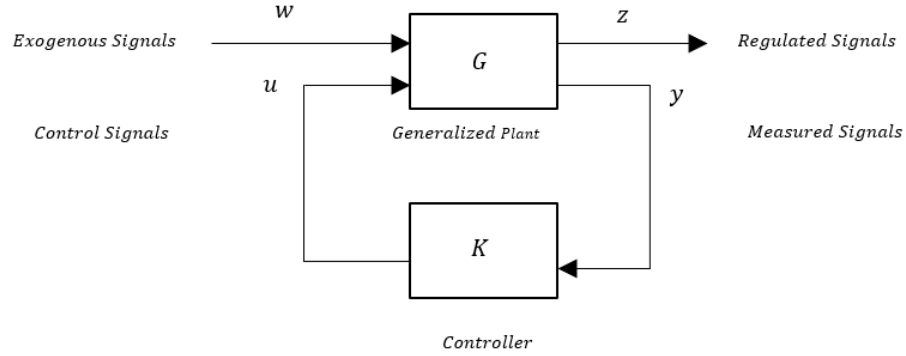
Figure 4.6.1: Generalized Feedback Structure

The $\mathcal{H}^\infty$ optimal controller satisfies a priori speicified peak performance bound γ $(\gamma > 0)$. The controller K stabilizes G such that the norm of the closed loop system $T_{wz}(K)$ satisfies:

Lower Bound on Peak Sensitivity

$$\|S\|_{\mathcal{H}^\infty} \geq \frac{\sqrt{\frac{g}{l}\left(1 + \frac{m}{M}\right)} + \sqrt{\frac{g}{l}}}{\sqrt{\frac{g}{l}\left(1 + \frac{m}{M}\right)} - \sqrt{\frac{g}{l}}} = \frac{\sqrt{1 + \frac{m}{M}} + 1}{\sqrt{1 + \frac{m}{M}} - 1} \tag{4.6.1}$$

$$\approx 10.5717 \quad (20.5\,\text{dB}) \tag{4.6.2}$$

Weighting Function W_1

$$W_1(s) = \frac{1}{M_s}\left[\frac{s + M_s\omega_b}{s + \epsilon\omega_b}\right] \tag{4.6.3}$$

Weighting Function W_2

Parameter	Value
M_s	10
ω_b	1
ϵ	0.1

Table 4.6.1: Numerical Choices For W_1 Design Parameters

$$W_2(s) = M_u \qquad (4.6.4)$$

Parameter	Value
M_u	0.1

Table 4.6.2: Numerical Choices For W_2 Design Parameters

Weighting Function W_3

$$W_3(s) = \left[\frac{s + \frac{\omega_{bc}}{M_y}}{\omega_{bc}} \right] \qquad (4.6.5)$$

Parameter	Value
M_y	10
ω_{bc}	1

Table 4.6.3: Numerical Choices For W_3 Design Parameters

Bilinear Transform Parameters.

$$\hat{s} = \frac{s - p_1}{1 - \frac{s}{p_2}} \qquad (4.6.6)$$

Parameter	Value
p_1	-0.1
p_2	-10^{-20}

Table 4.6.4: Numerical Choices For Bilinear Transform Parameters.

General Weighted $\mathcal{H}^\infty$ Suboptimal Mixed Sensitivity Problem:

$$\|T_{wz}(K)\|_{\mathcal{H}^\infty} = \left\| \begin{bmatrix} W_1 S \\ W_2 K S \\ W_3 T \end{bmatrix} \right\|_{\mathcal{H}^\infty} < \gamma \qquad (4.6.7)$$

$\mathcal{H}^\infty$-Optimal Controller K and Performance Measure γ_{opt}.

- $\gamma_{opt} = 5.5744e + 03$

- The weights W_1 , W_2 , W_3 are chosen for significantly influencing the sensitivity at controls, sensitivity at output and complementary sensitivity to address the closed loop stability, commmand following, disturbance attenuation, disturbance attenuation, sensor noise attenuation, reasonable control action.

- The weights are used to reduce the norms to a minimal which gives some reasonable gains for our controller to get good properties at the input, output and at the controls.

- This is achieved by using youla parameterization to convexify the problem. The convexified problem is optimized using convex optimization to get some good properties. The selected weights are chosen heuristically. The code for this design is used from my labmate Brent Wallace. I want to give credit to him for his work.

4.7 Design 4: Inner-Outer Loop Control Design Via PID parameter search

4.7.1 Zeigler Nichols Tuning
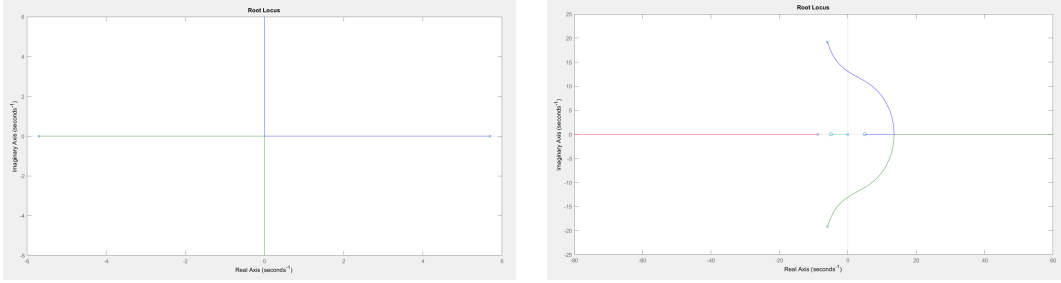
Figure 4.7.1: Root Locus of Inner Loop (Left) and Outer Loop (Right)

To tune the parameters of a pid controller of the inner loop, the inner plant is taken(Åström and Hägglund, 2004). As it is in the root locus in the above figure 4.7.1, the gain is increased till it reaches marginal stability. This gain is noted as K_p. The time constant T_u is the time of oscillations at marginal stability. The zeros of the inner loop are chosen to stabilize the inner plant P_i with the requirements mentioned in the table below:

Designs	K_p	T_i	T_d	$K_i = \frac{K_p}{T_i}$	$K_d = T_d K_p$
PID	$0.6K_u$	$\frac{T_u}{2}$	$\frac{T_u}{8}$	$1.2\frac{K_u}{T_u}$	$0.05K_u T_u$
Some overshoot	$0.7K_u$	$\frac{T_u}{2}$	$\frac{T_u}{3}$	$\frac{2}{3}\frac{K_u}{T_u}$	$\frac{1}{9}\frac{K_u}{T_u}$
No overshoot	$0.2K_u$	$\frac{T_u}{2}$	$\frac{T_u}{3}$	$\frac{2}{5}\frac{K_u}{T_u}$	$\frac{1}{15}\frac{K_u}{T_u}$

Table 4.7.1: Zeigler Nichols Tuning Method

The same method is applied for taking the outer loop system $K_i P_i [I + K_i P_i]^{-1} P_o$ to get the parameters for outer loop K_p, K_i, K_d. The outer loop zeros and inner loop zeros are found out corresponding to the parameters obtained above.

4.7.2   Proportional Integral Derivative Search

This design involves a blind search involving six for loops to get a family of designs corresponding to (Kp_i, Ki_i, Kd_i, Kp_o,Ki_o, Kd_o) for our design requirements.

Parameters for the search:

g_i	Proportional Inner Loop Gain
z_{i1}	Zero Inner Loop Gain
z_{i1}	Zero Inner Loop Gain
g_o	Proportional Outer Loop Gain
z_{o1}	Zero Outer Loop Gain
z_{o1}	Zero Outer Loop Gain

Table 4.7.2: PID Inner Outer Loop Parameter Definitions

This search mainly contributes to:

• Getting good properties at input and at controls.

• Relating this design to other designs achieved.

• Relating to various equilibrated designs for useful comparison.

The code for this search is given on the appendix. This search involves Zeigler Nichols tuning to choose the parameters used for stabilizing the loop. The parameters are chosen at the left and right of the parameter values corresponding to Zeigler Nichols design with a change of about 2 percent. The percentage is then changed to get required related designs.

LQR related pid search

From the aforementioned LQR design, the following peak sensitivities are obtained:

Designs	S_u	S_o
LQR	0dB	7 dB

Table 4.7.3: LQR Design Penstivities

The search is used with the following parameters corresponding to (Kp_i, Ki_i, Kd_i, Kp_o, Ki_o, Kd_o) given as below:

Design	g_i	z_{i1}	z_{i2}	g_o	z_{o1}	z_{o2}	ω_{gu}	ω_{go}	$\|S_u\|$	$\|S_o\|$
PID_{io}	50	2	2.2	12.17	0.5	1.6	35.9	2.4	0dB	7 dB

Table 4.7.4: PID Related LQR Design Parameters

Design	$\downarrow GM_u$	PM_u	$\uparrow GM_o$	PM_o
PID_{io}	0.33	82.42	2.97	32.85

Table 4.7.5: PID Related LQR Design Parameters

Sensitivity at Controls and at Output
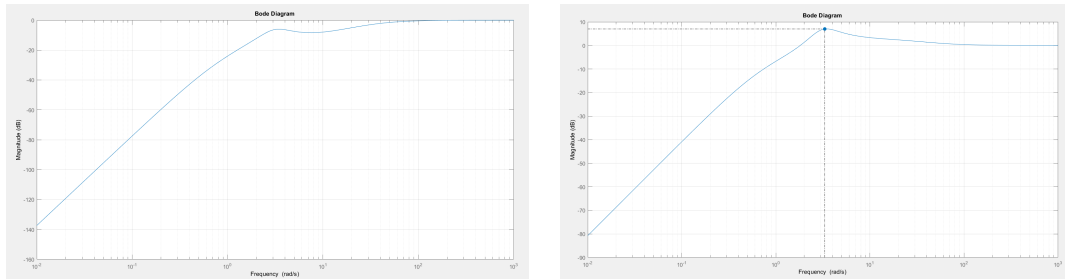
Figure 4.7.2: S_u, S_o to relate to LQR design

Relation to General weighted $\mathcal{H}^\infty$ mixed sensitivity design problem

From the aforementioned $\mathcal{H}^\infty$ design, the following peak sensitivities are obtained:

Designs	S_u	S_o
$\mathcal{H}^\infty$	2.8 dB	2.8 dB

Table 4.7.6: $\mathcal{H}^\infty$ Design Peak Sensitivities

The search is used with the following parameters corresponding to (Kp_i, Ki_i, Kd_i, Kp_o,Ki_o, Kd_o) given as below:

Design	g_i	z_{i1}	z_{i2}	g_o	z_{o1}	z_{o2}	ω_{gu}	ω_{go}	$\|S_u\|$	$\|S_o\|$
PID_{io}	17.85	2	2.2	0.95	0.4	1.4	15.3	0.57	2.8dB	2.8 dB

Table 4.7.7: PID Related $\mathcal{H}^\infty$ Design Parameters

Design	$\downarrow GM_u$	PM_u	$\uparrow GM_o$	PM_o
PID_{io}	0.5	74	6	59

Table 4.7.8: PID Related $\mathcal{H}^\infty$ Design Parameters

Sensitivity at Controls and at Output
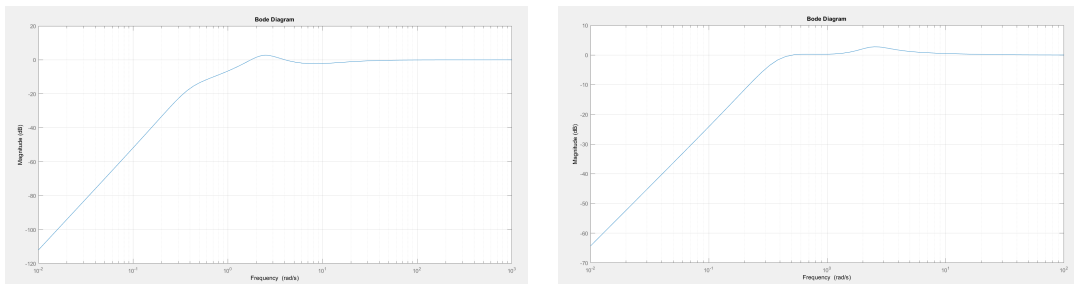
Figure 4.7.3: S_u, S_o to relate to $\mathcal{H}^\infty$ design

Relation to System 3 Design

From the system 3 design, the following peak sensitivities are obtained:

Designs	S_u	S_o
$\mathcal{H}^\infty$	7 dB	0 dB

Table 4.7.9: System 3 Design Peak Sensitivities

The search is used with the following parameters corresponding to (Kp_i, Ki_i, Kd_i, Kp_o, Ki_o, Kd_o) given as below:

Design	g_i	z_{i1}	z_{i2}	g_o	z_{o1}	z_{o2}	ω_{gu}	ω_{go}	$\|S_u\|$	$\|S_o\|$
PID_{io}	16.95	14	12.2	0.02	0.001	28	18.4	0	7dB	0 dB

Table 4.7.10: PID Related System 3 Design Parameters

Design	$\downarrow GM_u$	PM_u	$\uparrow GM_o$	PM_o
PID_{io}	0.5	28	255	78

Table 4.7.11: PID Related System 3 Design Parameters

Sensitivity at Controls and at Output
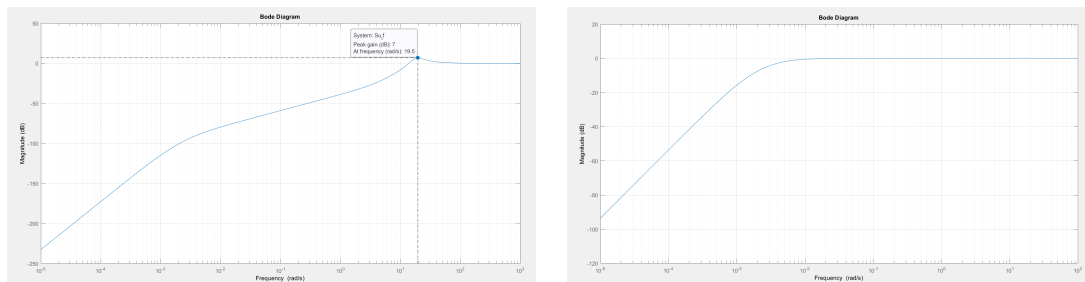
Figure 4.7.4: S_u, S_o to relate to system 3 design

Relation to System 1 Design

From the system 1 design, the following peak sensitivities are obtained:

Designs	S_u	S_o
$\mathcal{H}^\infty$	12 dB	0 dB

Table 4.7.12: System 1 Design Peak Sensitivities

The search is used with the following parameters corresponding to $(Kp_i, Ki_i, Kd_i, Kp_o, Ki_o, Kd_o)$ given as below:

Design	g_i	z_{i1}	z_{i2}	g_o	z_{o1}	z_{o2}	ω_{gu}	ω_{go}	$\|S_u\|$	$\|S_o\|$
PID_{io}	12.25	14	12.2	0.02	0.001	28	17.5	0	12dB	0 dB

Table 4.7.13: PID Related System 1 Design Parameters

Design	$\downarrow GM_u$	PM_u	$\uparrow GM_o$	PM_o
PID_{io}	0.6	17	115	81

Table 4.7.14: PID Related System 1 Design Parameters

Sensitivity and Complimentary Sensitivity
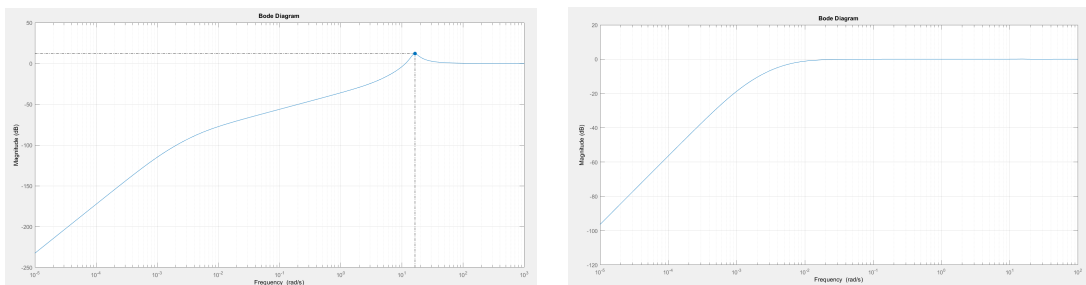
Figure 4.7.5: S_u, S_o to relate to system 1 design

4.7.3 Equilibrated Designs

The equilibrated designs pertain to sensitivities that are equilibrated at the input and at the controls. The procedure involves adjusting the parameters (Kp_i, Ki_i, Kd_i, Kp_o, Ki_o, Kd_o) such that the sensitivities at the controls and at the output are the same (pareto-optimal condition). Following this, if the bandwidth is fixed, adjusting the parameterst. Adjusting the gains depicts the sensitivity at controls increasing and sensitivity at output decreasing and vice versa.

The search is used with the following parameters corresponding to (Kp_i, Ki_i, Kd_i, Kp_o, Ki_o, Kd_o) given as below:

Examine Equilibrated PID Designs – Comparing Different Peaks

Parameter Set 1

g_i	z_{i1}	z_{i2}	g_o	z_{o1}	z_{o2}	ω_{gu}	ω_{go}	$\|S_u\|$	$\|S_o\|$
17.95	1.97	2.16	0.75	0.08	1	16	0.26	1.65dB	1.65 dB

ω_{gi}	ω_{gsu}	ω_{gso}	$\downarrow GM_u$	PM_u	$\uparrow GM_o$	PM_o
18	1.9	1.6	0.5	74	10	79

Parameter Set 2

g_i	z_{i1}	z_{i2}	g_o	z_{o1}	z_{o2}	ω_{gu}	ω_{go}	$\|S_u\|$	$\|S_o\|$
18.94	1.99	1.8	1.1	0.09	0.52	15.7	0.36	2.3dB	2.3 dB

ω_{gi}	ω_{gsu}	ω_{gso}	$\downarrow GM_u$	PM_u	$\uparrow GM_o$	PM_o
17.29	1.83	1.21	0.56	76	8	87.5

Parameter Set 3

g_i	z_{i1}	z_{i2}	g_o	z_{o1}	z_{o2}	ω_{gu}	ω_{go}	$\|S_u\|$	$\|S_o\|$
17.85	2	2.2	0.95	0.4	1.4	15	0.5	2.8dB	2.8 dB

ω_{gi}	ω_{gsu}	ω_{gso}	$\downarrow GM_u$	PM_u	$\uparrow GM_o$	PM_o
16.14	1.74	0.51	0.5	70	6	53

Parameter Set 4

16.94	2.01	2.37	1.7429	0.07	1.05	13.1	0.71	4.03dB	4.03 dB

ω_{gi}	ω_{gsu}	ω_{gso}	$\downarrow GM_u$	PM_u	$\uparrow GM_o$	PM_o
15.15	2.07	2.03	0.6	69	4.11	96

Parameter Set 5

17	2	2.2	3	0.4	1.4	11	2.73	12.2dB	12.2 dB

ω_{gi}	ω_{gsu}	ω_{gso}	$\downarrow GM_u$	PM_u	$\uparrow GM_o$	PM_o
15.19	2.08	2.13	0.8	65	1.67	19

- $\|S_u\|_{\mathcal{H}^\infty} \leq \|S_o\|_{\mathcal{H}^\infty}$

- As the peak increases, the open loop bandwidth at controls ω_{gu} decreases and the open loop bandwidth at output ω_{go} decreases.

- $\omega_{gi} \approx$ inner loop gain g_i

Examine Equilibrated PID Designs – Open Loop Plots
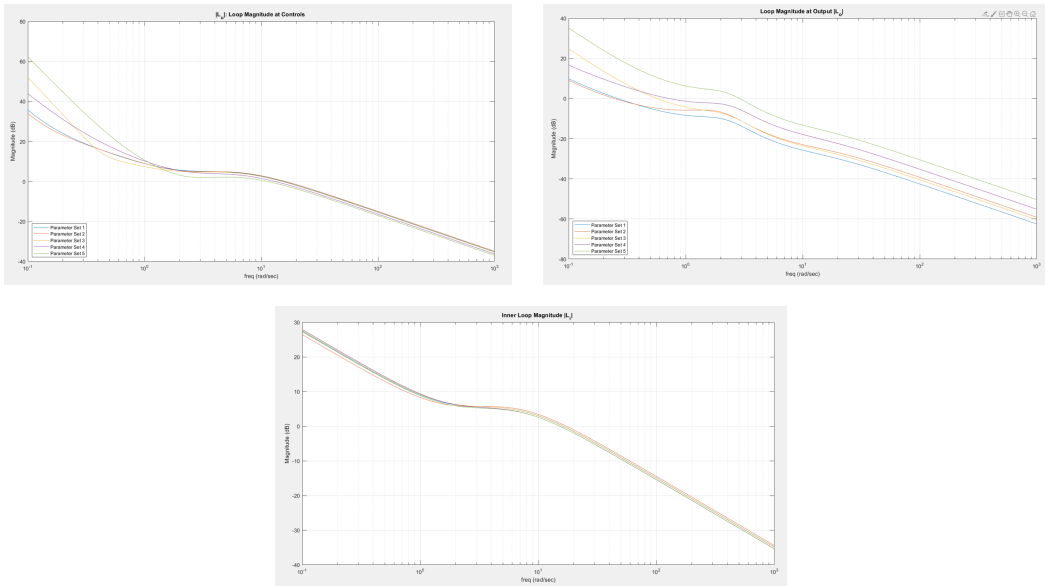
Figure 4.7.6: Examine Equilibrated PID Designs – Open Loop Plots

Examine Equilibrated PID Designs – Closed Loop Plots
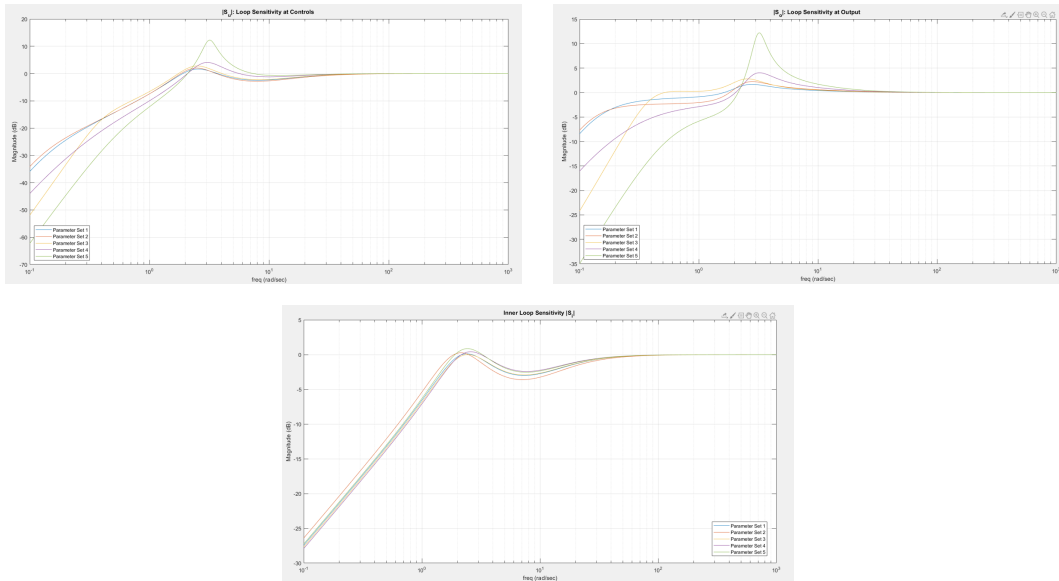
Figure 4.7.7: Examine Equilibrated PID Designs – Closed Loop Plots

Examine Equilibrated PID Designs – Comparing Same Peaks (2.8)

Parameter Set 1.1

g_i	z_{i1}	z_{i2}	g_o	z_{o1}	z_{o2}	ω_{gu}	ω_{go}	$\|S_u\|$	$\|S_o\|$
15.45	4.6	3.9	1.49	1.9	0.1	13.39	0.27	1.65dB	1.65 dB

ω_{gi}	ω_{gsu}	ω_{gso}	$\downarrow GM_u$	PM_u	$\uparrow GM_o$	PM_o
15.75	5.89	0.26	0.43	51.83	6	48.67

Parameter Set 1.2

g_i	z_{i1}	z_{i2}	g_o	z_{o1}	z_{o2}	ω_{gu}	ω_{go}	$\|S_u\|$	$\|S_o\|$
15.75	1.97	2.16	0.749	0.08	1	16	0.26	1.65dB	1.65 dB

ω_{gi}	ω_{gsu}	ω_{gso}	$\downarrow GM_u$	PM_u	$\uparrow GM_o$	PM_o
16.24	1.9	1.62	0.54	74.14	10.2	79.5

Parameter Set 1.3

g_i	z_{i1}	z_{i2}	g_o	z_{o1}	z_{o2}	ω_{gu}	ω_{go}	$\|S_u\|$	$\|S_o\|$
21.36	6.5	7	2.39	4.05	0.05	19.89	0.25	1.65 dB	1.65 dB

ω_{gi}	ω_{gsu}	ω_{gso}	$\downarrow GM_u$	PM_u	$\uparrow GM_o$	PM_o
21.91	11.8	1.87	0.31	49.51	6	78.47

Parameter Set 1.4

g_i	z_{i1}	z_{i2}	g_o	z_{o1}	z_{o2}	ω_{gu}	ω_{go}	$\|S_u\|$	$\|S_o\|$
27.9	0.9	1.487	0.475	0.02	0.568	26.22	0.19	1.65 dB	1.65 dB

ω_{gi}	ω_{gsu}	ω_{gso}	$\downarrow GM_u$	PM_u	$\uparrow GM_o$	PM_o
15.19	2.08	2.13	0.8	65	1.67	19

Parameter Set 1.5

g_i	z_{i1}	z_{i2}	g_o	z_{o1}	z_{o2}	ω_{gu}	ω_{go}	$\|S_u\|$	$\|S_o\|$
40	11	16.1	5.11	9.92	0.03	39.23	0.17	1.65 dB	1.65 dB

ω_{gi}	ω_{gsu}	ω_{gso}	$\downarrow GM_u$	PM_u	$\uparrow GM_o$	PM_o
43.28	25.98	1.14	0.23	48.94	5.78	79.97

Parameter Set 1.6

g_i	z_{i1}	z_{i2}	g_o	z_{o1}	z_{o2}	ω_{gu}	ω_{go}	$\|S_u\|$	$\|S_o\|$
47	11	21.1	6.29	9.92	0.03	46.5	0.14	1.65 dB	1.65 dB

ω_{gi}	ω_{gsu}	ω_{gso}	$\downarrow GM_u$	PM_u	$\uparrow GM_o$	PM_o
51.33	31.42	1.12	0.21	48.90	5.77	77.92

From these above tables, we get:

ω_{gu}	ω_{go}	$\frac{\omega_{go}}{\omega_{gu}}$	ω_{gi}
13.29	0.35	0.026	15.75
16	0.26	0.01	16.24
19.89	0.25	0.009	21.91
26.22	0.19	0.007	26.73
39.23	0.17	0.004	43.28
46.5	0.14	0.003	51.33

Table 4.7.15: Equilibrated Design Bandwidths

- The PID search using six for loops corresponding to each of the gain parameters involved in the controller, gives us designs related to every other design used in this thesis.

- As the peaks increases , open loop bandwidths - ω_{gu} drops and the ω_{go} rises up.

- Also the $\downarrow GM_u$ increases , $\uparrow GM_o$ decreases with rising peaks.

- Furthermore, the open loop bandwidth of the inner loop ω_{gi} is seen to be approximately equal to the inner loop gain g_i.

- The root locus helps in understanding the behavior of poles and zeros and how the peaks change with respect to them. The sensitivities associated depict the behavior of closed loops near the poles.

- The ratio decreases with respect to increasing ω_{gu}.

- The ratio is seen to be increasing with respect to increasing equilibrated peaks

Examine Equilibrated PID Designs – Open Loop Plots

Figure 4.7.8: Examine Equilibrated PID Designs – Open Loop Plots

Examine Equilibrated PID Designs – Closed Loop Plots
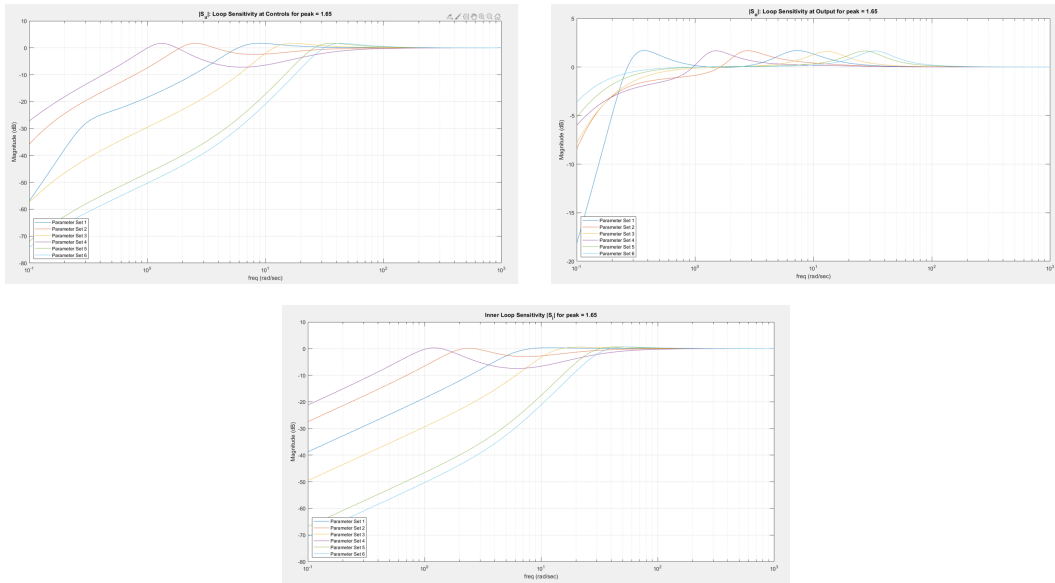
Figure 4.7.9: Examine Equilibrated PID Designs – Closed Loop Plots

Examine Equilibrated PID Designs – Comparing Same Peaks (2.3)

Parameter Set 2.1

g_i	z_{i1}	z_{i2}	g_o	z_{o1}	z_{o2}	ω_{gu}	ω_{go}	$\|S_u\|$	$\|S_o\|$
15.62	4.6	3.9	1.99	1.99	0.1	12.79	0.36	2.3 dB	2.3 dB

ω_{gi}	ω_{gsu}	ω_{gso}	$\downarrow GM_u$	PM_u	$\uparrow GM_o$	PM_o
13.13	5.95	2.7	0.43	52.07	6.37	71.5

Parameter Set 2.2

g_i	z_{i1}	z_{i2}	g_o	z_{o1}	z_{o2}	ω_{gu}	ω_{go}	$\|S_u\|$	$\|S_o\|$
18.738	1.94	1.8	1.1	0.09	0.52	15.7	0.357	2.3 dB	2.3 dB

ω_{gi}	ω_{gsu}	ω_{gso}	$\downarrow GM_u$	PM_u	$\uparrow GM_o$	PM_o
16.14	1.45	0.52	0.61	76.16	5.5	57.9

Parameter Set 2.3

g_i	z_{i1}	z_{i2}	g_o	z_{o1}	z_{o2}	ω_{gu}	ω_{go}	$\|S_u\|$	$\|S_o\|$
22	8.93	7.2	2.96	4.25	0.23	21.5	0.29	2.3 dB	2.3 dB

ω_{gi}	ω_{gsu}	ω_{gso}	$\downarrow GM_u$	PM_u	$\uparrow GM_o$	PM_o
23.58	8.6	1.8	0.46	46.069	1.71	14.95

Parameter Set 2.4

g_i	z_{i1}	z_{i2}	g_o	z_{o1}	z_{o2}	ω_{gu}	ω_{go}	$\|S_u\|$	$\|S_o\|$
28.9	0.8	1.48	0.57	0.09	0.56	27.15	0.282	2.3 dB	2.3 dB

ω_{gi}	ω_{gsu}	ω_{gso}	$\downarrow GM_u$	PM_u	$\uparrow GM_o$	PM_o
29.8	0.76	0.48	0.63	85	6.77	53.1

Parameter Set 2.5

g_i	z_{i1}	z_{i2}	g_o	z_{o1}	z_{o2}	ω_{gu}	ω_{go}	$\|S_u\|$	$\|S_o\|$
40	11	16.7	7	9.1	0.09	38.15	0.229	2.3 dB	2.3 dB

ω_{gi}	ω_{gsu}	ω_{gso}	$\downarrow GM_u$	PM_u	$\uparrow GM_o$	PM_o
43.2	24.16	2.45	0.26	42.6	3.5	67.77

Parameter Set 2.6

g_i	z_{i1}	z_{i2}	g_o	z_{o1}	z_{o2}	ω_{gu}	ω_{go}	$\|S_u\|$	$\|S_o\|$
47	11.95	21.1	8.29	11.5	0.2	45.24	0.2	2.3 dB	2.3 dB

ω_{gi}	ω_{gsu}	ω_{gso}	$\downarrow GM_u$	PM_u	$\uparrow GM_o$	PM_o
51.33	28.9	0.18	0.25	42.32	3.44	49.9

From these above tables, we get:

ω_{gu}	ω_{go}	$\frac{\omega_{go}}{\omega_{gu}}$	ω_{gi}
12.79	0.36	0.028	16.13
15.7	0.357	0.022	19.2
21.5	0.296	0.013	23.5
27.15	0.282	0.01	20.2
38.15	0.229	0.006	43.5
45.24	0.2	0.004	51.6

Table 4.7.16: Equilibrated Design Bandwidths

- The PID search using six for loops corresponding to each of the gain parameters involved in the controller, gives us designs related to every other design used in this thesis.

- As the peaks increases , open loop bandwidths - ω_{gu} drops and the ω_{go} rises up.

- Also the $\downarrow GM_u$ increases , $\uparrow GM_o$ decreases with rising peaks.

- Furthermore, the open loop bandwidth of the inner loop ω_{gi} is seen to be approximately equal to the inner loop gain g_i.

- The root locus helps in understanding the behavior of poles and zeros and how the peaks change with respect to them. The sensitivities associated depict the behavior of closed loops near the poles.

- The ratio decreases with respect to increasing ω_{gu}.

- The ratio is seen to be increasing with respect to increasing equilibrated peaks

Examine Equilibrated PID Designs – Open Loop Plots

Figure 4.7.10: Examine Equilibrated PID Designs – Open Loop Plots

Examine Equilibrated PID Designs – Closed Loop Plots
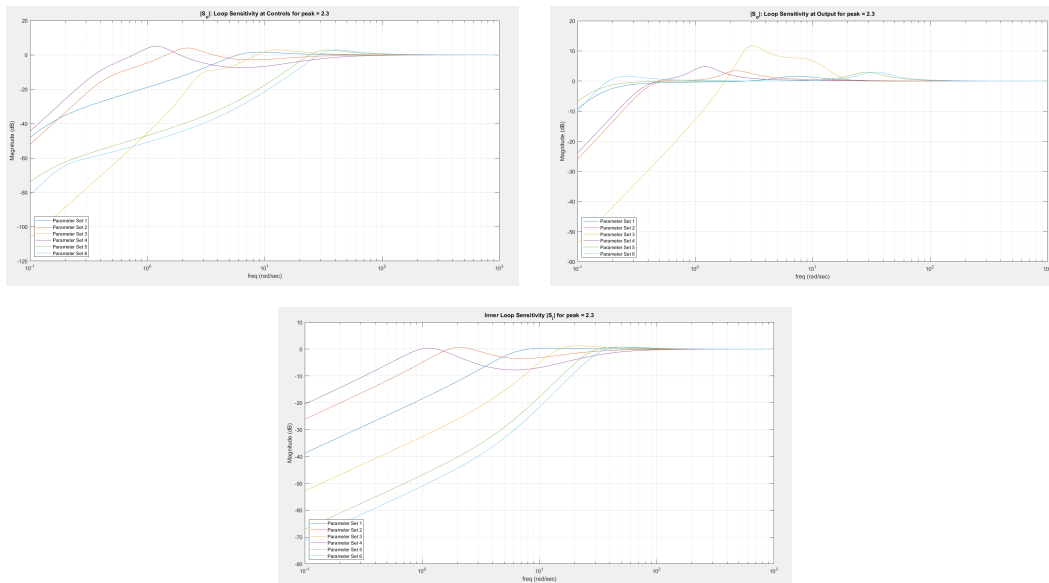
Figure 4.7.11: Examine Equilibrated PID Designs – Closed Loop Plots

Examine Equilibrated PID Designs – Comparing Same Peaks (2.8)

Parameter Set 3.1

g_i	z_{i1}	z_{i2}	g_o	z_{o1}	z_{o2}	ω_{gu}	ω_{go}	$\|S_u\|$	$\|S_o\|$
14.79	3.06	3.12	1.41	2	0.4	11.42	0.62	2.8 dB	2.8 dB

ω_{gi}	ω_{gsu}	ω_{gso}	$\downarrow GM_u$	PM_u	$\uparrow GM_o$	PM_o
13.13	2.98	0.6	0.54	74.16	10.2	9.5

Parameter Set 3.2

g_i	z_{i1}	z_{i2}	g_o	z_{o1}	z_{o2}	ω_{gu}	ω_{go}	$\|S_u\|$	$\|S_o\|$
17.85	2	2.2	0.949	0.4	1.4	15	0.569	2.8 dB	2.8 dB

ω_{gi}	ω_{gsu}	ω_{gso}	$\downarrow GM_u$	PM_u	$\uparrow GM_o$	PM_o
16.14	1.74	0.75	0.56	75.5	8.03	87.5

Parameter Set 3.3

g_i	z_{i1}	z_{i2}	g_o	z_{o1}	z_{o2}	ω_{gu}	ω_{go}	$\|S_u\|$	$\|S_o\|$
22.34	10.16	5.4	4.095	6.3	0.1	20.21	0.5319	2.8 dB	2.8 dB

ω_{gi}	ω_{gsu}	ω_{gso}	$\downarrow GM_u$	PM_u	$\uparrow GM_o$	PM_o
23.58	12.05	1.35	0.58	73.49	6.1	58.76

Parameter Set 3.4

g_i	z_{i1}	z_{i2}	g_o	z_{o1}	z_{o2}	ω_{gu}	ω_{go}	$\|S_u\|$	$\|S_o\|$
30.85	1	1.1	0.69	0.02	0.535	29.36	0.312	2.8 dB	2.8 dB

ω_{gi}	ω_{gsu}	ω_{gso}	$\downarrow GM_u$	PM_u	$\uparrow GM_o$	PM_o
29.8	0.93	0.95	0.61	69.04	4.11	96.423

Parameter Set 3.5

g_i	z_{i1}	z_{i2}	g_o	z_{o1}	z_{o2}	ω_{gu}	ω_{go}	$\|S_u\|$	$\|S_o\|$
40	11	16.1	8.71	2.92	0.03	37.55	0.09	2.8 dB	2.8 dB

ω_{gi}	ω_{gsu}	ω_{gso}	$\downarrow GM_u$	PM_u	$\uparrow GM_o$	PM_o
43.2	24.5	11.24	0.8	65.64	1.67	19.23

Parameter Set 3.6

g_i	z_{i1}	z_{i2}	g_o	z_{o1}	z_{o2}	ω_{gu}	ω_{go}	$\|S_u\|$	$\|S_o\|$
47	11	21.1	10.59	1.02	0.03	44.77	0.03	2.8 dB	2.8 dB

ω_{gi}	ω_{gsu}	ω_{gso}	$\downarrow GM_u$	PM_u	$\uparrow GM_o$	PM_o
51.33	29.71	0.02	0.8	65.64	1.67	19.23

From these above tables, we get:

ω_{gu}	ω_{go}	$\frac{\omega_{go}}{\omega_{gu}}$	ω_{gi}
11.42	0.62	0.05	13.13
15	0.569	0.037	16.14
20.21	0.531	0.026	23.58
28.3	0.312	0.011	29.8
37.55	0.09	0.002	43.2
44.77	0.03	0.0006	51.33

Table 4.7.17: Equilibrated Design Bandwidths

- The PID search using six for loops corresponding to each of the gain parameters involved in the controller, gives us designs related to every other design used in this thesis.

- As the peaks increases , open loop bandwidths - ω_{gu} drops and the ω_{go} rises up.

- Also the $\downarrow GM_u$ increases , $\uparrow GM_o$ decreases with rising peaks.

- Furthermore, the open loop bandwidth of the inner loop ω_{gi} is seen to be approximately equal to the inner loop gain g_i.

- The root locus helps in understanding the behavior of poles and zeros and how the peaks change with respect to them. The sensitivities associated depict the behavior of closed loops near the poles.

- The ratio decreases with respect to increasing ω_{gu}.

- The ratio is seen to be increasing with respect to increasing equilibrated peaks

Examine Equilibrated PID Designs – Open Loop Plots

Figure 4.7.12: Examine Equilibrated PID Designs – Open Loop Plots

Examine Equilibrated PID Designs – Closed Loop Plots
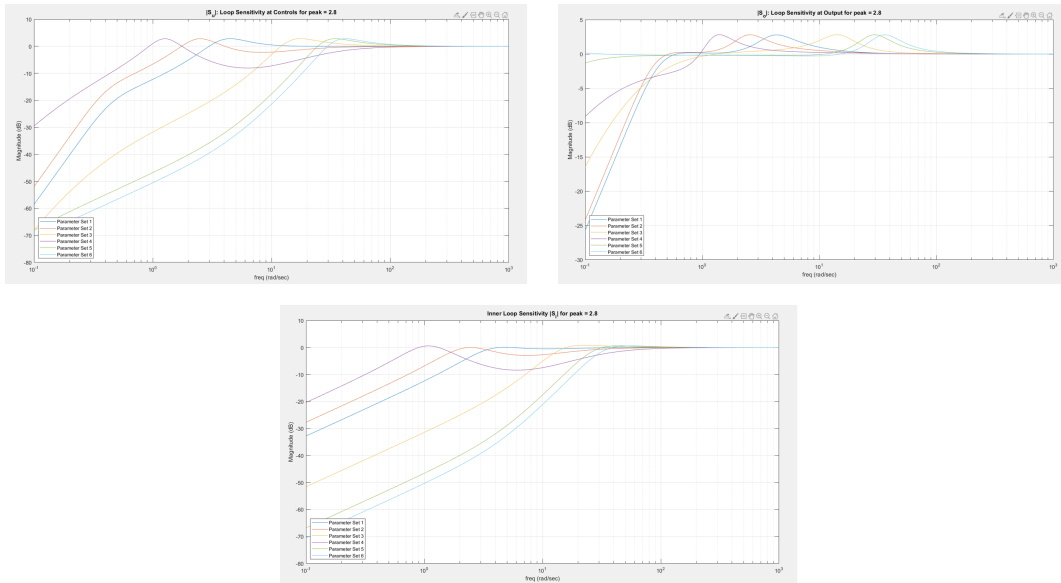
Figure 4.7.13: Examine Equilibrated PID Designs – Closed Loop Plots

Examine Equilibrated PID Designs – Comparing Same Peaks (4.03)

Parameter Set 4.1

g_i	z_{i1}	z_{i2}	g_o	z_{o1}	z_{o2}	ω_{gu}	ω_{go}	$\|S_u\|$	$\|S_o\|$
14.79	3.06	3.16	1.95	2	0.4	10.75	0.84	4.03 dB	4.03 dB

ω_{gi}	ω_{gsu}	ω_{gso}	$\downarrow GM_u$	PM_u	$\uparrow GM_o$	PM_o
13.13	2.92	1.8	0.59	54.59	3.17	67.01

Parameter Set 4.2

g_i	z_{i1}	z_{i2}	g_o	z_{o1}	z_{o2}	ω_{gu}	ω_{go}	$\|S_u\|$	$\|S_o\|$
16.94	2.01	2.37	1.749	0.07	1.05	13.1	0.71	4.03 dB	4.03 dB

ω_{gi}	ω_{gsu}	ω_{gso}	$\downarrow GM_u$	PM_u	$\uparrow GM_o$	PM_o
15.15	2.07	2.03	0.61	69.02	4.1	96.46

Parameter Set 4.3

g_i	z_{i1}	z_{i2}	g_o	z_{o1}	z_{o2}	ω_{gu}	ω_{go}	$\|S_u\|$	$\|S_o\|$
22.34	10.16	5.4	5.8	4.3	0.1	19.32	0.5	4.03 dB	4.03 dB

ω_{gi}	ω_{gsu}	ω_{gso}	$\downarrow GM_u$	PM_u	$\uparrow GM_o$	PM_o
23.58	11.62	2.53	0.36	37.89	2.69	11.81

Parameter Set 4.4

g_i	z_{i1}	z_{i2}	g_o	z_{o1}	z_{o2}	ω_{gu}	ω_{go}	$\|S_u\|$	$\|S_o\|$
29.85	1	1.1	0.88	0.55	0.01	27.8	0.42	4.03 dB	4.03 dB

ω_{gi}	ω_{gsu}	ω_{gso}	$\downarrow GM_u$	PM_u	$\uparrow GM_o$	PM_o
28.76	0.95	1.01	0.61	85.57	11.81	89

Parameter Set 4.5

g_i	z_{i1}	z_{i2}	g_o	z_{o1}	z_{o2}	ω_{gu}	ω_{go}	$\|S_u\|$	$\|S_o\|$
40	11	16.1	11.81	2.92	0.03	35.93	0.12	4.03 dB	4.03 dB

ω_{gi}	ω_{gsu}	ω_{gso}	$\downarrow GM_u$	PM_u	$\uparrow GM_o$	PM_o
43.38	28.4	0.035	0.25	37.16	2.69	77.69

Parameter Set 4.6

g_i	z_{i1}	z_{i2}	g_o	z_{o1}	z_{o2}	ω_{gu}	ω_{go}	$\|S_u\|$	$\|S_o\|$
47	11	21.1	14.5	1.02	0.03	42.91	0.04	4.03 dB	4.03 dB

ω_{gi}	ω_{gsu}	ω_{gso}	$\downarrow GM_u$	PM_u	$\uparrow GM_o$	PM_o
51.33	28.4	0.03	0.25	37.169	2.67	55.96

From these above tables, we get:

ω_{gu}	ω_{go}	$\frac{\omega_{go}}{\omega_{gu}}$	ω_{gi}
10.75	0.89	0.08	13.13
13.1	0.71	0.05	15.15
19.32	0.5	0.025	23.58
27.8	0.42	0.015	28.76
35.93	0.12	0.003	43.38
42.91	0.04	0.0009	51.33

Table 4.7.18: Equilibrated Design Bandwidths

- The PID search using six for loops corresponding to each of the gain parameters involved in the controller, gives us designs related to every other design used in this thesis.

- As the peaks increases , open loop bandwidths - ω_{gu} drops and the ω_{go} rises up.

- Also the $\downarrow GM_u$ increases , $\uparrow GM_o$ decreases with rising peaks.

- Furthermore, the open loop bandwidth of the inner loop ω_{gi} is seen to be approximately equal to the inner loop gain g_i.

- The root locus helps in understanding the behavior of poles and zeros and how the peaks change with respect to them. The sensitivities associated depict the behavior of closed loops near the poles.

- The ratio decreases with respect to increasing ω_{gu}.

- The ratio is seen to be increasing with respect to increasing equilibrated peaks

Examine Equilibrated PID Designs – Open Loop Plots

Figure 4.7.14: Examine Equilibrated PID Designs – Open Loop Plots

Examine Equilibrated PID Designs – Closed Loop Plots
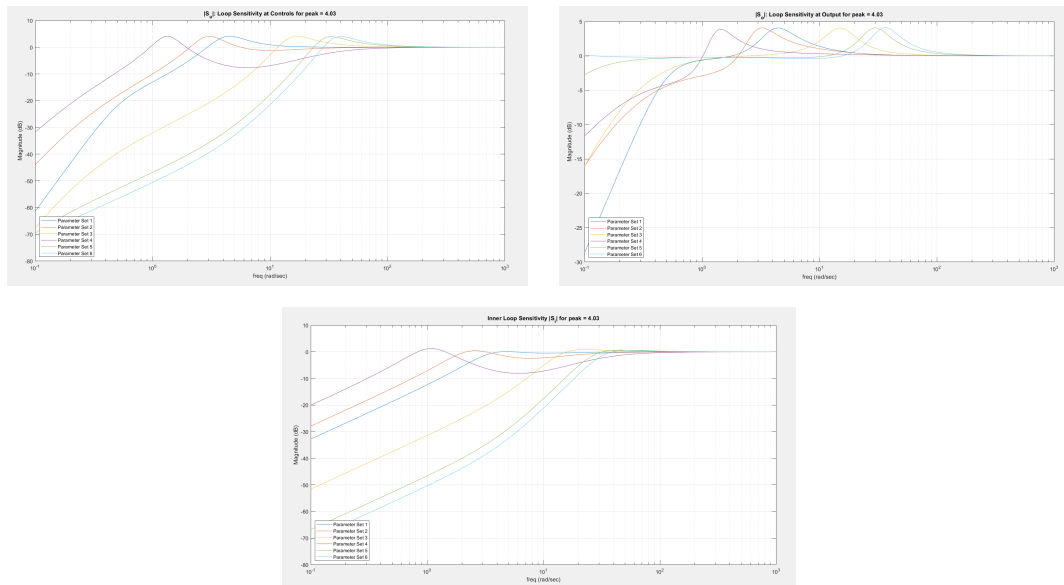
Figure 4.7.15: Examine Equilibrated PID Designs – Closed Loop Plots

Examine Equilibrated PID Designs – Comparing Same Peaks (12.2)

Parameter Set 5.1

g_i	z_{i1}	z_{i2}	g_o	z_{o1}	z_{o2}	ω_{gu}	ω_{go}	$\|S_u\|$	$\|S_o\|$
14.79	3.06	3.16	4.35	2	0.4	7.62	4.01	12.2 dB	12.2 dB

ω_{gi}	ω_{gsu}	ω_{gso}	$\downarrow GM_u$	PM_u	$\uparrow GM_o$	PM_o
13.15	3.21	2.92	0.78	31.61	1.43	29.22

Parameter Set 5.2

g_i	z_{i1}	z_{i2}	g_o	z_{o1}	z_{o2}	ω_{gu}	ω_{go}	$\|S_u\|$	$\|S_o\|$
16.64	2.01	2.7	3.749	0.07	1	9.51	3.3	12.2 dB	12.2 dB

ω_{gi}	ω_{gsu}	ω_{gso}	$\downarrow GM_u$	PM_u	$\uparrow GM_o$	PM_o
14.76	2.66	2.64	0.69	56.63	2.05	37.88

Parameter Set 5.3

g_i	z_{i1}	z_{i2}	g_o	z_{o1}	z_{o2}	ω_{gu}	ω_{go}	$\|S_u\|$	$\|S_o\|$
22.342	10.16	5.4	11.8	4.3	0.1	15.58	1.98	12.2 dB	12.2 dB

ω_{gi}	ω_{gsu}	ω_{gso}	$\downarrow GM_u$	PM_u	$\uparrow GM_o$	PM_o
23.58	10.56	4.27	0.61	15.58	1.32	83.03

Parameter Set 5.4

g_i	z_{i1}	z_{i2}	g_o	z_{o1}	z_{o2}	ω_{gu}	ω_{go}	$\|S_u\|$	$\|S_o\|$
25.35	1	1.1	1.88	0.55	0.01	21.96	1.69	12.2 dB	12.2 dB

ω_{gi}	ω_{gsu}	ω_{gso}	$\downarrow GM_u$	PM_u	$\uparrow GM_o$	PM_o
26.23	1.19	1.35	0.8	84.18	1.32	85.66

Parameter Set 5.5

g_i	z_{i1}	z_{i2}	g_o	z_{o1}	z_{o2}	ω_{gu}	ω_{go}	$\|S_u\|$	$\|S_o\|$
40	11.1	16.1	23.9	2.92	0.03	31.92	0.24	12.2 dB	12.2 dB

ω_{gi}	ω_{gsu}	ω_{gso}	$\downarrow GM_u$	PM_u	$\uparrow GM_o$	PM_o
43.3	21.78	13.30	0.5	14.97	1.32	85.66

Parameter Set 5.6

g_i	z_{i1}	z_{i2}	g_o	z_{o1}	z_{o2}	ω_{gu}	ω_{go}	$\|S_u\|$	$\|S_o\|$
47	11	21.1	29.3	1.02	0.03	38.65	0.07	12.2 dB	12.2 dB

ω_{gi}	ω_{gsu}	ω_{gso}	$\downarrow GM_u$	PM_u	$\uparrow GM_o$	PM_o
51.33	26.5	17.63	0.46	14.54	1.32	71.19

From these above tables, we get:

ω_{gu}	ω_{go}	$\frac{\omega_{go}}{\omega_{gu}}$	ω_{gi}
7.62	4.01	0.526	13.15
9.51	3.3	0.347	14.76
15.58	1.98	0.127	23.58
21.96	1.69	0.076	26.23
31.92	0.24	0.007	43.3
28.65	0.07	0.001	51.33

Table 4.7.19: Equilibrated Design Bandwidths

- The PID search using six for loops corresponding to each of the gain parameters involved in the controller, gives us designs related to every other design used in this thesis.

- As the peaks increases , open loop bandwidths - ω_{gu} drops and the ω_{go} rises up.

- Also the $\downarrow GM_u$ increases , $\uparrow GM_o$ decreases with rising peaks.

- Furthermore, the open loop bandwidth of the inner loop ω_{gi} is seen to be approximately equal to the inner loop gain g_i.

- The root locus helps in understanding the behavior of poles and zeros and how the peaks change with respect to them. The sensitivities associated depict the behavior of closed loops near the poles.

- The ratio decreases with respect to increasing ω_{gu}.

- The ratio is seen to be increasing with respect to increasing equilibrated peaks

Examine Equilibrated PID Designs – Open Loop Plots

Figure 4.7.16: Examine Equilibrated PID Designs – Open Loop Plots

Examine Equilibrated PID Designs – Closed Loop Plots
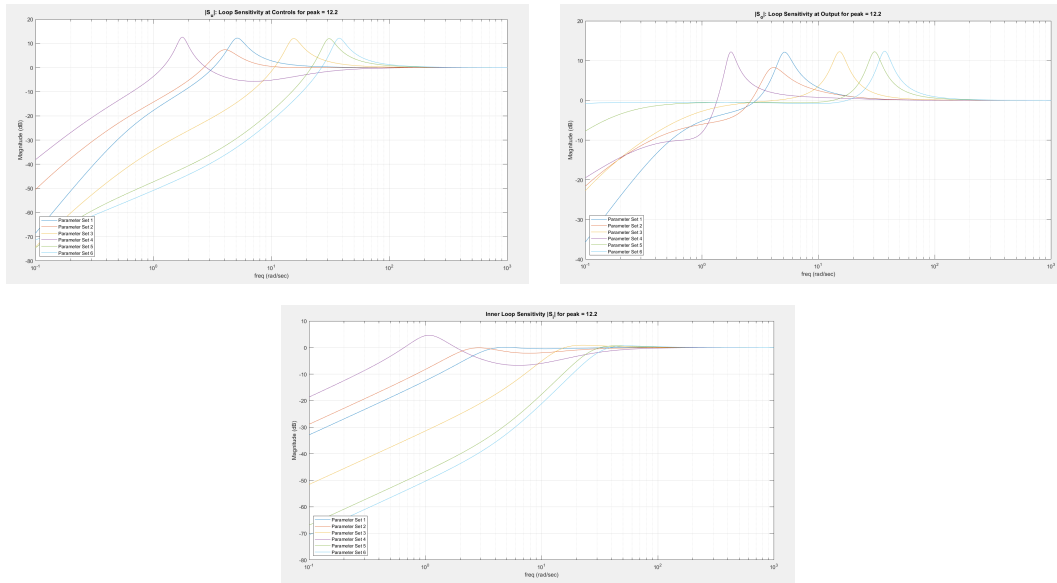
Figure 4.7.17: Examine Equilibrated PID Designs – Closed Loop Plots

4.8   Summary and Conclusions

From this chapter, it is concluded that it is possible to get good properties at the input and at the output using inner-outer loop control architecture. The conventional $P - K$ structure is not viable for a good design to get useful properties. Also, the pole-placement design does not shed any light on improving the properties of the inner loop and the outer loop. Moreover, it has the same properties at the input as well as the output. Thus, this motivates the need of this inner-outer loop architecture. The different designs using linear quadratic regulator, generalized mixed sensitivity earn us a new pathway for shaping the loop sensitivity to get a good design.

The Zeigler Nichols initiated pid search takes about more than three hours to get desired peaks. Furthermore, it obtains the family of designs, with excellent $|S_u|$ and $|S_o|$ peaks. Additionally, it also obtains a family of equilibrated designs (peak $|S_u| = |S_o|$). The identified best equilibrated designs peak $|S_u| = |S_o|$ ($1.65dB, 2.8dB$). These have maximum open loop bandwidth ratio $\frac{\omega_{go}}{\omega_{gu}}$ for given peak (no design with larger ratio exists). One of the equilibrated designs (found via search) was obtained via Generalized Mixed Sensitivity (Youla Parameterization, convex optimization) which takes about one hour.

The PID search using six for loops corresponding to each of the gain parameters involved in the controller, gives us designs related to every other design used in this thesis However, the drawback is that it takes a lot more time to get the requirements of a design on focus. The linear quadratic regulator on the other hand and the general mixed sensitivity using $\mathcal{H}^\infty$ design uses norms to get good PID parameters faster than the search. However the weights are chosen heuristically but they take lesser time to construct an algorithm.

Chapter 5

SUMMARY, CONCLUSIONS AND FUTURE DIRECTIONS

5.1   Summary and Conclusions

This thesis illuminates fundamental design trade-offs (peak sensitivity at controls and at output) associated with the cart-inverted pendulum system. It illustrates how to tackle the control issues such as unstable, non-minimum phase associated with the system under inner-outer loop PID control. The shortest pendulum requires the maximum bandwidth at controls (u - bandwidth). It is most likely to saturate the control input. However, this system, will have the largest bandwidth at output (x - bandwidth) due to the largest right hand plane zero.

For large K_i, $|S_i| < 1$ which implies that $|S_u| < |S_o|$ which results in an aggressive inner loop and a non-aggressive outer loop. On the contrary, for small K_i, $|S_i| > 1$ which implies that $|S_u| > |S_o|$ which results in an non-aggressive inner loop and an aggressive outer loop. The Ziegler Nichols initiated pid search helps us attain good peak $|S_o|$ (3dB), barely acceptable peak $|S_u|$ (7dB) which requires work (modified Zeigler Nichols) to improve peak $|S_u|$. The LQR PI has been used to obtain good peak $|S_u|$ (0dB as expected), unacceptable peak $|S_o|$ (21dB) which requires work (derivative action) to improve peak $|S_u|$. The LQR PID on the other hand, obtains good peak $|S_u|$ (0dB as expected); acceptable peak $|S_o|$ (3dB) using diagonal Q and R.

The Ziegler Nichols initiated pid search takes about more than three hours to get desired peaks. Furthermore, it obtains the family of designs, with excellent $|S_u|$ and $|S_o|$ peaks. Additionally, it also obtains a family of equilibrated designs (peak $|S_u| =$

$|S_o|$). The identified best equilibrated designs peak $|S_u| = |S_o|$ ($1.65dB$, $2.8dB$). These have maximum open loop bandwidth ratio $\frac{\omega_{go}}{\omega_{gu}}$ for given peak (no design with larger ratio exists). One of the equilibrated designs (found via search) was obtained via Generalized Mixed Sensitivity (Youla Parameterization, convex optimization) which takes about one hour. The inner loop zeros move towards inner loop integrators and pole. The outer loop zeros move towards outer loop integrators (used for a good design). Root locus ideas help in understanding the equilibrated designs better. Inner loop zeros provide lead to deal with inner loop integrators and plant poles. Outer loop zeros provide lead which causes outer loop integrators to move towards them. The best bandwidth ratio is observed at 4 dB. If 6 dB is our sensitivity limit, then there exists only one equilibrated design with sensitivity less than 6 dB and a maximum bandwidth ratio. Similarly, if 3 dB is our sensitivity limit, then there exists one equilibrated design with sensitivity less than 3 dB and a maximum bandwidth ratio.

5.2   Directions for Future Research

These analogies will be used for computing stabilized controllers for biped walkers. Using the work and directions from this thesis, this work will be used to combine the analogies into walkers - quadrupeds, bipeds, unipeds.

From this thesis, it is evident that one can quantify the trade-offs used and use them for getting required robust designs to get good properties at the inner loop and the outer loop. The proportional integral derivative search assists us with this and helps us find some interesting equilibrated designs. However, there are infinite equilibrated designs associated with this design. Hence, the future scope includes investigating these equilibrated designs and obtain some patterns that will assist us with understanding the trade-offs better.

REFERENCES

Åström, K., & Hägglund, T. (2004). Revisiting the ziegler–nichols step response method for pid control. *Journal of Process Control*, *14*(6), 635–650. https://doi.org/https://doi.org/10.1016/j.jprocont.2004.01.002

Bertsekas, D. (2009). Convex optimization theory.

Bertsekas, D. (2015). Convex optimization algorithms.

Boyd, S., & Vandenberghe, L. (2004). *Convex optimization*. Cambridge University Press.

Chen, M.-Y., & Byl, K. (2012). Analysis and control techniques for the compass gait with a torso walking on stochastically rough terrain. *2012 American Control Conference (ACC)*, 3451–3458. https://doi.org/10.1109/ACC.2012.6315669

Chrif, L., & Kadda, Z. M. (2014). Aircraft control system using lqg and lqr controller with optimal estimation-kalman filter design [3rd International Symposium on Aircraft Airworthiness (ISAA 2013)]. *Procedia Engineering*, *80*, 245–257. https://doi.org/https://doi.org/10.1016/j.proeng.2014.09.084

Fung, H. (2011). Modeling, simulation and control for a bipedal walking robot.

Gandhi, P. S., Borja, P., & Ortega, R. (2016). Energy shaping control of an inverted flexible pendulum fixed to a cart. *CoRR*, *abs/1605.02991*. http://arxiv.org/abs/1605.02991

Gattami, A., & Rantzer, A. (2005). Linear quadratic performance criteria for cascade control. *Proceedings of the 44th IEEE Conference on Decision and Control*, 3632–3637. https://doi.org/10.1109/CDC.2005.1582726

Gu, W. (2009). Design and control of a 10 dof biped robot.

He, B., Wang, S., & Liu, Y. (2019). Underactuated robotics: A review. *International Journal of Advanced Robotic Systems*, *16*(4), 1729881419862164. https://doi.org/10.1177/1729881419862164

He, J.-B., Wang, Q.-G., & Lee, T.-H. (2000). Pi/pid controller tuning via lqr approach. *Chemical Engineering Science*, *55*(13), 2429–2439. https://doi.org/https://doi.org/10.1016/S0009-2509(99)00512-6

Hernandez, J. I. (2000). Development of an extensible interactive modeling, simulation, animation and real-time control (mosart) flexible inverted pendulum environment: A tool for enhancing research and education.

Hu, T., & Lin, Z. (2001). *Control systems with actuator saturation: Analysis and design*. Birkhauser Boston, Inc.

Irfan, S., Mehmood, A., Razzaq, M. T., & Iqbal, J. (2018). Advanced sliding mode control techniques for inverted pendulum: Modelling and simulation. *Engineering Science and Technology, an International Journal*, *21*(4), 753–759. https://doi.org/https://doi.org/10.1016/j.jestch.2018.06.010

Kaya, İ., Tan, N., & Atherton, D. P. (2007). Improved cascade control structure for enhanced performance. *Journal of Process Control*, *17*(1), 3–16. https://doi.org/https://doi.org/10.1016/j.jprocont.2006.08.008

Kwakernaak, H. (1993). Robust control and h-optimization—tutorial paper. *Automatica*, *29*(2), 255–273. https://doi.org/https://doi.org/10.1016/0005-1098(93)90122-A

Shahrokhi, M., & Zomorrokhi, A. (2013). Comparison of pid controller tuning methods.

TAMADDONI, S. H., JAFARI, F., MEGHDARI, A., & SOHRABPOUR, S. (2010). Biped hopping control bazsed on spring loaded inverted pendulum model. *International Journal of Humanoid Robotics, 07*(02), 263–280. https://doi.org/10.1142/S0219843610002106

Middleton, R. (1991). Trade-offs in linear control system design. *Automatica, 27*(2), 281–292. https://doi.org/https://doi.org/10.1016/0005-1098(91)90077-F

Mondal, K., Rodriguez, A., Manne, S., Das, N., & Wallace, B. (2019). Comparison of kinematic and dynamic model based linear model predictive control of non-holonomic robot for trajectory tracking: Critical trade-offs addressed. *Proceedings of the IASTED International Conference on Mechatronics and Control, MC 2019*, 9–17. https://doi.org/10.2316/P.2019.860-017

Mondal, K., Wallace, B., & Rodriguez, A. A. (2020). Stability versus maneuverability of non-holonomic differential drive mobile robot: Focus on aggressive position control applications. *2020 IEEE Conference on Control Technology and Applications (CCTA)*, 388–395. https://doi.org/10.1109/CCTA41146.2020.9206155

Mu, X. (2004a). A dual slip model for dynamic walking in a humanoid over unevern terrain.

Mu, X. (2004b). Dynamics and motion of a five-link biped robot walking in the saggital plane.

Mu, X. (2004c). Dynamics and motion regulation of a five link biped robot walking in the saggital plane.

Namasivayam, V. (2020). Design, modeling and control of inverted pendulum on a cart.

Ogata, K. (2001). *Modern control engineering* (4th). Prentice Hall PTR.

Persson, J. (2013). Comparison of three linearization methods.

Ponce, P., Molina, A., & Alvarez, E. (2014). A review of intelligent control systems applied to the inverted-pendulum problem. *American Journal of Engineering and Applied Sciences, 7*, 194–240.

Prasad, L. B., Tyagi, B., & Gupta, H. (2014). Optimal control of nonlinear inverted pendulum system using pid controller and lqr: Performance analysis without and with disturbance input. *International Journal of Automation and Computing, 11*, 661–670.

Puttannaiah, K., Rodriguez, A. A., Mondal, K., Echols, J. A., & Cartagena, D. G. (2016). A generalized mixed-sensitivity convex approach to hierarchical multivariable inner-outer loop control design subject to simultaneous input and output loop breaking specifications. *2016 American Control Conference (ACC)*, 5632–5637. https://doi.org/10.1109/ACC.2016.7526553

Raibert, M. H. (1986). *Legged robots that balance*. Massachusetts Institute of Technology.

Rigatos, G., Siano, P., Abbaszadeh, M., Ademi, S., & Melkikh, A. (2018). Nonlinear h-infinity control for underactuated systems: The furuta pendulum example. *International Journal of Dynamics and Control, 6*, 835–847.

Rodriguez, A. A. (2003). Analysis and design of feedback systems.

Rodriguez, A. A. (2004). Analysis and design of multivariable feedback control systems.

Turkoglu, K., & Jafarov, E. M. (2007). Augmented optimal lqr control system design for the longitudinal flight dynamics of an uav: Inner and outer loop concepts. *Proceedings of the 9th WSEAS International Conference on Automatic Control, Modelling and Simulation*, 100–105.

Vázquez, J. A., & Velasco-Villa, M. (2013). Design and real-time control of a 4-dof biped robot. *International Journal of Advanced Robotic Systems*, *10*(8), 312. https://doi.org/10.5772/56738

Vinodh Kumar, E., & Jerome, J. (2013). Robust lqr controller design for stabilizing and trajectory tracking of inverted pendulum [International Conference on Design and Manufacturing (IConDM2013)]. *Procedia Engineering*, *64*, 169–178. https://doi.org/https://doi.org/10.1016/j.proeng.2013.09.088

Westervelt, E. R., Grizzle, J. W., Chevallereau, C., Choi, J. H., & Morris, B. (2018). Feedback control of dynamic bipedal robot locomotion. https://doi.org/10.1201/9781420053739

Xin, X. (2008). Analysis of the energy based swing-up control for a double pendulum on a cart [17th IFAC World Congress]. *IFAC Proceedings Volumes*, *41*(2), 4828–4833. https://doi.org/https://doi.org/10.3182/20080706-5-KR-1001.00811

Yi, J., Zhu, Q., Xiong, R., & Wu, J. (2016). Walking algorithm of humanoid robot on uneven terrain with terrain estimation. *International Journal of Advanced Robotic Systems*, *13*(1), 35. https://doi.org/10.5772/62245

Younis, W., & Abdelati, M. (2009). Design and Implementation of an Experimental Segway Model. In L. Beji, S. Otmane, & A. Abichou (Eds.), *Intelligent systems and automation: 2nd mediterranean conference on intelligent systems and automation (cisa'09)* (pp. 350–354). https://doi.org/10.1063/1.3106501

Yu, X., Fu, C., & Chen, K. (2011). Modeling and control of a single-legged robot [International Conference on Advances in Engineering 2011]. *Procedia Engineering*, *24*, 788–792. https://doi.org/https://doi.org/10.1016/j.proeng.2011.11.2738

Zhang, C., Hu, H., Gu, D., & Wang, J. (2016). Cascaded control for balancing an inverted pendulum on a flying quadrotor. *Robotica*, *35*, 1263–1279.

APPENDIX A

A   DERIVATION OF NON-LINEAR CART PENDULUM MODEL

DERIVATION OF NON-LINEAR CART PENDULUM MODEL

Kinematics

$$x_m = x + l\sin(\theta), \dot{x}_m = \dot{x}_m + \dot{\theta}l\cos(\theta) \tag{A.1}$$

$$y_m = l\cos(\theta), \dot{y}_m = -\dot{\theta}l\sin(\theta) \tag{A.2}$$

Kinetic Energy

$$K = \frac{1}{2}M\dot{x}^2 + \frac{1}{2}m(\dot{x}_m^2 + \dot{y}_m^2) + \frac{1}{2}I\dot{\theta}^2 = \frac{1}{2}(m+M)\dot{x}^2 + \frac{1}{2}m\dot{\theta}^2 l^2 + m\dot{\theta}^2 l^2 + \frac{1}{2}I\dot{\theta}^2 \tag{A.3}$$

Potential Energy

$$u = mgh = mgl\cos(\theta) \tag{A.4}$$

$$L = K - U = \frac{1}{2}(m+M)\dot{x}^2 + \frac{1}{2}m\dot{\theta}^2 l^2 + m\dot{\theta}^2 l^2 + \frac{1}{2}I\dot{\theta}^2 - mgl\cos(\theta) \tag{A.5}$$

Langrange Equations

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{q}_i} - \frac{\partial L}{\partial q_i} = Q_i \tag{A.6}$$

Equations of motion

$$(I + ml^2)\ddot{\theta} + ml\ddot{x}\cos(\theta) - mgl\sin(\theta) = 0 \tag{A.7}$$

$$(m+M)\ddot{x} + b\dot{\theta} + m\ddot{\theta}l\cos(\theta) - m\dot{\theta}^2 l\sin(\theta) = F \tag{A.8}$$

APPENDIX B

B   MATLAB CODE

```
%% example 1
function [Pi,Ki,Po,Ko]  = example_1(p,z,gi,go,zi1,zi2,zo1,zo2)
s = tf('s')
Pi = zpk([],[p],[1])
Ki = gi
Po = zpk([z],[0],[-1])
Ko = zpk([],[0],[go]);  %works but oscillatory
end
%% example 1 sym
function [Pi,Ki,Po,Ko]  = example_1sym(p,z,gi,go,zi1,zi2,zo1,zo2,s)
zo = zo1;
Pi = 1/(s-p); %inner plant
Po = (z-s)/s; %outer plant
Ki = gi; % inner loop gain
Ko = go/s; % outer loop gain
end
%% example 2
function [Pi,Ki,Po,Ko]  = example_2(p,z,gi,go,zi1,zi2,zo1,zo2)
s = tf('s')
zo = zo1;
Pi = zpk([],[p],[1])
Ki = gi
Po = zpk([z],[0],[-1])
Ko = zpk([-zo],[0],[go]); %works but oscillatory
end
%% example 2 sym
function [Pi,Ki,Po,Ko]  = example_2sym(p,z,gi,go,zi1,zi2,zo1,zo2,s)
zo = zo1;
Pi = 1/(s-p); %inner plant
Po = (z-s)/s; %outer plant
Ki = gi; % inner loop gain
Ko = go*(s + zo)/s; % outer loop gain
end
%% example 3
function [Pi,Ki,Po,Ko]  = example_3(p,z,gi,go,zi1,zi2,zo1,zo2)
s = tf('s')
Pi = zpk([],[p],[1])
Ki = gi % does not work
Po = zpk([z],[0,0],[-1])
Ko = zpk([-zo1,-zo2],[0],[go])
end
%% example 3 syms
function [Pi,Ki,Po,Ko]  = example_3sym(p,z,gi,go,zi1,zi2,zo1,zo2,s)
Pi = 1/(s-p); %inner plant
Po = (z-s)/s^2; %outer plant
Ki = gi; % inner loop gain
Ko = go*(s + zo1)*(s + zo2)/s; % outer loop gain
end
%% example 4
function [Pi,Ki,Po,Ko]  = example_4(p,z,gi,go,zi1,zi2,zo1,zo2)
s = tf('s')
zi = zi1;
Pi = zpk([],[p],[1])
Ki = zpk([-zi],[0],[gi])
Po = zpk([z],[0,0],[-1])
Ko = zpk([-zo1,-zo2],[0],[go]);
end
%% example 4 sym
function [Pi,Ki,Po,Ko]  = example_4sym(p,z,gi,go,zi1,zi2,zo1,zo2,s)
zi = zi1;
Pi = 1/(s-p); %inner plant
Po = (z-s)/s^2; %outer plant
Ki = gi*(s + zi)/s; % inner loop gain
Ko = go*(s + zo1)*(s + zo2)/s; % outer loop gain
end
%% example 5
function [Pi,Ki,Po,Ko]  = example_5(p,z,gi,go,zi1,zi2,zo1,zo2)
s = tf('s')
zi = zi1;
Pi = zpk([],[p,-p],[1])
Ki = zpk([-zi],[],[gi])
Po = zpk([z],[0,0],[-1])
Ko = zpk([-zo1,-zo2],[0],[go])
end
%% example 5 sym
function [Pi,Ki,Po,Ko]  = example_5sym(p,z,gi,go,zi1,zi2,zo1,zo2)
zi = zi1;
Pi = 1/((s-p)*(s+p)); %inner plant
Po = (z-s)/s^2; %outer plant
Ki = gi*(s + zi); % inner loop gain
Ko = go*(s + zo1)*(s + zo2)/s; % outer loop gain
end
%% example 6
function [Pi,Ki,Po,Ko]  = example_6(p,z,gi,go,zi1,zi2,zo1,zo2)
s = tf('s')
Pi = zpk([],[p,-p],[1])
Ki = zpk([-zi1,-zi2],[0],[gi])
Li_tf = series(Pi,Ki)
Po = zpk([z],[0,0],[-1])
Ko = zpk([-zo1,-zo2],[0],[go])
end
%% example 6 sym
function [Pi,Ki,Po,Ko]  = example_6sym(p,z,gi,go,zi1,zi2,zo1,zo2)
Pi = 1/((s-p)*(s+p)); %inner plant
Po = (z-s)/s^2; %outer plant
```

```
Ki = gi*(s + zi1)*(s + zi2)/s; % inner loop gain
Ko = go*(s + zo1)*(s + zo2)/s; % outer loop gain
end
%% example 7
function [Pi,Ki,Po,Ko]  = example_7(p,z,gi,go,zi1,zi2,zo1,zo2)
gpi = 1.928;
Pi = zpk([],[p,-p],[gpi])
Ki = zpk([-zi1,-zi2],[0],[gi])
Li_tf = series(Pi,Ki)
Po = zpk([z,-z],[0,0],[-1])
Ko = zpk([-zo1,-zo2],[0],[go])
end
%% example 7 sym
function [Pi,Ki,Po,Ko]  = example_7sym(p,z,gi,go,zi1,zi2,zo1,zo2)
Pi = 1/((s-p)*(s+p)); %inner plant
Po = (z-s)*(z+s)/s^2; %outer plant
Ki = gi*(s + zi1)*(s + zi2)/s; % inner loop gain
Ko = go*(s + zo1)*(s + zo2)/s; % outer loop gain
end
%% Symbolic Representation
clc
clear all
close all

syms p z zi1 zi2 zo1 zo2 gi go s 'real';

% Pi = 1/((s-p)*(s+p)); %inner plant
% Po = (z-s)*(z+s)/s^2; %outer plant
% Ki = gi*(s + zi1)*(s + zi2)/s; % inner loop gain
% Ko = go*(s + zo1)*(s + zo2)/s; % outer loop gain
[Pi,Ki,Po,Ko]  = example_1sym(p,z,gi,go,zi1,zi2,zo1,zo2,s)
pretty(Pi)
pretty(Po)
pretty(Ki)
pretty(Ko)

Li = Pi*Ki; % inner loop
Li = simplifyFraction(Li);
disp('Inner Loop Li');
pretty(Li)
Lo = Po*Pi*Ko*[1/(1+(Pi*Ki))]; % loop at output
Lo = simplifyFraction(Lo);
disp('Loop at Output Lo');
pretty(Lo)
Lu = ((Ki*Pi)+(Ko*Po*Pi)); % loop at controls
Lu = simplifyFraction(Lu);
disp('Loop at Controls Lu');
pretty(Lu)

Si = 1/(1 + Li); % Sensitivity at Loop at Output
Si = simplifyFraction(Si);
disp('Sensitivity at Inner Loop');
pretty(Si)
Ti = Li/(1 + Li); % Complementary Sensitivity at Loop at Output
Ti = simplifyFraction(Ti);
disp('Complementary Sensitivity at Inner Loop');
pretty(Ti)
So = 1/(1 + Lo); % Sensitivity at Loop at Output
So = simplifyFraction(So);
disp('Sensitivity at Loop at Output');
pretty(So)
To = Lo/(1 + Lo); % Complementary Sensitivity at Loop at Output
To = simplifyFraction(To);
disp('Complementary Sensitivity at Loop at Output');
pretty(To)
Su = 1/(1 + Lu); % Loop Sensitivity at Controls
Su = simplifyFraction(Su);
disp('Sensitivity at Loop at Controls');
pretty(Su)
Tu = Lu/(1 + Lu); % Loop Sensitivity at Controls
Tu = simplifyFraction(Tu);
disp('Complementary Sensitivity at Loop at Controls');
pretty(Tu)
%% For gi_stability
[num,eqn] = numden(So);
syms w 'real';
s = w*i;
p  = 1;
z  = 1;
zi1 = 0.1;
zi2 = 3;
zo1 = 0.05;
zo2 = 0.05;
go = 2;
zi = zi1;
zo = zo1;
eqn = subs(eqn);
aw = real(eqn);
bw = imag(eqn);
a = solve(aw,w);
b = solve(bw,w);
b = b(2:5);
f = a-b == 0;
f1 = f(2);
gi = solve(f1,gi)
gi = vpa(subs(gi))
%% Benchmark Example 8 cart inverted
```

```
%% Cart Inverted Pendulum Analysis
clc
clear all
close all


%% Derivation of Equations (Matthew Peter Kelly)
syms pos(t) pos_dot(t) pos_dbldot(t) theta(t) theta_dot(t) theta_dbldot(t);
syms m M l b g I u;

Xp = pos(t) + l*sin(theta(t));
Yp = l*cos(theta(t));
Xp_dot = diff(Xp);
Yp_dot = diff(Yp);


Xp_dot = subs(Xp_dot,[diff(pos(t),t), diff(theta(t),t)],[pos_dot(t) theta_dot(t)]);
Yp_dot = subs(Yp_dot,[diff(pos(t),t), diff(theta(t),t)],[pos_dot(t) theta_dot(t)]);

pos_dot(t) = subs(pos_dot(t),diff(pos(t),t),pos_dot(t));
T =  ((1/2)*M*pos_dot(t)) + ((1/2)*m*((Xp_dot^2) + (Yp_dot^2)));

T = ((1/2)*M*pos_dot(t)^2)+((1/2)*(m)*(pos_dot(t)^2))...
    +((1/2)*m*(2*l*pos_dot(t)*theta_dot(t)*cos(theta(t))))...
    +((1/2)*m*(l^2)*(theta_dot^2))+((1/2)*I*(theta_dot^2))
U = m*g*l*cos(theta(t));
L = T - U;

% First Equation with respect to theta1(t)
dL_dpos_dot(t) = functionalDerivative(L,pos_dot(t));

dL_dpos(t) = functionalDerivative(L,pos(t));

% Second Equation with respect to theta
dL_dtheta_dot(t) = functionalDerivative(L,theta_dot(t));

dL_dtheta(t) = functionalDerivative(L,theta(t));


dL_dpos_dot_dt = jacobian(dL_dpos_dot(t),t);
dL_dpos_dot_dt = subs(dL_dpos_dot_dt,[diff(pos(t),t),...
    diff(pos_dot(t),t), diff(theta(t),t), diff(theta_dot(t),t)]...
    ,[pos_dot(t),pos_dbldot(t),theta_dot(t),theta_dbldot(t)]);


eqn1 = dL_dpos_dot_dt - dL_dpos(t) + b*theta_dot(t) - u;

% theta_1dbldot(t) = solve(eqn1,theta_1dbldot(t),'Real',true);

fprintf('Equation 1: \n');
disp(eqn1)
% fprintf('Equation 1 in terms of theta_1dbldot(t): \n');
% disp(theta_1dbldot(t))

% Remember this trick that you did, apply for all
dL_dtheta_dot_dt = jacobian(dL_dtheta_dot(t),t);
dL_dtheta_dot_dt = subs(dL_dtheta_dot_dt,[diff(pos(t),t),...
    diff(pos_dot(t),t), diff(theta(t),t), diff(theta_dot(t),t)]...
    ,[pos_dot(t),pos_dbldot(t),theta_dot(t),theta_dbldot(t)]);
eqn2 = dL_dtheta_dot_dt - dL_dtheta(t);


fprintf('Equation 2: \n');
disp(eqn2)
%% Finding the Langrange Equations with variables
syms x th x_dot th_dot x_dbldot th_dbldot;

eqn1 = subs(eqn1,[pos(t),theta(t),pos_dot(t),theta_dot(t),...
    pos_dbldot(t),theta_dbldot(t)], [x,th,x_dot,th_dot,...
    x_dbldot,th_dbldot]);
fprintf('Equation of Motion 1: \n');
disp(eqn1)

eqn2 = subs(eqn2,[pos(t),theta(t),pos_dot(t),theta_dot(t),...
    pos_dbldot(t),theta_dbldot(t)], [x,th,x_dot,th_dot,...
    x_dbldot,th_dbldot]);
fprintf('Equation of Motion 2: \n');
disp(eqn2)

%% Linearization
syms u;
[x_dbldot,th_dbldot]= solve([eqn1,eqn2],[x_dbldot,th_dbldot])

q = [x x_dot th th_dot];
u = [u];

% Non Linear Function
f1 = x_dot;
f2 = x_dbldot;
f3 = th_dot;
f4 = th_dbldot;

F = [f1 f2 f3 f4];

% Jacobian
J = jacobian(F,q)
```

```matlab
% Equilibrium Points
q_e = [0 0 0 0];
J = subs(J,q,q_e);
disp(J);

% System Matrix
A = J;
% Input Matrix
B = jacobian(F,u);
B = subs(B,q,q_e);



%% Plant Nominal Values (hernandez)
m = 0.267; % in Kilograms
L = 2.016; % in metres
l = L/2; % in metres
b = 0.0015; % in Kg m^2/s
k = 0; % in Kg/s^2
g = 9.8; % in m/s^2
I = 0; % in Nm/rad^2/sec^2
M = 0.8; %in Kilograms

%% Plant Nominal Values (ogata)
m = 0.21; % in Kilograms
L = 0.61; % in metres
l = L/2; % in metres
b = 0; % in Kg m^2/s
k = 0; % in Kg/s^2
g = 9.8; % in m/s^2
I = 0.006; % in Nm/rad^2/sec^2
M = 0.455; %in Kilograms

%% Plant Nominal Values (vignesh)
m = 0.20; % in Kilograms
% L = 0.115; % in metres
l = 0.115; % in metres
b = 0.0005; % in Kg m^2/s
k = 0; % in Kg/s^2
g = 9.81; % in m/s^2
I = 0.0002545; % in Nm/rad^2/sec^2
M = 1.21; %in Kilograms

%% State Space
Ap = double(subs(A)) %System Matrix
Bp = double(subs(B)) % Input Matrix
% Cp = [1 0; 0 1];
% Cp = [1 0 0 0;
%       0 0 1 0] % Output Matrix
Cp = [1 0 0 0] % Output Matrix
% Dp = [0; 0];
Dp = zeros(size(Cp,1),size(Bp,2)) % Feedforward Matrix
%% Scaling the Matrices
% FACTS ON SCALING: Scaling affects the shape of singular value plots
%                   (i.e. multivariable frequency responses).
%                   It does not alter pole locations, zero locations.
%                   It does alter directionality information.
%
% Scaling Matrices
%
%   unew = su uold    (foot pounds, foot pounds)
%   xnew = sx xold    (degrees, degrees, degrees/sec, degrees/sec)
%   ynew = sy yold    (degrees, degrees)
%-
%
% 0.2247 lb/N          <--- Convert N to lb
% 4.45  N/lb
% 3.281 ft/m           <--- Convert m to ft
% 0.3048 m/ft
% 180 degrees/pi rad   <--- Convert rad to deg
%
r2d = 180/pi;                                    % Convert radians to degrees
su  = diag([1] );   % Convert N-m to lb-ft
sx  = diag( [ 1, 1, r2d, r2d ]);                 % Convert radians to degrees
% sy  = diag( [ 1 1,r2d,r2d ] );
sy  = diag( [1] );              % Convert radians to degrees


%
% Scaled System Dynamics
%
%
% u = [ torque_1     (lb-ft)    torque_2     (lb-ft) ]
% x = [ theta_1     (deg)     theta_2     (deg)     dot_theta_1 (deg/sec)     dot_theta_2 (deg/sec) ]
% y = [ theta_1     (deg)     theta_2     (deg) ]
%
Ap = sx*Ap*inv(sx)
Bp = sx*Bp*inv(su)
Cp = sy*Cp*inv(sx)
Dp = sy*Dp*inv(su)
%% State Space Representation
states = {'x','\theta','x_dot','\theta_dot'}; % States
inputs = {'force'}; % Inputs
% outputs = {'x','theta_dot'}
outputs = {'x'}; % Outputs
sys_ss = ss(Ap,Bp,Cp,Dp,'statename',states,'inputname',...
    inputs,'outputname',outputs) % State Space
sys_tf = zpk(sys_ss) % Transfer Function
```

193

```
%% Lqr
% An = [zeros(1,1) Cp
%        zeros(4,1) Ap]
% Bn = [zeros(1,1)
%        Bp]
An = [zeros(2,2) [ones(size(Cp));Cp]
      zeros(4,2) Ap]
Bn = [zeros(2,1)
      Bp]
%
%
% LQR Design Parameters
% q = diag([1,1, 1, 1, 1, 1]) % 7 db at output r = 0.01
% q = diag([1,1, 1, 10, 1, 1]) % 6 db at output r = 0.01
q = diag([1,1, 100, 1, 100, 1]) % 3 db at output r = 0.01
% q = diag([2, 10, 10, 0.1, 100])
r = diag ([0.01])

% q = diag([1000000000, 200, 10, 10000, 1]) % 4.8 db at output r = 0.1
% r = diag ([0.01])
[g, k, clpoles] = lqr(An, Bn, q, r)

%%
% Form Open Loop Dynamical System
%
% State x = [ z' xp' ]' = [ z'  y'  xr' ]'
%
% where
%       z is the integrator state
%       y is the output (position and pendulum angle)
%       xr contains the rest of the state (velocity and angular velocity)
%
gz = g(:,1:2)
gy = g(:,[3])
gr = g(:,[4,5,6])

aol = [ 0*ones(2,2) 0*ones(2,4)
        -Bp*gz    Ap-Bp*[0*ones(1,1) gr] ]

bol = [ - eye(2,1)
          Bp*gy         ]

col = [ 0*ones(1,2) Cp ]

dol = 0*ones(1,1)

%%
% Form CLosed Loop Dynamics
%
acl = aol - bol*col;
bcl = bol;
ccl = col;
dcl = dol;
cls = ss(acl,bcl,ccl,dcl);
%%
Su_tf = ss(An-Bn*g, Bn, -g, eye(1,1)-0*ones(1,1))
bodemag(Su_tf);
title('LQ  Sensitivity: Plant Input')
grid
xlabel('Frequency (rad/sec)')
ylabel('Singular Values (dB)')
% hold on
% pause
%return
%%
step(Su_tf)
title('Step S_u')
grid on
%%
So_tf = ss(acl, bcl, -ccl, eye(1,1)-dcl);
bodemag(So_tf);
title('LQ Sensitivity: Error Signal')
grid
xlabel('Frequency (rad/sec)')
ylabel('Singular Values (dB)')
hold on
% pause
%%
step(So_tf)
title('Step S_o')
grid on
%%
Tu_tf = ss(An-Bn*g, Bn, g, 0*ones(1,1));
bodemag(Tu_tf);
title('LQ Complementary Sensitivity: Plant Input')
grid
xlabel('Frequency (rad/sec)')
ylabel('Singular Values (dB)')
%%
step(Tu_tf)
title('Step T_u')
grid on
%%
To_tf = ss(acl, bcl, ccl, dcl)
bodemag(To_tf(1,1));
title('LQ Complementary Sensitivity: Plant Output')
grid
```

```
xlabel('Frequency (rad/sec)')
ylabel('Singular Values (dB)')
%%
step(To_tf)
title('Step T_o')
grid on
%% NOMINAL PARAMETERS
p1  = max(pole(sys_tf));
p2  = min(pole(sys_tf));
z1  = max(tzero(sys_tf));
z2  = min(tzero(sys_tf));
zi1 = 3.06;
zi2 = 3.12;
zo1 = 2;
zo2 = 0.4;
gi = 14.79;
go = 1.41;
Pi = zpk([],[p1,p2],[1])
Ki = zpk([-zi1,-zi2],[0],[gi])
Li_tf = series(Pi,Ki)
Po = zpk([z1,z2],[0,0],[-1])
Ko = zpk([-zo1,-zo2],[0],[go])
Li_tf = series(Pi,Ki)
Lo_tf = minreal(Po*Pi*Ko*(feedback(1,(Pi*Ki))))
Lu_tf = minreal((Ki*Pi)+(Ko*Po*Pi))
So_tf = feedback(1,Lo_tf)
Su_tf = feedback(1,Lu_tf)
To_tf = feedback(Lo_tf,1)
Tu_tf = feedback(Lu_tf,1)
%% System Information
% Tz = tzero(sys_tf) % transmission zeros
% iopzplot(sys_tf) % pole-zero map
pzmap(sys_tf)
%
% December 16, 2020
%
% Inner-Outer Loop Feedback Structure
% by Soham Sarkar, Armando Rodriguez, Brent Wallace

%%
close all;
clear all;
clc;

% Varying g_o with g_i as a parameter


% NOMINAL PARAMETERS
p   = 5.693;
z   = 4.958;
zi1 = 4.6;
zi2 = 3.9;
zo1 = 1.9;
zo2 = 0.1;
gi = 15.65; % gi >17
go = 1.49;
% s = tf('s')
Pi = zpk([],[p,-p],[1])
Ki = zpk([-zi1,-zi2],[0],[gi])
Li_tf = series(Pi,Ki)
Si_tf = feedback(1,Li_tf)
Po = zpk([z,-z],[0,0],[-1])
Ko = zpk([-zo1,-zo2],[0],[go])
Li_tf = series(Pi,Ki)
Lo_tf = minreal(Po*Pi*Ko*(feedback(1,(Pi*Ki))))
Lu_tf = minreal((Ki*Pi)+(Ko*Po*Pi))
So_tf = feedback(1,Lo_tf)
Su_tf = feedback(1,Lu_tf)
To_tf = feedback(Lo_tf,1)
Tu_tf = feedback(Lu_tf,1)
%% Open Loop
figure(1)
w   = logspace(-1,2, 2000);
[L_i_mag, L_i_phase] = bode(Li_tf,w);
semilogx(w, 20*log10(L_i_mag(1,:)))
ylabel('Magnitude (dB)')
xlabel('freq (rad/sec)')
% title('Magnitude |S_i|')
grid on
hold on
% %% Su
% figure(2)
w   = logspace(-1,2, 2000);
[L_u_mag, L_u_phase] = bode(Lu_tf,w);
semilogx(w, 20*log10(L_u_mag(1,:)))
ylabel('Magnitude (dB)')
xlabel('freq (rad/sec)')
% title('Magnitude |S_u|')
grid on
hold on
% %% So
% figure(2)
w   = logspace(-1,2, 2000);
[L_o_mag, L_o_phase] = bode(Lo_tf,w);
semilogx(w, 20*log10(L_o_mag(1,:)))
ylabel('Magnitude (dB)')
xlabel('freq (rad/sec)')
```

```
% title('Magnitude |S_o|')
title('Open Loop Magnitude Plot - 1.65')
grid on
legend(['L_i'],['L_u'],['L_o'])
%% Sensitivity
figure(2)
w  = logspace(-1,2, 2000);
[S_i_mag, S_i_phase] = bode(Si_tf,w);
semilogx(w, 20*log10(S_i_mag(1,:)))
ylabel('Magnitude (dB)')
xlabel('freq (rad/sec)')
% title('Magnitude |S_i|')
grid on
hold on
% %% Su
% figure(2)
w  = logspace(-1,2, 2000);
[S_u_mag, S_u_phase] = bode(Su_tf,w);
semilogx(w, 20*log10(S_u_mag(1,:)))
ylabel('Magnitude (dB)')
xlabel('freq (rad/sec)')
% title('Magnitude |S_u|')
grid on
hold on
% %% So
% figure(2)
w  = logspace(-1,2, 2000);
[S_o_mag, S_o_phase] = bode(So_tf,w);
semilogx(w, 20*log10(S_o_mag(1,:)))
ylabel('Magnitude (dB)')
xlabel('freq (rad/sec)')
% title('Magnitude |S_o|')
title('Closed Loop Magnitude Plot - 1.65')
grid on
legend(['S_i'],['S_u'],['S_o'])
%%
close all;
clear all;
clc;

% NOMINAL PARAMETERS
p   = 5.693;
z   = 4.958;
zi1 = 3.06;
zi2 = 3.12;
zo1 = 2;
zo2 = 0.4;
gi = 14.79; % gi >17
go = 1.41;
% s = tf('s')
Pi = zpk([],[p,-p],[1])
Ki = zpk([-zi1,-zi2],[0],[gi])
Li_tf = series(Pi,Ki)
Si_tf = feedback(1,Li_tf)
Po = zpk([z,-z],[0,0],[-1])
Ko = zpk([-zo1,-zo2],[0],[go])
Li_tf = series(Pi,Ki)
Lo_tf = minreal(Po*Pi*Ko*(feedback(1,(Pi*Ki))))
Lu_tf = minreal((Ki*Pi)+(Ko*Po*Pi))
So_tf = feedback(1,Lo_tf)
Su_tf = feedback(1,Lu_tf)
To_tf = feedback(Lo_tf,1)
Tu_tf = feedback(Lu_tf,1)
%% Open Loop
figure(1)
w  = logspace(-1,2, 2000);
[L_i_mag, L_i_phase] = bode(Li_tf,w);
semilogx(w, 20*log10(L_i_mag(1,:)))
ylabel('Magnitude (dB)')
xlabel('freq (rad/sec)')
% title('Magnitude |S_i|')
grid on
hold on
% %% Su
% figure(2)
w  = logspace(-1,2, 2000);
[L_u_mag, L_u_phase] = bode(Lu_tf,w);
semilogx(w, 20*log10(L_u_mag(1,:)))
ylabel('Magnitude (dB)')
xlabel('freq (rad/sec)')
% title('Magnitude |S_u|')
grid on
hold on
% %% So
% figure(2)
w  = logspace(-1,2, 2000);
[L_o_mag, L_o_phase] = bode(Lo_tf,w);
semilogx(w, 20*log10(L_o_mag(1,:)))
ylabel('Magnitude (dB)')
xlabel('freq (rad/sec)')
% title('Magnitude |S_o|')
title('Open Loop Magnitude Plot - 2.8')
grid on
legend(['L_i'],['L_u'],['L_o'])
%% Sensitivity
figure(2)
w  = logspace(-1,2, 2000);
```

```
[S_i_mag, S_i_phase] = bode(Si_tf,w);
semilogx(w, 20*log10(S_i_mag(1,:)))
ylabel('Magnitude (dB)')
xlabel('freq (rad/sec)')
% title('Magnitude |S_i|')
grid on
hold on
% %% Su
% figure(2)
w  = logspace(-1,2, 2000);
[S_u_mag, S_u_phase] = bode(Su_tf,w);
semilogx(w, 20*log10(S_u_mag(1,:)))
ylabel('Magnitude (dB)')
xlabel('freq (rad/sec)')
% title('Magnitude |S_u|')
grid on
hold on
% %% So
% figure(2)
w  = logspace(-1,2, 2000);
[S_o_mag, S_o_phase] = bode(So_tf,w);
semilogx(w, 20*log10(S_o_mag(1,:)))
ylabel('Magnitude (dB)')
xlabel('freq (rad/sec)')
% title('Magnitude |S_o|')
title('Closed Loop Magnitude Plot - 2.8')
grid on
legend(['S_i'],['S_u'],['S_o'])

%% Varying gi with go, zo1, zo2, zi1, zi2 as a paramter (fixed)

clc
clear all
close all

% NOMINAL PARAMETERS
p  = 5.693;
z  = 4.958;
% z  = 2.5;
% z  = 1;

% Range of gi
gi_min = 20
gi_max = 25

% Range of go
go_min = 0.8
go_max = 2.0
go_count = 4
go_inc = (go_max-go_min)/go_count

% Range of zi1
zi1_min = 2;
zi1_max = 3;

% Range of zi2
zi2_min = 2.2;
zi2_max = 3.2;

% Range of zo1
zo1_min = 0.4;
zo1_max = 1.4;

% Range of zo2
zo2_min = 0.08;
zo2_max = 0.4;


% gi = 4;
% fixed parameters
gi = gi_min;
zi1 = zi1_min;
zi2 = zi2_min;
zo1 = zo1_max;
zo2 = zo2_max;

s = tf('s');
Po = zpk([z,-z],[0,0],[-1]);


counter = 0;

for go = go_min:go_inc:go_max
    counter = counter + 1;
    go_values(:,counter) = go;


    % transfer function analogy
    Pi = zpk([],[p,-p],[1])
    Ki = zpk([-zi1,-zi2],[0],[gi])
    Li_tf = series(Pi,Ki)

    % inner loop state space
    Li = ss(Li_tf)
    ai = Li.a
    bi = Li.b
```

197

```
        ci = Li.c
        di = Li.d

%      figurecounter = 1;

        % inner loop poles
        damp(Li)

        % inner loop plots

        % figure (figurecounter)
        % w  = logspace(-1,3, 2000);
        % bode(Li,w)
        % ylabel('dB')
        % xlabel('freq (rad/sec)')
        % title('Inner Loop Magnitude (Li)')
        % grid on
        % hold on
        % figurecounter = figurecounter + 1;


        figure (1)
        w  = logspace(-1,3, 2000);
        [L_i_mag, L_i_phase] = bode(Li_tf,w);
        semilogx(w, 20*log10(L_i_mag(1,:)))
        ylabel('Magnitude (dB)')
        xlabel('freq (rad/sec)')
        title(['Inner Loop Magnitude |L_i| for z = ',num2str(z),',p = ',num2str(p)...
            ,',zo1 = ',num2str(zo1),',zo2 = ',num2str(zo2),',zi1 = ',num2str(zi1)...
            ,',zi2 = ',num2str(zi2)])
        grid on
%      hold on
%      figurecounter = figurecounter + 1;

        figure (2)
        w  = logspace(-1,3, 2000);
        [L_i_mag, L_i_phase] = bode(Li_tf,w);
        semilogx(w, L_i_phase(1,:))
        ylabel('Angle (deg)')
        xlabel('freq (rad/sec)')
        title(['Inner Loop Phase \angle L_i for z = ',num2str(z),',p = ',num2str(p)...
            ,',zo1 = ',num2str(zo1),',zo2 = ',num2str(zo2),',zi1 = ',num2str(zi1)...
            ,',zi2 = ',num2str(zi2)])
        grid on

        % figure (figurecounter)
        % rlocus(Li)
        % title('Root Locus')
        % grid on
        % hold on
        % figurecounter = figurecounter + 1;
        %
        % figure (figurecounter)
        % nyquist(Li)
        % title('Nyquist')
        % grid on
        % hold on
        % figurecounter = figurecounter + 1;


        figure (3)
        w  = logspace(-1,3, 2000);
        sen_i = feedback(1,Li_tf)
        [sen_i_mag, sen_i_phase] = bode(sen_i,w);
        [hinf_sen_i,fpeak_sen_i] = hinfnorm(sen_i)
        semilogx(w, 20*log10(sen_i_mag(1,:)))
        ylabel('Magnitude (dB)')
        xlabel('freq (rad/sec)')
        title(['Inner Loop Sensitivity |S_i| for z = ',num2str(z),',p = ',num2str(p)...
            ,',zo1 = ',num2str(zo1),',zo2 = ',num2str(zo2),',zi1 = ',num2str(zi1)...
            ,',zi2 = ',num2str(zi2)])
        grid on
%      hold on
%      figurecounter = figurecounter + 1;


        figure (4)
        w  = logspace(-1,3, 2000);
        comp_sen_i = feedback(Li_tf,1)
        [comp_sen_i_mag, comp_sen_i_phase] = bode(comp_sen_i,w);
        [hinf_comp_sen_i,fpeak_comp_sen_i] = hinfnorm(comp_sen_i)
        semilogx(w, 20*log10(comp_sen_i_mag(1,:)))
        ylabel('Magnitude (dB)')
        xlabel('freq (rad/sec)')
        title(['Inner Loop Complementary Sensitivity |T_i| for z = ',num2str(z)...
            ,',p = ',num2str(p)...
            ,',zo1 = ',num2str(zo1),',zo2 = ',num2str(zo2),',zi1 = ',num2str(zi1)...
            ,',zi2 = ',num2str(zi2)])
        grid on
%      hold on
%      figurecounter = figurecounter + 1;


        figure (5)
        semilogx(w, 20*log10(comp_sen_i_mag(1,:)), 'r')
        hold on
```

```matlab
        semilogx(w, 20*log10(sen_i_mag(1,:)), 'b')
        ylabel('Magnitude (dB)')
        xlabel('freq (rad/sec)')
        title(['Inner Loop Sensitivity |S_i| & Complementary Sensitivity |T_i|for z = ',...
            num2str(z),',p = ',num2str(p)...
            ,',zo1 = ',num2str(zo1),',zo2 = ',num2str(zo2),',zi1 = ',num2str(zi1)...
            ,',zi2 = ',num2str(zi2)])
        grid on
%       hold on
        legend({'Complementary Sensitivity','Sensitivity'}...
            ,'Location','Southwest','NumColumns',1)

%       figurecounter = figurecounter + 1;

        % Outer Loop at Output

        % transfer function
%       Lo_tf = zpk([z],[0,-(gi-p)],[-go])
        Ko = zpk([-zo1,-zo2],[0],[go]);
        Lo_tf = minreal(Po*Pi*Ko*(feedback(1,(Pi*Ki))));


        % outer loop state space at output
        Lo = ss(Lo_tf)
        ao = Lo.a
        bo = Lo.b
        co = Lo.c
        do = Lo.d


        % outer loop poles at output
        damp(Lo)

        % outer loop plots at output
        figure (6)
        w  = logspace(-1,3, 2000);
        [L_o_mag, L_o_phase] = bode(Lo_tf,w);
        semilogx(w, 20*log10(L_o_mag(1,:)))
        ylabel('Magnitude (dB)')
        xlabel('freq (rad/sec)')
        title(['Loop Magnitude at Output |L_o| (g_i = ',...
            num2str(gi),'), for z = ',num2str(z),',p = ',num2str(p)...
            ,',zo1 = ',num2str(zo1),',zo2 = ',num2str(zo2),',zi1 = ',num2str(zi1)...
            ,',zi2 = ',num2str(zi2)])
        grid on
        hold on
%       figurecounter = figurecounter + 1;


        figure (7)
        w  = logspace(-1,3, 2000);
        [L_o_mag, L_o_phase] = bode(Lo_tf,w);
        semilogx(w, L_o_phase(1,:)-360)
        ylabel('Angle (deg)')
        xlabel('freq (rad/sec)')
        title(['\angle L_o : Loop Phase at Output (g_i = ',...
            num2str(gi),'), for z = ',num2str(z),',p = ',num2str(p)...
            ,',zo1 = ',num2str(zo1),',zo2 = ',num2str(zo2),',zi1 = ',num2str(zi1)...
            ,',zi2 = ',num2str(zi2)])
        grid on
        hold on
%       figurecounter = figurecounter + 1;


%       figure (6)
        figure (8)
        w  = logspace(-1,3, 2000);
        sen_o = feedback(1,Lo_tf)
        [sen_o_mag, sen_o_phase] = bode(sen_o,w);
        [hinf_sen_o,fpeak_sen_o] = hinfnorm(sen_o)
        hinf_so(:,counter) = hinf_sen_o
        semilogx(w, 20*log10(sen_o_mag(1,:)))
        ylabel('Magnitude (dB)')
        xlabel('freq (rad/sec)')
        title(['|S_o|: Loop Sensitivity at Output (g_i = '...
            ,num2str(gi),'), for z = ',num2str(z),',p = ',num2str(p)...
            ,',zo1 = ',num2str(zo1),',zo2 = ',num2str(zo2),',zi1 = ',num2str(zi1)...
            ,',zi2 = ',num2str(zi2)])
        grid on
        hold on
%       figurecounter = figurecounter + 1;


%       figure (7)
        figure (9)
        w  = logspace(-1,3, 2000);
        comp_sen_o = feedback(Lo_tf,1)
        [comp_sen_o_mag, comp_sen_o_phase] = bode(comp_sen_o,w);
        [hinf_comp_sen_o,fpeak_sen_o] = hinfnorm(comp_sen_o)
        hinf_cso(:,counter) = hinf_comp_sen_o
        semilogx(w, 20*log10(comp_sen_o_mag(1,:)))
        ylabel('Magnitude (dB)')
        xlabel('freq (rad/sec)')
        title(['|T_o|: Loop Complementary Sensitivity at Output (g_i = ',...
            num2str(gi),'), for z = ',num2str(z),',p = ',num2str(p)...
            ,',zo1 = ',num2str(zo1),',zo2 = ',num2str(zo2),',zi1 = ',num2str(zi1)...
```

```
                 ,',zi2 = ',num2str(zi2)])
         grid on
         hold on

%        figurecounter = figurecounter + 1;


%        figure (8)
%        semilogx(w, 20*log10(comp_sen_o_mag(1,:)), 'r')
%        hold on
%        semilogx(w, 20*log10(sen_o_mag(1,:)), 'b')
%        ylabel('dB')
%        xlabel('freq (rad/sec)')
%        title('Outer Loop Sensitivity (S_o) & Complementary Sensitivity at Output(T_o)')
%        grid on
%        hold on
%        legend({'Complementary Sensitivity','Sensitivity'},'Location','Southwest','NumColumns',1)

%        figurecounter = figurecounter + 1;


         % Outer Loop at Controls

         Lu_tf = minreal((Ki*Pi)+(Ko*Po*Pi));

%          % transfer function
%          num_Lu = [(gi-go), go*z]
%          den_Lu = [1 -p 0]
%          Lu_tf = tf(num_Lu,den_Lu)
%          Lu_tf = zpk(Lu_tf)
%
         % outer loop state space at controls
         Lu = ss(Lu_tf)
         au = Lu.a
         bu = Lu.b
         cu = Lu.c
         du = Lu.d


         % Outer Loop poles at controls
         damp(Lo)

         % Outer Loop poles at controls
%          figure (9)
         figure (10)
         w  = logspace(-1,3, 2000);
         [L_u_mag, L_u_phase] = bode(Lu_tf,w);
         semilogx(w, 20*log10(L_u_mag(1,:)))
         ylabel('Magnitude (dB)')
         xlabel('freq (rad/sec)')
         title(['|L_u|: Loop Magnitude at Controls (g_i = ',...
             num2str(gi),'), for z = ',num2str(z),',p = ',num2str(p)...
             ,',zo1 = ',num2str(zo1),',zo2 = ',num2str(zo2),',zi1 = ',num2str(zi1)...
             ,',zi2 = ',num2str(zi2)])
         grid on
         hold on
%        figurecounter = figurecounter + 1;


         figure (11)
         w  = logspace(-1,3, 2000);
         [L_u_mag, L_u_phase] = bode(Lu_tf,w);
         semilogx(w, L_u_phase(1,:))
         ylabel('Angle (deg)')
         xlabel('freq (rad/sec)')
         title(['\angle L_u: Loop Phase at Controls (g_i = ',...
             num2str(gi),'), for z = ',num2str(z),',p = ',num2str(p)...
             ,',zo1 = ',num2str(zo1),',zo2 = ',num2str(zo2),',zi1 = ',num2str(zi1)...
             ,',zi2 = ',num2str(zi2)])
         grid on
         hold on


         figure (12)
         w  = logspace(-1,3, 2000);
         sen_u = feedback(1,Lu_tf)
         [sen_u_mag, sen_u_phase] = bode(sen_u,w);
         [hinf_sen_u,fpeak_sen_u] = hinfnorm(sen_u)
         hinf_su(:,counter) = hinf_sen_u
         semilogx(w, 20*log10(sen_u_mag(1,:)))
         ylabel('Magnitude (dB)')
         xlabel('freq (rad/sec)')
         title(['|S_u|: Loop Sensitivity at Controls (g_i = ',...
             num2str(gi),'), for z = ',num2str(z),',p = ',num2str(p)...
             ,',zo1 = ',num2str(zo1),',zo2 = ',num2str(zo2),',zi1 = ',num2str(zi1)...
             ,',zi2 = ',num2str(zi2)])
         grid on
         hold on
%        figurecounter = figurecounter + 1;



         figure (13)
         w  = logspace(-1,3, 2000);
         comp_sen_u = feedback(Lu_tf,1)
         [comp_sen_u_mag, comp_sen_u_phase] = bode(comp_sen_u,w);
         [hinf_comp_sen_u,fpeak_comp_sen_u] = hinfnorm(comp_sen_u)
```

```
        hinf_csu(:,counter) = hinf_comp_sen_u
        semilogx(w, 20*log10(comp_sen_u_mag(1,:)))
        ylabel('Magnitude (dB)')
        xlabel('freq (rad/sec)')
        title(['|T_u|: Loop Complementary Sensitivity at Controls (g_i = ',...
            num2str(gi),'), for z = ',num2str(z),',p = ',num2str(p)...
            ,',zo1 = ',num2str(zo1),',zo2 = ',num2str(zo2),',zi1 = ',num2str(zi1)...
            ,',zi2 = ',num2str(zi2)])
        grid on
        hold on
%       figurecounter = figurecounter + 1;


        figure (14)
        w  = logspace(-1,3, 2000);
        comp_sen_u_o = (comp_sen_u/comp_sen_o)
        [comp_sen_u_o_mag, comp_sen_u_o_phase] = bode(comp_sen_u_o,w);
%       [hinf_comp_sen_u,fpeak_comp_sen_u] = hinfnorm(comp_sen_u)
%       hinf_csu(:,counter) = hinf_comp_sen_u
%       comp_sen_u_o = tf([gi-go,go*z],[-go go*z])
%       [comp_sen_u_o_mag, comp_sen_u_o_phase] = bode(comp_sen_u_o,w);


%       comp_sen_u_o = 1 + ((sen_o/comp_sen_o)*comp_sen_i);
%       [comp_sen_u_o_mag, comp_sen_u_o_phase] = bode(comp_sen_u_o,w);
        semilogx(w, 20*log10(comp_sen_u_o_mag(1,:)))
        ylabel('Magnitude (dB)')
        xlabel('freq (rad/sec)')
        title(['$\frac{|T_u|}{|T_o|} > 1 \; \forall \; \omega > 0 \; (p = ',num2str(p)...
            ,') \; for \; z = ',num2str(z),' $'],'Interpreter','latex')
        grid on
        hold on

%       figure (12)
%       semilogx(w, 20*log10(comp_sen_u_mag(1,:)), 'r')
%       hold on
%       semilogx(w, 20*log10(sen_u_mag(1,:)), 'b')
%       ylabel('dB')
%       xlabel('freq (rad/sec)')
%       title('Outer Loop Sensitivity (S_u) & Complementary Sensitivity at Output(T_u)')
%       grid on
%       hold on
%       legend({'Complementary Sensitivity','Sensitivity'},'Location','Southwest','NumColumns',1)

%       figurecounter = figurecounter + 1;



        % Control Response to Step Ref. Command at controls
        figure (15)
        tinc            = 0.0005;
        tmax = 10;
        t               = 0:tinc:tmax;
        plot(t,step(sen_u,t));
        xlabel('Time (sec)')
        ylabel(['u(t)'])
        title(['Step S_u: Control Response u_p to Step Input Disturbance d_i (g_i = '...
            ,num2str(gi),'), for z = ',num2str(z),',p = ',num2str(p),',zo1 = ',...
            num2str(zo1),',zo2 = ',num2str(zo2),',zi1 = ',num2str(zi1),',zi2 = ',...
            num2str(zi2)])
        grid on
        hold on
%       figurecounter = figurecounter + 1;

        % Output Response to Step Ref. Command at Controls
        figure (16)
        tinc            = 0.0005;
        tmax = 10;
        t               = 0:tinc:tmax;
        plot(t,step(comp_sen_u,t))
        xlabel('Time (sec)')
        ylabel(['y(t)'])
        title(['Step T_u: Control Response u to Step Input Disturbance d_i (g_i = '...
            ,num2str(gi),'), for z = ',num2str(z),',p = ',num2str(p),',zo1 = ',...
            num2str(zo1),',zo2 = ',num2str(zo2),',zi1 = ',num2str(zi1),',zi2 = ',...
            num2str(zi2)])
        grid on
        hold on
%       figurecounter = figurecounter + 1;

        % Control Response to Step Ref. Command at Output
        figure (17)
        tinc            = 0.0005;
        tmax = 10;
        t               = 0:tinc:tmax;
        plot(t,step(sen_o,t));
        xlabel('Time (sec)')
        ylabel(['u(t)'])
        title(['Step S_o: Error Response e to Step Ref. Command r (g_i = '...
            ,num2str(gi),'), for z = ',num2str(z),',p = ',num2str(p),',zo1 = ',...
            num2str(zo1),',zo2 = ',num2str(zo2),',zi1 = ',num2str(zi1),',zi2 = ',...
            num2str(zi2)])
        grid on
        hold on
%       figurecounter = figurecounter + 1;

        % Output Response to Step Ref. Command at Output
        figure (18)
```

```
        tinc            = 0.0005;
        tmax = 10;
        t               = 0:tinc:tmax;
        plot(t,step(comp_sen_o,t))
        xlabel('Time (sec)')
        ylabel(['y(t)'])
        title(['Step T_o: Output Response y to Step Ref. Command r (g_i = '...
            ,num2str(gi),'), for z = ',num2str(z),',p = ',num2str(p),',zo1 = ',...
            num2str(zo1),',zo2 = ',num2str(zo2),',zi1 = ',num2str(zi1),',zi2 = ',...
            num2str(zi2)])
        grid on
        hold on

%    figurecounter = figurecounter + 1;

% impulse

% Control Response to Step Ref. Command at controls
        figure (19)
        tinc            = 0.0005;
        tmax = 10;
        t               = 0:tinc:tmax;
        plot(t,impulse(sen_u,t));
        xlabel('Time (sec)')
        ylabel(['u(t)'])
        title(['Impulse S_u: Control Response u_p to Impulse Input Disturbance d_i (g_i = ',...
            num2str(gi),'), for z = ',num2str(z),',p = ',...
            num2str(p),',zo1 = ',num2str(zo1),',zo2 = ',...
            num2str(zo2),',zi1 = ',num2str(zi1),',zi2 = ',num2str(zi2)])
        grid on
        hold on
%    figurecounter = figurecounter + 1;

        % Output Response to Step Ref. Command at Controls
        figure (20)
        tinc            = 0.0005;
        tmax = 10;
        t               = 0:tinc:tmax;
        plot(t,impulse(comp_sen_u,t))
        xlabel('Time (sec)')
        ylabel(['y(t)'])
        title(['Impulse T_u: Output Response y to Impulse Ref. Command r (g_i = '...
            ,num2str(gi),'), for z = ',num2str(z),',p = ',num2str(p),',zo1 = ',...
            num2str(zo1),',zo2 = ',num2str(zo2),',zi1 = ',num2str(zi1),',zi2 = ',...
            num2str(zi2)])
        grid on
        hold on
%    figurecounter = figurecounter + 1;

        % Control Response to Step Ref. Command at Output
        figure (21)
        tinc            = 0.0005;
        tmax = 10;
        t               = 0:tinc:tmax;
        plot(t,impulse(sen_o,t));
        xlabel('Time (sec)')
        ylabel(['u(t)'])
        title(['Impulse S_o: Error Response e to Impulse Ref. Command r (g_i = '...
            ,num2str(gi),'), for z = ',num2str(z),',p = ',num2str(p),',zo1 = ',...
            num2str(zo1),',zo2 = ',num2str(zo2),',zi1 = ',num2str(zi1),',zi2 = ',...
            num2str(zi2)])
        grid on
        hold on
%    figurecounter = figurecounter + 1;

        % Output Response to Step Ref. Command at Output
        figure (22)
        tinc            = 0.0005;
        tmax = 10;
        t               = 0:tinc:tmax;
        plot(t,impulse(comp_sen_o,t))
        xlabel('Time (sec)')
        ylabel(['y(t)'])
        title(['Impulse T_o: Control Response u to Impulse Input Disturbance d_i (g_i = '...
            ,num2str(gi),'), for z = ',num2str(z),',p = ',num2str(p),',zo1 = ',...
            num2str(zo1),',zo2 = ',num2str(zo2),',zi1 = ',num2str(zi1),',zi2 = ',...
            num2str(zi2)])
        grid on
        hold on


end

% figure (2)
% yline(6.02 ,'-.b','6.02 db');

figure (3)
yline(6.02 ,'-.b','6.02 db');

figure (4)
yline(6.02 ,'-.b','6.02 db');

figure (5)
yline(6.02 ,'-.b','6.02 db');

figure (6)
% yline(6.02 ,'-.b','6.02 db');
```

```matlab
legend(['g_o = ' num2str(go_values(1,1))],['g_o = ' num2str(go_values(1,2))]...
    ,['g_o = ' num2str(go_values(1,3))],['g_o = ' num2str(go_values(1,4))]...
    ,['g_o = ' num2str(go_values(1,5))],'Location','Southwest','NumColumns',1);

figure (7)
% yline(6.02 ,'-.b','6.02 db');
legend(['g_o = ' num2str(go_values(1,1))],['g_o = ' num2str(go_values(1,2))]...
    ,['g_o = ' num2str(go_values(1,3))],['g_o = ' num2str(go_values(1,4))]...
    ,['g_o = ' num2str(go_values(1,5))],'Location','Southwest','NumColumns',1);

figure (8)
yline(6.02 ,'-.b','6.02 db');
legend(['g_o = ' num2str(go_values(1,1))],['g_o = ' num2str(go_values(1,2))]...
    ,['g_o = ' num2str(go_values(1,3))],['g_o = ' num2str(go_values(1,4))]...
    ,['g_o = ' num2str(go_values(1,5))],'Location','Southwest','NumColumns',1);

figure (9)
yline(6.02 ,'-.b','6.02 db');
legend(['g_o = ' num2str(go_values(1,1))],['g_o = ' num2str(go_values(1,2))]...
    ,['g_o = ' num2str(go_values(1,3))],['g_o = ' num2str(go_values(1,4))]...
    ,['g_o = ' num2str(go_values(1,5))],'Location','Southwest','NumColumns',1);

figure (10)
% yline(6.02 ,'-.b','6.02 db');
legend(['g_o = ' num2str(go_values(1,1))],['g_o = ' num2str(go_values(1,2))]...
    ,['g_o = ' num2str(go_values(1,3))],['g_o = ' num2str(go_values(1,4))]...
    ,['g_o = ' num2str(go_values(1,5))],'Location','Southwest','NumColumns',1);

figure (11)
% yline(6.02 ,'-.b','6.02 db');
legend(['g_o = ' num2str(go_values(1,1))],['g_o = ' num2str(go_values(1,2))]...
    ,['g_o = ' num2str(go_values(1,3))],['g_o = ' num2str(go_values(1,4))]...
    ,['g_o = ' num2str(go_values(1,5))],'Location','Southwest','NumColumns',1);

figure (12)
yline(6.02 ,'-.b','6.02 db');
legend(['g_o = ' num2str(go_values(1,1))],['g_o = ' num2str(go_values(1,2))]...
    ,['g_o = ' num2str(go_values(1,3))],['g_o = ' num2str(go_values(1,4))]...
    ,['g_o = ' num2str(go_values(1,5))],'Location','Southwest','NumColumns',1);

figure (13)
yline(6.02 ,'-.b','6.02 db');
legend(['g_o = ' num2str(go_values(1,1))],['g_o = ' num2str(go_values(1,2))]...
    ,['g_o = ' num2str(go_values(1,3))],['g_o = ' num2str(go_values(1,4))]...
    ,['g_o = ' num2str(go_values(1,5))],'Location','Southwest','NumColumns',1);

figure (14)
legend(['g_o = ' num2str(go_values(1,1))],['g_o = ' num2str(go_values(1,2))]...
    ,['g_o = ' num2str(go_values(1,3))],['g_o = ' num2str(go_values(1,4))]...
    ,['g_o = ' num2str(go_values(1,5))],'Location','Southwest','NumColumns',1);

figure (15)
legend(['g_o = ' num2str(go_values(1,1))],['g_o = ' num2str(go_values(1,2))]...
    ,['g_o = ' num2str(go_values(1,3))],['g_o = ' num2str(go_values(1,4))]...
    ,['g_o = ' num2str(go_values(1,5))],'Location','Southwest','NumColumns',1);

figure (16)
legend(['g_o = ' num2str(go_values(1,1))],['g_o = ' num2str(go_values(1,2))]...
    ,['g_o = ' num2str(go_values(1,3))],['g_o = ' num2str(go_values(1,4))]...
    ,['g_o = ' num2str(go_values(1,5))],'Location','Southwest','NumColumns',1);

figure (17)
legend(['g_o = ' num2str(go_values(1,1))],['g_o = ' num2str(go_values(1,2))]...
    ,['g_o = ' num2str(go_values(1,3))],['g_o = ' num2str(go_values(1,4))]...
    ,['g_o = ' num2str(go_values(1,5))],'Location','Southwest','NumColumns',1);

figure (18)
legend(['g_o = ' num2str(go_values(1,1))],['g_o = ' num2str(go_values(1,2))]...
    ,['g_o = ' num2str(go_values(1,3))],['g_o = ' num2str(go_values(1,4))]...
    ,['g_o = ' num2str(go_values(1,5))],'Location','Southwest','NumColumns',1);

figure (19)
legend(['g_o = ' num2str(go_values(1,1))],['g_o = ' num2str(go_values(1,2))]...
    ,['g_o = ' num2str(go_values(1,3))],['g_o = ' num2str(go_values(1,4))]...
    ,['g_o = ' num2str(go_values(1,5))],'Location','Southwest','NumColumns',1);

figure (20)
legend(['g_o = ' num2str(go_values(1,1))],['g_o = ' num2str(go_values(1,2))]...
    ,['g_o = ' num2str(go_values(1,3))],['g_o = ' num2str(go_values(1,4))]...
    ,['g_o = ' num2str(go_values(1,5))],'Location','Southwest','NumColumns',1);

figure (21)
legend(['g_o = ' num2str(go_values(1,1))],['g_o = ' num2str(go_values(1,2))]...
    ,['g_o = ' num2str(go_values(1,3))],['g_o = ' num2str(go_values(1,4))]...
    ,['g_o = ' num2str(go_values(1,5))],'Location','Southwest','NumColumns',1);

figure (22)
legend(['g_o = ' num2str(go_values(1,1))],['g_o = ' num2str(go_values(1,2))]...
    ,['g_o = ' num2str(go_values(1,3))],['g_o = ' num2str(go_values(1,4))]...
    ,['g_o = ' num2str(go_values(1,5))],'Location','Southwest','NumColumns',1);

%% Benchmark Example 8 cart inverted
%% Cart Inverted Pendulum Analysis
clc
clear all
close all
```

```
%% Derivation of Equations (Matthew Peter Kelly)
syms pos(t) pos_dot(t) pos_dbldot(t) theta(t) theta_dot(t) theta_dbldot(t);
syms m M l b g I u;

Xp = pos(t) + l*sin(theta(t));
Yp = l*cos(theta(t));
Xp_dot = diff(Xp);
Yp_dot = diff(Yp);


Xp_dot = subs(Xp_dot,[diff(pos(t),t), diff(theta(t),t)],[pos_dot(t) theta_dot(t)]);
Yp_dot = subs(Yp_dot,[diff(pos(t),t), diff(theta(t),t)],[pos_dot(t) theta_dot(t)]);

pos_dot(t) = subs(pos_dot(t),diff(pos(t),t),pos_dot(t));
T =  ((1/2)*M*pos_dot(t)) + ((1/2)*m*((Xp_dot^2) + (Yp_dot^2)));

T = ((1/2)*M*pos_dot(t)^2)+((1/2)*(m)*(pos_dot(t)^2))+((1/2)*m...
    *(2*l*pos_dot(t)*theta_dot(t)*cos(theta(t))))+((1/2)*m*(l^2)...
    *(theta_dot^2))+((1/2)*I*(theta_dot^2))
U = m*g*l*cos(theta(t));
L = T - U;

% First Equation with respect to theta1(t)
dL_dpos_dot(t) = functionalDerivative(L,pos_dot(t));

dL_dpos(t) = functionalDerivative(L,pos(t));

% Second Equation with respect to theta
dL_dtheta_dot(t) = functionalDerivative(L,theta_dot(t));

dL_dtheta(t) = functionalDerivative(L,theta(t));


dL_dpos_dot_dt = jacobian(dL_dpos_dot(t),t);
dL_dpos_dot_dt = subs(dL_dpos_dot_dt,[diff(pos(t),t),...
    diff(pos_dot(t),t), diff(theta(t),t), diff(theta_dot(t),t)]...
    ,[pos_dot(t),pos_dbldot(t),theta_dot(t),theta_dbldot(t)]);


eqn1 = dL_dpos_dot_dt - dL_dpos(t) + b*theta_dot(t) - u;

% theta_1dbldot(t) = solve(eqn1,theta_1dbldot(t),'Real',true);

fprintf('Equation 1: \n');
disp(eqn1)
% fprintf('Equation 1 in terms of theta_1dbldot(t): \n');
% disp(theta_1dbldot(t))

% Remember this trick that you did, apply for all
dL_dtheta_dot_dt = jacobian(dL_dtheta_dot(t),t);
dL_dtheta_dot_dt = subs(dL_dtheta_dot_dt,[diff(pos(t),t),...
    diff(pos_dot(t),t), diff(theta(t),t), diff(theta_dot(t),t)]...
    ,[pos_dot(t),pos_dbldot(t),theta_dot(t),theta_dbldot(t)]);
eqn2 = dL_dtheta_dot_dt - dL_dtheta(t);


fprintf('Equation 2: \n');
disp(eqn2)
%% Finding the Langrange Equations with variables
syms x th x_dot th_dot x_dbldot th_dbldot;

eqn1 = subs(eqn1,[pos(t),theta(t),pos_dot(t),theta_dot(t),...
    pos_dbldot(t),theta_dbldot(t)], [x,th,x_dot,th_dot,...
    x_dbldot,th_dbldot]);
fprintf('Equation of Motion 1: \n');
disp(eqn1)

eqn2 = subs(eqn2,[pos(t),theta(t),pos_dot(t),theta_dot(t),...
    pos_dbldot(t),theta_dbldot(t)], [x,th,x_dot,th_dot,...
    x_dbldot,th_dbldot]);
fprintf('Equation of Motion 2: \n');
disp(eqn2)

%% Linearization
syms u;
[x_dbldot,th_dbldot]= solve([eqn1,eqn2],[x_dbldot,th_dbldot])

q = [x x_dot th th_dot];
u = [u];

% Non Linear Function
f1 = x_dot;
f2 = x_dbldot;
f3 = th_dot;
f4 = th_dbldot;

F = [f1 f2 f3 f4];

% Jacobian
J = jacobian(F,q)

% Equilibrium Points
q_e = [0 0 0 0];
J = subs(J,q,q_e);
disp(J);
```

```
% System Matrix
A = J;
% Input Matrix
B = jacobian(F,u);
B = subs(B,q,q_e);




%% Plant Nominal Values (hernandez)
m = 0.267; % in Kilograms
L = 2.016; % in metres
l = L/2; % in metres
b = 0.0015; % in Kg m^2/s
k = 0; % in Kg/s^2
g = 9.8; % in m/s^2
I = 0; % in Nm/rad^2/sec^2
M = 0.8; %in Kilograms

%% Plant Nominal Values (ogata)
m = 0.21; % in Kilograms
L = 0.61; % in metres
l = L/2; % in metres
b = 0; % in Kg m^2/s
k = 0; % in Kg/s^2
g = 9.8; % in m/s^2
I = 0.006; % in Nm/rad^2/sec^2
M = 0.455; %in Kilograms

%% Plant Nominal Values (vignesh)
m = 0.20; % in Kilograms
% L = 0.115; % in metres
l = 0.115; % in metres
b = 0.0005; % in Kg m^2/s
k = 0; % in Kg/s^2
g = 9.81; % in m/s^2
I = 0.0002545; % in Nm/rad^2/sec^2
M = 1.21; %in Kilograms

%% State Space
Ap = double(subs(A)) %System Matrix
Bp = double(subs(B)) % Input Matrix
% Cp = [1 0; 0 1];
% Cp = [1 0 0 0;
%        0 0 1 0] % Output Matrix
Cp = [1 0 0 0] % Output Matrix
% Dp = [0; 0];
Dp = zeros(size(Cp,1),size(Bp,2)) % Feedforward Matrix
%% Scaling the Matrices
% FACTS ON SCALING: Scaling affects the shape of singular value plots
%                   (i.e. multivariable frequency responses).
%                   It does not alter pole locations, zero locations.
%                   It does alter directionality information.
%
% Scaling Matrices
%
%   unew = su uold    (foot pounds, foot pounds)
%   xnew = sx xold    (degrees, degrees, degrees/sec, degrees/sec)
%   ynew = sy yold    (degrees, degrees)
%-
%
% 0.2247 lb/N          <--- Convert N to lb
% 4.45  N/lb
% 3.281 ft/m           <--- Convert m to ft
% 0.3048 m/ft
% 180 degrees/pi rad   <--- Convert rad to deg
%
r2d = 180/pi;                                       % Convert radians to degrees
su  = diag([1] );    % Convert N-m to lb-ft
sx  = diag( [ 1, 1, r2d, r2d ]);                   % Convert radians to degrees
% sy  = diag( [ 1 1,r2d,r2d ] );
sy  = diag( [1] );                % Convert radians to degrees


%
% Scaled System Dynamics
%
%
% u = [ torque_1    (lb-ft)   torque_2     (lb-ft) ]
% x = [ theta_1     (deg)     theta_2      (deg)    dot_theta_1 (deg/sec)    dot_theta_2 (deg/sec) ]
% y = [ theta_1     (deg)     theta_2      (deg) ]
%
Ap = sx*Ap*inv(sx)
Bp = sx*Bp*inv(su)
Cp = sy*Cp*inv(sx)
Dp = sy*Dp*inv(su)
%% State Space Representation
states = {'x','\theta','x_dot','\theta_dot'}; % States
inputs = {'force'}; % Inputs
% outputs = {'x','theta_dot'}
outputs = {'x'}; % Outputs
sys_ss = ss(Ap,Bp,Cp,Dp,'statename',states,'inputname',inputs,'outputname',outputs) % State Space
sys_tf = zpk(sys_ss) % Transfer Function
%% Relations to different designs

% Varying g_o with g_i as a parameter


% NOMINAL PARAMETERS
```

```
p   = max(pole(sys_tf));
z   = max(tzero(sys_tf));


%% Relations to different designs
close all;
clear all;
clc;

% Varying g_o with g_i as a parameter


% NOMINAL PARAMETERS
p   = 5.693;
z   = 4.958;
zi1_values = [14,0.001,2,2]
zi2_values = [12.2,28,2.2,2.2];
zo1_values = [0.001,0.001,0.4,0.5];
zo2_values = [28,28,1.4,1.6];
gi_values = [16.95,34.5,17.85,50]; % gi >17
go_values = [0.02,0.02,0.949,12.17];


for i = 1:length(go_values)
    gi = gi_values(:,i);
    zi1 = zi1_values(:,i);
    zi2 = zi2_values(:,i);
    go = go_values(:,i);
    zo1 = zo1_values(:,i);
    zo2 = zo2_values(:,i);


    % transfer function analogy
    Pi = zpk([],[p,-p],[1])
    Ki = zpk([-zi1,-zi2],[0],[gi])
    Li_tf = series(Pi,Ki)


    figure (1)
    w   = logspace(-1,3, 2000);
    [L_i_mag, L_i_phase] = bode(Li_tf,w);
    semilogx(w, 20*log10(L_i_mag(1,:)))
    ylabel('Magnitude (dB)')
    xlabel('freq (rad/sec)')
    title(['Inner Loop Magnitude |L_i| '])
    grid on
    hold on


    figure (2)
    w   = logspace(-1,3, 2000);
    sen_i = feedback(1,Li_tf)
    [sen_i_mag, sen_i_phase] = bode(sen_i,w);
    semilogx(w, 20*log10(sen_i_mag(1,:)))
    ylabel('Magnitude (dB)')
    xlabel('freq (rad/sec)')
    title(['Inner Loop Sensitivity |S_i| '])
    grid on
    hold on


    figure (3)
    w   = logspace(-1,3, 2000);
    comp_sen_i = feedback(Li_tf,1)
    [comp_sen_i_mag, comp_sen_i_phase] = bode(comp_sen_i,w);
    semilogx(w, 20*log10(comp_sen_i_mag(1,:)))
    ylabel('Magnitude (dB)')
    xlabel('freq (rad/sec)')
    title(['Inner Loop Complementary Sensitivity |T_i| '])
    grid on
    hold on

    % transfer function
    Po = zpk([z,-z],[0,0],[-1])
    Ko = zpk([-zo1,-zo2],[0],[go])
    Lo_tf = minreal(Po*Pi*Ko*(feedback(1,(Pi*Ki))));

    % outer loop plots at output
    figure (4)
    w   = logspace(-1,3, 2000);
    [L_o_mag, L_o_phase] = bode(Lo_tf,w);
    semilogx(w, 20*log10(L_o_mag(1,:)))
    ylabel('Magnitude (dB)')
    xlabel('freq (rad/sec)')
    title(['Loop Magnitude at Output |L_o| '])
    grid on
    hold on

    figure (5)
    w   = logspace(-1,3, 2000);
    sen_o = feedback(1,Lo_tf)
    [sen_o_mag, sen_o_phase] = bode(sen_o,w);
    semilogx(w, 20*log10(sen_o_mag(1,:)))
    ylabel('Magnitude (dB)')
    xlabel('freq (rad/sec)')
    title(['|S_o|: Loop Sensitivity at Output '])
    grid on
```

```
%% Relations to different designs
```

```
    hold on

    figure (6)
    w  = logspace(-1,3, 2000);
    comp_sen_o = feedback(Lo_tf,1)
    [comp_sen_o_mag, comp_sen_o_phase] = bode(comp_sen_o,w);
    semilogx(w, 20*log10(comp_sen_o_mag(1,:)))
    ylabel('Magnitude (dB)')
    xlabel('freq (rad/sec)')
    title(['|T_o|: Loop Complementary Sensitivity at Output '])
    grid on
    hold on


    % Loop at Controls
    Lu_tf = minreal((Ki*Pi)+(Ko*Po*Pi));

    figure (7)
    w  = logspace(-1,3, 2000);
    [L_u_mag, L_u_phase] = bode(Lu_tf,w);
    semilogx(w, 20*log10(L_u_mag(1,:)))
    ylabel('Magnitude (dB)')
    xlabel('freq (rad/sec)')
    title(['|L_u|: Loop Magnitude at Controls '])
    grid on
    hold on

    figure (8)
    w  = logspace(-1,3, 2000);
    sen_u = feedback(1,Lu_tf)
    [sen_u_mag, sen_u_phase] = bode(sen_u,w);
    semilogx(w, 20*log10(sen_u_mag(1,:)))
    ylabel('Magnitude (dB)')
    xlabel('freq (rad/sec)')
    title(['|S_u|: Loop Sensitivity at Controls '])
    grid on
    hold on

    figure (9)
    w  = logspace(-1,3, 2000);
    comp_sen_u = feedback(Lu_tf,1)
    [comp_sen_u_mag, comp_sen_u_phase] = bode(comp_sen_u,w);
    semilogx(w, 20*log10(comp_sen_u_mag(1,:)))
    ylabel('Magnitude (dB)')
    xlabel('freq (rad/sec)')
    title(['|T_u|: Loop Complementary Sensitivity at Controls '])
    grid on
    hold on

    % Compute the margins of open loop system brken at different points
    Li_margins = allmargin(Li_tf);
    Lu_margins = allmargin(Lu_tf);
    Lo_margins = allmargin(Lo_tf);
    Si_margins = allmargin(sen_i);
    Su_margins = allmargin(sen_u);
    So_margins = allmargin(sen_o);


    Lu_downward_gain_margin(:,i) = Lu_margins.GainMargin(1);
    Lo_upward_gain_margin(:,i) = Lo_margins.GainMargin(2);

    Lu_downward_gain_margin_frequency(:,i) = Lu_margins.GMFrequency(1);
    Lo_upward_gain_margin_frequency(:,i) = Lo_margins.GMFrequency(1);

%     Lu_upward_gain_margin(:,i) = Lu_margins.GainMargin(2);
%     Lo_upward_gain_margin(:,i) = Lo_margins.GainMargin(2);
%
%     Lu_upward_gain_margin_frequency(:,i) = Lu_margins.GMFrequency(2);
%     Lo_upward_gain_margin_frequency(:,i) = Lo_margins.GMFrequency(2);

    Lu_phase_margin(:,i) = Lu_margins.PhaseMargin;
    Lo_phase_margin(:,i) = Lo_margins.PhaseMargin;

    Li_phase_margin_frequency(:,i) = Li_margins.PMFrequency;
    Lu_phase_margin_frequency(:,i) = Lu_margins.PMFrequency;
    Lo_phase_margin_frequency(:,i) = Lo_margins.PMFrequency;


    Lu_delay_margin(:,i) = Lu_margins.DelayMargin;
    Lo_delay_margin(:,i) = Lo_margins.DelayMargin;

    Lu_delay_margin_frequency(:,i) = Lu_margins.DMFrequency;
    Lo_delay_margin_frequency(:,i) = Lo_margins.DMFrequency;

    Si_phase_margin_frequency(:,i) = Si_margins.PMFrequency(1);
    Su_phase_margin_frequency(:,i) = Su_margins.PMFrequency(1);
    So_phase_margin_frequency(:,i) = So_margins.PMFrequency(1);
end

figure (1)
legend(['Parameter Set 1'],['Parameter Set 2'],...
    ['Parameter Set 3'],['Parameter Set 4'],...
    ['Parameter Set 5'],['Parameter Set 6'],...
    'Location','Southwest','NumColumns',1);

figure (2)
legend(['Parameter Set 1'],['Parameter Set 2'],...
```

```
    ['Parameter Set 3'],['Parameter Set 4'],...
    ['Parameter Set 5'],['Parameter Set 6'],...
    'Location','Southwest','NumColumns',1);

figure (3)
legend(['Parameter Set 1'],['Parameter Set 2'],...
    ['Parameter Set 3'],['Parameter Set 4'],...
    ['Parameter Set 5'],['Parameter Set 6'],...
    'Location','Southwest','NumColumns',1);
figure (4)
legend(['Parameter Set 1'],['Parameter Set 2'],...
    ['Parameter Set 3'],['Parameter Set 4'],...
    ['Parameter Set 5'],['Parameter Set 6'],...
    'Location','Southwest','NumColumns',1);
figure (5)
legend(['Parameter Set 1'],['Parameter Set 2'],...
    ['Parameter Set 3'],['Parameter Set 4'],...
    ['Parameter Set 5'],['Parameter Set 6'],...
    'Location','Southwest','NumColumns',1);
figure (6)
legend(['Parameter Set 1'],['Parameter Set 2'],...
    ['Parameter Set 3'],['Parameter Set 4'],...
    ['Parameter Set 5'],['Parameter Set 6'],...
    'Location','Southwest','NumColumns',1);

figure (7)
legend(['Parameter Set 1'],['Parameter Set 2'],...
    ['Parameter Set 3'],['Parameter Set 4'],...
    ['Parameter Set 5'],['Parameter Set 6'],...
    'Location','Southwest','NumColumns',1);

figure (8)
legend(['Parameter Set 1'],['Parameter Set 2'],...
    ['Parameter Set 3'],['Parameter Set 4'],...
    ['Parameter Set 5'],['Parameter Set 6'],...
    'Location','Southwest','NumColumns',1);

figure (9)
legend(['Parameter Set 1'],['Parameter Set 2'],...
    ['Parameter Set 3'],['Parameter Set 4'],...
    ['Parameter Set 5'],['Parameter Set 6'],...
    'Location','Southwest','NumColumns',1);

Li_phase_margin_frequency
Lu_phase_margin_frequency
Lo_phase_margin_frequency

Lu_downward_gain_margin
Lo_upward_gain_margin

Lu_phase_margin
Lo_phase_margin

Si_phase_margin_frequency
Su_phase_margin_frequency
So_phase_margin_frequency

%% Comparing different peaks
close all;
clear all;
clc;

% Varying g_o with g_i as a parameter


% NOMINAL PARAMETERS
p   = 5.693;
z   = 4.958;
zi1_values = [1.97, 1.99, 2, 2.01, 2]
zi2_values = [2.16, 1.8, 2.2, 2.37, 2.2];
zo1_values = [0.08,0.09,0.4,0.07,0.4];
zo2_values = [1,0.52,1.4,1.05,1.4];
gi_values = [17.95, 18.94, 17.85, 16.94, 17]; % gi >17
go_values = [0.749,1.1,0.949,1.742,3];


for i = 1:length(go_values)
    gi = gi_values(:,i);
    zi1 = zi1_values(:,i);
    zi2 = zi2_values(:,i);
    go = go_values(:,i);
    zo1 = zo1_values(:,i);
    zo2 = zo2_values(:,i);


    % transfer function analogy
    Pi = zpk([],[p,-p],[1])
    Ki = zpk([-zi1,-zi2],[0],[gi])
    Li_tf = series(Pi,Ki)


    figure (1)
    w   = logspace(-1,3, 2000);
    [L_i_mag, L_i_phase] = bode(Li_tf,w);
    semilogx(w, 20*log10(L_i_mag(1,:)))
    ylabel('Magnitude (dB)')
    xlabel('freq (rad/sec)')
```

```
title(['Inner Loop Magnitude |L_i| '])
grid on
hold on


figure (2)
w  = logspace(-1,3, 2000);
sen_i = feedback(1,Li_tf)
[sen_i_mag, sen_i_phase] = bode(sen_i,w);
semilogx(w, 20*log10(sen_i_mag(1,:)))
ylabel('Magnitude (dB)')
xlabel('freq (rad/sec)')
title(['Inner Loop Sensitivity |S_i| '])
grid on
hold on


figure (3)
w  = logspace(-1,3, 2000);
comp_sen_i = feedback(Li_tf,1)
[comp_sen_i_mag, comp_sen_i_phase] = bode(comp_sen_i,w);
semilogx(w, 20*log10(comp_sen_i_mag(1,:)))
ylabel('Magnitude (dB)')
xlabel('freq (rad/sec)')
title(['Inner Loop Complementary Sensitivity |T_i| '])
grid on
hold on

% transfer function
Po = zpk([z,-z],[0,0],[-1])
Ko = zpk([-zo1,-zo2],[0],[go])
Lo_tf = minreal(Po*Pi*Ko*(feedback(1,(Pi*Ki))));

% outer loop plots at output
figure (4)
w  = logspace(-1,3, 2000);
[L_o_mag, L_o_phase] = bode(Lo_tf,w);
semilogx(w, 20*log10(L_o_mag(1,:)))
ylabel('Magnitude (dB)')
xlabel('freq (rad/sec)')
title(['Loop Magnitude at Output |L_o| '])
grid on
hold on

figure (5)
w  = logspace(-1,3, 2000);
sen_o = feedback(1,Lo_tf)
[sen_o_mag, sen_o_phase] = bode(sen_o,w);
semilogx(w, 20*log10(sen_o_mag(1,:)))
ylabel('Magnitude (dB)')
xlabel('freq (rad/sec)')
title(['|S_o|: Loop Sensitivity at Output '])
grid on
hold on

figure (6)
w  = logspace(-1,3, 2000);
comp_sen_o = feedback(Lo_tf,1)
[comp_sen_o_mag, comp_sen_o_phase] = bode(comp_sen_o,w);
semilogx(w, 20*log10(comp_sen_o_mag(1,:)))
ylabel('Magnitude (dB)')
xlabel('freq (rad/sec)')
title(['|T_o|: Loop Complementary Sensitivity at Output '])
grid on
hold on


% Loop at Controls
Lu_tf = minreal((Ki*Pi)+(Ko*Po*Pi));

figure (7)
w  = logspace(-1,3, 2000);
[L_u_mag, L_u_phase] = bode(Lu_tf,w);
semilogx(w, 20*log10(L_u_mag(1,:)))
ylabel('Magnitude (dB)')
xlabel('freq (rad/sec)')
title(['|L_u|: Loop Magnitude at Controls '])
grid on
hold on

figure (8)
w  = logspace(-1,3, 2000);
sen_u = feedback(1,Lu_tf)
[sen_u_mag, sen_u_phase] = bode(sen_u,w);
semilogx(w, 20*log10(sen_u_mag(1,:)))
ylabel('Magnitude (dB)')
xlabel('freq (rad/sec)')
title(['|S_u|: Loop Sensitivity at Controls '])
grid on
hold on

figure (9)
w  = logspace(-1,3, 2000);
comp_sen_u = feedback(Lu_tf,1)
[comp_sen_u_mag, comp_sen_u_phase] = bode(comp_sen_u,w);
semilogx(w, 20*log10(comp_sen_u_mag(1,:)))
ylabel('Magnitude (dB)')
```

```
    xlabel('freq (rad/sec)')
    title(['|T_u|: Loop Complementary Sensitivity at Controls '])
    grid on
    hold on

    % Compute the margins of open loop system brken at different points
    Li_margins = allmargin(Li_tf);
    Lu_margins = allmargin(Lu_tf);
    Lo_margins = allmargin(Lo_tf);
    Si_margins = allmargin(sen_i);
    Su_margins = allmargin(sen_u);
    So_margins = allmargin(sen_o);


    Lu_downward_gain_margin(:,i) = Lu_margins.GainMargin(1);
    Lo_upward_gain_margin(:,i) = Lo_margins.GainMargin(2);

    Lu_downward_gain_margin_frequency(:,i) = Lu_margins.GMFrequency(1);
    Lo_upward_gain_margin_frequency(:,i) = Lo_margins.GMFrequency(1);

%     Lu_upward_gain_margin(:,i) = Lu_margins.GainMargin(2);
%     Lo_upward_gain_margin(:,i) = Lo_margins.GainMargin(2);
%
%     Lu_upward_gain_margin_frequency(:,i) = Lu_margins.GMFrequency(2);
%     Lo_upward_gain_margin_frequency(:,i) = Lo_margins.GMFrequency(2);

    Lu_phase_margin(:,i) = Lu_margins.PhaseMargin;
    Lo_phase_margin(:,i) = Lo_margins.PhaseMargin;

    Li_phase_margin_frequency(:,i) = Li_margins.PMFrequency;
    Lu_phase_margin_frequency(:,i) = Lu_margins.PMFrequency;
    Lo_phase_margin_frequency(:,i) = Lo_margins.PMFrequency;


    Lu_delay_margin(:,i) = Lu_margins.DelayMargin;
    Lo_delay_margin(:,i) = Lo_margins.DelayMargin;

    Lu_delay_margin_frequency(:,i) = Lu_margins.DMFrequency;
    Lo_delay_margin_frequency(:,i) = Lo_margins.DMFrequency;

    Si_phase_margin_frequency(:,i) = Si_margins.PMFrequency(1);
    Su_phase_margin_frequency(:,i) = Su_margins.PMFrequency(1);
    So_phase_margin_frequency(:,i) = So_margins.PMFrequency(1);
end

figure (1)
legend(['Parameter Set 1'],['Parameter Set 2'],...
    ['Parameter Set 3'],['Parameter Set 4'],...
    ['Parameter Set 5'],['Parameter Set 6'],...
    'Location','Southwest','NumColumns',1);

figure (2)
legend(['Parameter Set 1'],['Parameter Set 2'],...
    ['Parameter Set 3'],['Parameter Set 4'],...
    ['Parameter Set 5'],['Parameter Set 6'],...
    'Location','Southwest','NumColumns',1);

figure (3)
legend(['Parameter Set 1'],['Parameter Set 2'],...
    ['Parameter Set 3'],['Parameter Set 4'],...
    ['Parameter Set 5'],['Parameter Set 6'],...
    'Location','Southwest','NumColumns',1);
figure (4)
legend(['Parameter Set 1'],['Parameter Set 2'],...
    ['Parameter Set 3'],['Parameter Set 4'],...
    ['Parameter Set 5'],['Parameter Set 6'],...
    'Location','Southwest','NumColumns',1);
figure (5)
legend(['Parameter Set 1'],['Parameter Set 2'],...
    ['Parameter Set 3'],['Parameter Set 4'],...
    ['Parameter Set 5'],['Parameter Set 6'],...
    'Location','Southwest','NumColumns',1);
figure (6)
legend(['Parameter Set 1'],['Parameter Set 2'],...
    ['Parameter Set 3'],['Parameter Set 4'],...
    ['Parameter Set 5'],['Parameter Set 6'],...
    'Location','Southwest','NumColumns',1);

figure (7)
legend(['Parameter Set 1'],['Parameter Set 2'],...
    ['Parameter Set 3'],['Parameter Set 4'],...
    ['Parameter Set 5'],['Parameter Set 6'],...
    'Location','Southwest','NumColumns',1);

figure (8)
legend(['Parameter Set 1'],['Parameter Set 2'],...
    ['Parameter Set 3'],['Parameter Set 4'],...
    ['Parameter Set 5'],['Parameter Set 6'],...
    'Location','Southwest','NumColumns',1);

figure (9)
legend(['Parameter Set 1'],['Parameter Set 2'],...
    ['Parameter Set 3'],['Parameter Set 4'],...
    ['Parameter Set 5'],['Parameter Set 6'],...
    'Location','Southwest','NumColumns',1);

Li_phase_margin_frequency
```

```
Lu_phase_margin_frequency
Lo_phase_margin_frequency

Lu_downward_gain_margin
Lo_upward_gain_margin

Lu_phase_margin
Lo_phase_margin

Si_phase_margin_frequency
Su_phase_margin_frequency
So_phase_margin_frequency

%% Peak - 1.65 Su/So

% Varying g_o with g_i as a parameter


% NOMINAL PARAMETERS
p  = max(pole(sys_tf));
z  = max(tzero(sys_tf));
%%
zi1_values = [4.6, 1.97, 6.5, 0.9, 11, 11]
zi2_values = [3.9, 2.16, 7, 1.487, 16.1, 21.1];
zo1_values = [1.9, 0.08, 4.05, 0.02, 9.92, 9.92];
zo2_values = [0.351, 1, 0.05, 0.568, 0.03, 0.03];
gi_values = [15.65, 17.95, 21.36, 27.9 , 40, 47]; % gi >17
go_values = [1.49, 0.749, 2.39, 0.475, 5.11, 6.29];


for i = 1:length(go_values)
    gi = gi_values(:,i);
    zi1 = zi1_values(:,i);
    zi2 = zi2_values(:,i);
    go = go_values(:,i);
    zo1 = zo1_values(:,i);
    zo2 = zo2_values(:,i);


    % transfer function analogy
    Pi = zpk([],[p,-p],[1])
    Ki = zpk([-zi1,-zi2],[0],[gi])
    Li_tf = series(Pi,Ki)


    figure (1)
    w  = logspace(-1,3, 2000);
    [L_i_mag, L_i_phase] = bode(Li_tf,w);
    semilogx(w, 20*log10(L_i_mag(1,:)))
    ylabel('Magnitude (dB)')
    xlabel('freq (rad/sec)')
    title(['Inner Loop Magnitude |L_i| for peak = 1.65'])
    grid on
    hold on


    figure (2)
    w  = logspace(-1,3, 2000);
    sen_i = feedback(1,Li_tf)
    [sen_i_mag, sen_i_phase] = bode(sen_i,w);
    semilogx(w, 20*log10(sen_i_mag(1,:)))
    ylabel('Magnitude (dB)')
    xlabel('freq (rad/sec)')
    title(['Inner Loop Sensitivity |S_i| for peak = 1.65'])
    grid on
    hold on


    figure (3)
    w  = logspace(-1,3, 2000);
    comp_sen_i = feedback(Li_tf,1)
    [comp_sen_i_mag, comp_sen_i_phase] = bode(comp_sen_i,w);
    semilogx(w, 20*log10(comp_sen_i_mag(1,:)))
    ylabel('Magnitude (dB)')
    xlabel('freq (rad/sec)')
    title(['Inner Loop Complementary Sensitivity |T_i| for peak = 1.65'])
    grid on
    hold on

    % transfer function
    Po = zpk([z,-z],[0,0],[-1])
    Ko = zpk([-zo1,-zo2],[0],[go])
    Lo_tf = minreal(Po*Pi*Ko*(feedback(1,(Pi*Ki))));

    % outer loop plots at output
    figure (4)
    w  = logspace(-1,3, 2000);
    [L_o_mag, L_o_phase] = bode(Lo_tf,w);
    semilogx(w, 20*log10(L_o_mag(1,:)))
    ylabel('Magnitude (dB)')
    xlabel('freq (rad/sec)')
    title(['Loop Magnitude at Output |L_o| for peak = 1.65'])
    grid on
    hold on

    figure (5)
    w  = logspace(-1,3, 2000);
```

```
sen_o = feedback(1,Lo_tf)
[sen_o_mag, sen_o_phase] = bode(sen_o,w);
semilogx(w, 20*log10(sen_o_mag(1,:)))
ylabel('Magnitude (dB)')
xlabel('freq (rad/sec)')
title(['|S_o|: Loop Sensitivity at Output for peak = 1.65'])
grid on
hold on

figure (6)
w  = logspace(-1,3, 2000);
comp_sen_o = feedback(Lo_tf,1)
[comp_sen_o_mag, comp_sen_o_phase] = bode(comp_sen_o,w);
semilogx(w, 20*log10(comp_sen_o_mag(1,:)))
ylabel('Magnitude (dB)')
xlabel('freq (rad/sec)')
title(['|T_o|: Loop Complementary Sensitivity at Output for peak = 1.65'])
grid on
hold on


% Loop at Controls
Lu_tf = minreal((Ki*Pi)+(Ko*Po*Pi));

figure (7)
w  = logspace(-1,3, 2000);
[L_u_mag, L_u_phase] = bode(Lu_tf,w);
semilogx(w, 20*log10(L_u_mag(1,:)))
ylabel('Magnitude (dB)')
xlabel('freq (rad/sec)')
title(['|L_u|: Loop Magnitude at Controls for peak = 1.65'])
grid on
hold on

figure (8)
w  = logspace(-1,3, 2000);
sen_u = feedback(1,Lu_tf)
[sen_u_mag, sen_u_phase] = bode(sen_u,w);
semilogx(w, 20*log10(sen_u_mag(1,:)))
ylabel('Magnitude (dB)')
xlabel('freq (rad/sec)')
title(['|S_u|: Loop Sensitivity at Controls for peak = 1.65'])
grid on
hold on

figure (9)
w  = logspace(-1,3, 2000);
comp_sen_u = feedback(Lu_tf,1)
[comp_sen_u_mag, comp_sen_u_phase] = bode(comp_sen_u,w);
semilogx(w, 20*log10(comp_sen_u_mag(1,:)))
ylabel('Magnitude (dB)')
xlabel('freq (rad/sec)')
title(['|T_u|: Loop Complementary Sensitivity at Controls for peak = 1.65'])
grid on
hold on

% Compute the margins of open loop system brken at different points
Li_margins = allmargin(Li_tf);
Lu_margins = allmargin(Lu_tf);
Lo_margins = allmargin(Lo_tf);
Si_margins = allmargin(sen_i);
Su_margins = allmargin(sen_u);
So_margins = allmargin(sen_o);


Lu_downward_gain_margin(:,i) = Lu_margins.GainMargin(1);
Lo_upward_gain_margin(:,i) = Lo_margins.GainMargin(2);

Lu_downward_gain_margin_frequency(:,i) = Lu_margins.GMFrequency(1);
Lo_upward_gain_margin_frequency(:,i) = Lo_margins.GMFrequency(1);

%    Lu_upward_gain_margin(:,i) = Lu_margins.GainMargin(2);
%    Lo_upward_gain_margin(:,i) = Lo_margins.GainMargin(2);
%
%    Lu_upward_gain_margin_frequency(:,i) = Lu_margins.GMFrequency(2);
%    Lo_upward_gain_margin_frequency(:,i) = Lo_margins.GMFrequency(2);

Lu_phase_margin(:,i) = Lu_margins.PhaseMargin;
Lo_phase_margin(:,i) = Lo_margins.PhaseMargin;

Li_phase_margin_frequency(:,i) = Li_margins.PMFrequency;
Lu_phase_margin_frequency(:,i) = Lu_margins.PMFrequency;
Lo_phase_margin_frequency(:,i) = Lo_margins.PMFrequency;


Lu_delay_margin(:,i) = Lu_margins.DelayMargin;
Lo_delay_margin(:,i) = Lo_margins.DelayMargin;

Lu_delay_margin_frequency(:,i) = Lu_margins.DMFrequency;
Lo_delay_margin_frequency(:,i) = Lo_margins.DMFrequency;

Si_phase_margin_frequency(:,i) = Si_margins.PMFrequency(1);
Su_phase_margin_frequency(:,i) = Su_margins.PMFrequency(1);
So_phase_margin_frequency(:,i) = So_margins.PMFrequency(1);
end

figure (1)
```

```
legend(['Parameter Set 1'],['Parameter Set 2'],...
    ['Parameter Set 3'],['Parameter Set 4'],...
    ['Parameter Set 5'],['Parameter Set 6'],...
    'Location','Southwest','NumColumns',1);

figure (2)
legend(['Parameter Set 1'],['Parameter Set 2'],...
    ['Parameter Set 3'],['Parameter Set 4'],...
    ['Parameter Set 5'],['Parameter Set 6'],...
    'Location','Southwest','NumColumns',1);

figure (3)
legend(['Parameter Set 1'],['Parameter Set 2'],...
    ['Parameter Set 3'],['Parameter Set 4'],...
    ['Parameter Set 5'],['Parameter Set 6'],...
    'Location','Southwest','NumColumns',1);
figure (4)
legend(['Parameter Set 1'],['Parameter Set 2'],...
    ['Parameter Set 3'],['Parameter Set 4'],...
    ['Parameter Set 5'],['Parameter Set 6'],...
    'Location','Southwest','NumColumns',1);
figure (5)
legend(['Parameter Set 1'],['Parameter Set 2'],...
    ['Parameter Set 3'],['Parameter Set 4'],...
    ['Parameter Set 5'],['Parameter Set 6'],...
    'Location','Southwest','NumColumns',1);
figure (6)
legend(['Parameter Set 1'],['Parameter Set 2'],...
    ['Parameter Set 3'],['Parameter Set 4'],...
    ['Parameter Set 5'],['Parameter Set 6'],...
    'Location','Southwest','NumColumns',1);

figure (7)
legend(['Parameter Set 1'],['Parameter Set 2'],...
    ['Parameter Set 3'],['Parameter Set 4'],...
    ['Parameter Set 5'],['Parameter Set 6'],...
    'Location','Southwest','NumColumns',1);

figure (8)
legend(['Parameter Set 1'],['Parameter Set 2'],...
    ['Parameter Set 3'],['Parameter Set 4'],...
    ['Parameter Set 5'],['Parameter Set 6'],...
    'Location','Southwest','NumColumns',1);

figure (9)
legend(['Parameter Set 1'],['Parameter Set 2'],...
    ['Parameter Set 3'],['Parameter Set 4'],...
    ['Parameter Set 5'],['Parameter Set 6'],...
    'Location','Southwest','NumColumns',1);

Li_phase_margin_frequency
Lu_phase_margin_frequency
Lo_phase_margin_frequency

Lu_downward_gain_margin
Lo_upward_gain_margin

Lu_phase_margin
Lo_phase_margin

Si_phase_margin_frequency
Su_phase_margin_frequency
So_phase_margin_frequency

%% Peak - 2.8 Su/So
close all;
clear all;
clc;

% Varying g_o with g_i as a parameter


% NOMINAL PARAMETERS
p  = 5.693;
z  = 4.958;
zi1_values = [3.06, 2, 10.165, 1, 11, 11]
zi2_values = [3.12, 2.2, 5.4, 1.1, 16.1, 21.1];
zo1_values = [2, 0.4, 6.3, 0.02, 2.92, 1.02];
zo2_values = [0.4, 1.4, 0.1, 0.5346, 0.03, 0.03];
gi_values = [14.79, 17.85, 22.342, 30.858, 40, 47]; % gi >17
go_values = [1.41, 0.949, 4.095, 0.69, 8.71, 10.59];


for i = 1:length(go_values)
    gi = gi_values(:,i);
    zi1 = zi1_values(:,i);
    zi2 = zi2_values(:,i);
    go = go_values(:,i);
    zo1 = zo1_values(:,i);
    zo2 = zo2_values(:,i);


    % transfer function analogy
    Pi = zpk([],[p,-p],[1])
    Ki = zpk([-zi1,-zi2],[0],[gi])
    Li_tf = series(Pi,Ki)
```

```
figure (1)
w   = logspace(-1,3, 2000);
[L_i_mag, L_i_phase] = bode(Li_tf,w);
semilogx(w, 20*log10(L_i_mag(1,:)))
ylabel('Magnitude (dB)')
xlabel('freq (rad/sec)')
title(['Inner Loop Magnitude |L_i| for peak = 2.8'])
grid on
hold on


figure (2)
w   = logspace(-1,3, 2000);
sen_i = feedback(1,Li_tf)
[sen_i_mag, sen_i_phase] = bode(sen_i,w);
semilogx(w, 20*log10(sen_i_mag(1,:)))
ylabel('Magnitude (dB)')
xlabel('freq (rad/sec)')
title(['Inner Loop Sensitivity |S_i| for peak = 2.8'])
grid on
hold on


figure (3)
w   = logspace(-1,3, 2000);
comp_sen_i = feedback(Li_tf,1)
[comp_sen_i_mag, comp_sen_i_phase] = bode(comp_sen_i,w);
semilogx(w, 20*log10(comp_sen_i_mag(1,:)))
ylabel('Magnitude (dB)')
xlabel('freq (rad/sec)')
title(['Inner Loop Complementary Sensitivity |T_i| for peak = 2.8'])
grid on
hold on

% transfer function
Po = zpk([z,-z],[0,0],[-1])
Ko = zpk([-zo1,-zo2],[0],[go])
Lo_tf = minreal(Po*Pi*Ko*(feedback(1,(Pi*Ki))));

% outer loop plots at output
figure (4)
w   = logspace(-1,3, 2000);
[L_o_mag, L_o_phase] = bode(Lo_tf,w);
semilogx(w, 20*log10(L_o_mag(1,:)))
ylabel('Magnitude (dB)')
xlabel('freq (rad/sec)')
title(['Loop Magnitude at Output |L_o| for peak = 2.8'])
grid on
hold on

figure (5)
w   = logspace(-1,3, 2000);
sen_o = feedback(1,Lo_tf)
[sen_o_mag, sen_o_phase] = bode(sen_o,w);
semilogx(w, 20*log10(sen_o_mag(1,:)))
ylabel('Magnitude (dB)')
xlabel('freq (rad/sec)')
title(['|S_o|: Loop Sensitivity at Output for peak = 2.8'])
grid on
hold on

figure (6)
w   = logspace(-1,3, 2000);
comp_sen_o = feedback(Lo_tf,1)
[comp_sen_o_mag, comp_sen_o_phase] = bode(comp_sen_o,w);
semilogx(w, 20*log10(comp_sen_o_mag(1,:)))
ylabel('Magnitude (dB)')
xlabel('freq (rad/sec)')
title(['|T_o|: Loop Complementary Sensitivity at Output for peak = 2.8'])
grid on
hold on


% Loop at Controls
Lu_tf = minreal((Ki*Pi)+(Ko*Po*Pi));

figure (7)
w   = logspace(-1,3, 2000);
[L_u_mag, L_u_phase] = bode(Lu_tf,w);
semilogx(w, 20*log10(L_u_mag(1,:)))
ylabel('Magnitude (dB)')
xlabel('freq (rad/sec)')
title(['|L_u|: Loop Magnitude at Controls for peak = 2.8'])
grid on
hold on

figure (8)
w   = logspace(-1,3, 2000);
sen_u = feedback(1,Lu_tf)
[sen_u_mag, sen_u_phase] = bode(sen_u,w);
semilogx(w, 20*log10(sen_u_mag(1,:)))
ylabel('Magnitude (dB)')
xlabel('freq (rad/sec)')
title(['|S_u|: Loop Sensitivity at Controls for peak = 2.8'])
grid on
hold on
```

```
        figure (9)
        w  = logspace(-1,3, 2000);
        comp_sen_u = feedback(Lu_tf,1)
        [comp_sen_u_mag, comp_sen_u_phase] = bode(comp_sen_u,w);
        semilogx(w, 20*log10(comp_sen_u_mag(1,:)))
        ylabel('Magnitude (dB)')
        xlabel('freq (rad/sec)')
        title(['|T_u|: Loop Complementary Sensitivity at Controls for peak = 2.8'])
        grid on
        hold on

        % Compute the margins of open loop system brken at different points
        Li_margins = allmargin(Li_tf);
        Lu_margins = allmargin(Lu_tf);
        Lo_margins = allmargin(Lo_tf);
        Si_margins = allmargin(sen_i);
        Su_margins = allmargin(sen_u);
        So_margins = allmargin(sen_o);


        Lu_downward_gain_margin(:,i) = Lu_margins.GainMargin(1);
        Lo_upward_gain_margin(:,i) = Lo_margins.GainMargin(2);

        Lu_downward_gain_margin_frequency(:,i) = Lu_margins.GMFrequency(1);
        Lo_upward_gain_margin_frequency(:,i) = Lo_margins.GMFrequency(1);

%       Lu_upward_gain_margin(:,i) = Lu_margins.GainMargin(2);
%       Lo_upward_gain_margin(:,i) = Lo_margins.GainMargin(2);
%
%       Lu_upward_gain_margin_frequency(:,i) = Lu_margins.GMFrequency(2);
%       Lo_upward_gain_margin_frequency(:,i) = Lo_margins.GMFrequency(2);

        Lu_phase_margin(:,i) = Lu_margins.PhaseMargin;
        Lo_phase_margin(:,i) = Lo_margins.PhaseMargin;

        Li_phase_margin_frequency(:,i) = Li_margins.PMFrequency;
        Lu_phase_margin_frequency(:,i) = Lu_margins.PMFrequency;
        Lo_phase_margin_frequency(:,i) = Lo_margins.PMFrequency;


        Lu_delay_margin(:,i) = Lu_margins.DelayMargin;
        Lo_delay_margin(:,i) = Lo_margins.DelayMargin;

        Lu_delay_margin_frequency(:,i) = Lu_margins.DMFrequency;
        Lo_delay_margin_frequency(:,i) = Lo_margins.DMFrequency;

        Si_phase_margin_frequency(:,i) = Si_margins.PMFrequency(1);
        Su_phase_margin_frequency(:,i) = Su_margins.PMFrequency(1);
        So_phase_margin_frequency(:,i) = So_margins.PMFrequency(1);
end

figure (1)
legend(['Parameter Set 1'],['Parameter Set 2'],...
    ['Parameter Set 3'],['Parameter Set 4'],...
    ['Parameter Set 5'],['Parameter Set 6'],...
    'Location','Southwest','NumColumns',1);

figure (2)
legend(['Parameter Set 1'],['Parameter Set 2'],...
    ['Parameter Set 3'],['Parameter Set 4'],...
    ['Parameter Set 5'],['Parameter Set 6'],...
    'Location','Southwest','NumColumns',1);

figure (3)
legend(['Parameter Set 1'],['Parameter Set 2'],...
    ['Parameter Set 3'],['Parameter Set 4'],...
    ['Parameter Set 5'],['Parameter Set 6'],...
    'Location','Southwest','NumColumns',1);
figure (4)
legend(['Parameter Set 1'],['Parameter Set 2'],...
    ['Parameter Set 3'],['Parameter Set 4'],...
    ['Parameter Set 5'],['Parameter Set 6'],...
    'Location','Southwest','NumColumns',1);
figure (5)
legend(['Parameter Set 1'],['Parameter Set 2'],...
    ['Parameter Set 3'],['Parameter Set 4'],...
    ['Parameter Set 5'],['Parameter Set 6'],...
    'Location','Southwest','NumColumns',1);
figure (6)
legend(['Parameter Set 1'],['Parameter Set 2'],...
    ['Parameter Set 3'],['Parameter Set 4'],...
    ['Parameter Set 5'],['Parameter Set 6'],...
    'Location','Southwest','NumColumns',1);

figure (7)
legend(['Parameter Set 1'],['Parameter Set 2'],...
    ['Parameter Set 3'],['Parameter Set 4'],...
    ['Parameter Set 5'],['Parameter Set 6'],...
    'Location','Southwest','NumColumns',1);

figure (8)
legend(['Parameter Set 1'],['Parameter Set 2'],...
    ['Parameter Set 3'],['Parameter Set 4'],...
    ['Parameter Set 5'],['Parameter Set 6'],...
    'Location','Southwest','NumColumns',1);
```

```
figure (9)
legend([['Parameter Set 1'],['Parameter Set 2'],...
    ['Parameter Set 3'],['Parameter Set 4'],...
    ['Parameter Set 5'],['Parameter Set 6'],...
    'Location','Southwest','NumColumns',1);

Li_phase_margin_frequency
Lu_phase_margin_frequency
Lo_phase_margin_frequency

Lu_downward_gain_margin
Lo_upward_gain_margin

Lu_phase_margin
Lo_phase_margin

Si_phase_margin_frequency
Su_phase_margin_frequency
So_phase_margin_frequency
%% December 16, 2020
%
% Inner-Outer Loop Feedback Structure
% by Soham Sarkar, Armando Rodriguez, Brent Wallace
%% Benchmark Example 8 cart inverted
%% Cart Inverted Pendulum Analysis
clc
clear all
close all


%% Derivation of Equations (Matthew Peter Kelly)
syms pos(t) pos_dot(t) pos_dbldot(t) theta(t) theta_dot(t) theta_dbldot(t);
syms m M l b g I u;

Xp = pos(t) + l*sin(theta(t));
Yp = l*cos(theta(t));
Xp_dot = diff(Xp);
Yp_dot = diff(Yp);


Xp_dot = subs(Xp_dot,[diff(pos(t),t), diff(theta(t),t)],[pos_dot(t) theta_dot(t)]);
Yp_dot = subs(Yp_dot,[diff(pos(t),t), diff(theta(t),t)],[pos_dot(t) theta_dot(t)]);

pos_dot(t) = subs(pos_dot(t),diff(pos(t),t),pos_dot(t));
T =  ((1/2)*M*pos_dot(t)) + ((1/2)*m*((Xp_dot^2) + (Yp_dot^2)));

T = ((1/2)*M*pos_dot(t)^2)+((1/2)*(m)*(pos_dot(t)^2))...
    +((1/2)*m*(2*l*pos_dot(t)*theta_dot(t)*cos(theta(t))))...
    +((1/2)*m*(l^2)*(theta_dot^2))+((1/2)*I*(theta_dot^2))
U = m*g*l*cos(theta(t));
L = T - U;

% First Equation with respect to theta1(t)
dL_dpos_dot(t) = functionalDerivative(L,pos_dot(t));

dL_dpos(t) = functionalDerivative(L,pos(t));

% Second Equation with respect to theta
dL_dtheta_dot(t) = functionalDerivative(L,theta_dot(t));

dL_dtheta(t) = functionalDerivative(L,theta(t));


dL_dpos_dot_dt = jacobian(dL_dpos_dot(t),t);
dL_dpos_dot_dt = subs(dL_dpos_dot_dt,[diff(pos(t),t),...
    diff(pos_dot(t),t), diff(theta(t),t), diff(theta_dot(t),t)]...
    ,[pos_dot(t),pos_dbldot(t),theta_dot(t),theta_dbldot(t)]);


eqn1 = dL_dpos_dot_dt - dL_dpos(t) + b*theta_dot(t) - u;

% theta_1dbldot(t) = solve(eqn1,theta_1dbldot(t),'Real',true);

fprintf('Equation 1: \n');
disp(eqn1)
% fprintf('Equation 1 in terms of theta_1dbldot(t): \n');
% disp(theta_1dbldot(t))

% Remember this trick that you did, apply for all
dL_dtheta_dot_dt = jacobian(dL_dtheta_dot(t),t);
dL_dtheta_dot_dt = subs(dL_dtheta_dot_dt,[diff(pos(t),t),...
    diff(pos_dot(t),t), diff(theta(t),t), diff(theta_dot(t),t)]...
    ,[pos_dot(t),pos_dbldot(t),theta_dot(t),theta_dbldot(t)]);
eqn2 = dL_dtheta_dot_dt - dL_dtheta(t);


fprintf('Equation 2: \n');
disp(eqn2)
%% Finding the Langrange Equations with variables
syms x th x_dot th_dot x_dbldot th_dbldot;

eqn1 = subs(eqn1,[pos(t),theta(t),pos_dot(t),theta_dot(t),...
    pos_dbldot(t),theta_dbldot(t)], [x,th,x_dot,th_dot,...
    x_dbldot,th_dbldot]);
fprintf('Equation of Motion 1: \n');
disp(eqn1)
```

```matlab
eqn2 = subs(eqn2,[pos(t),theta(t),pos_dot(t),theta_dot(t),...
    pos_dbldot(t),theta_dbldot(t)], [x,th,x_dot,th_dot,...
    x_dbldot,th_dbldot]);
fprintf('Equation of Motion 2: \n');
disp(eqn2)

%% Linearization
syms u;
[x_dbldot,th_dbldot]= solve([eqn1,eqn2],[x_dbldot,th_dbldot])

q = [x x_dot th th_dot];
u = [u];

% Non Linear Function
f1 = x_dot;
f2 = x_dbldot;
f3 = th_dot;
f4 = th_dbldot;

F = [f1 f2 f3 f4];

% Jacobian
J = jacobian(F,q)

% Equilibrium Points
q_e = [0 0 0 0];
J = subs(J,q,q_e);
disp(J);

% System Matrix
A = J;
% Input Matrix
B = jacobian(F,u);
B = subs(B,q,q_e);


%% Plant Nominal Values (hernandez)
m = 0.267; % in Kilograms
L = 2.016; % in metres
l = L/2; % in metres
b = 0.0015; % in Kg m^2/s
k = 0; % in Kg/s^2
g = 9.8; % in m/s^2
I = 0; % in Nm/rad^2/sec^2
M = 0.8; %in Kilograms

%% Plant Nominal Values (ogata)
m = 0.21; % in Kilograms
L = 0.61; % in metres
l = L/2; % in metres
b = 0; % in Kg m^2/s
k = 0; % in Kg/s^2
g = 9.8; % in m/s^2
I = 0.006; % in Nm/rad^2/sec^2
M = 0.455; %in Kilograms

%% Plant Nominal Values (vignesh)
m = 0.20; % in Kilograms
% L = 0.115; % in metres
l = 0.115; % in metres
b = 0.0005; % in Kg m^2/s
k = 0; % in Kg/s^2
g = 9.81; % in m/s^2
I = 0.0002545; % in Nm/rad^2/sec^2
M = 1.21; %in Kilograms

%% State Space
Ap = double(subs(A)) %System Matrix
Bp = double(subs(B)) % Input Matrix
% Cp = [1 0; 0 1];
% Cp = [1 0 0 0;
%       0 0 1 0] % Output Matrix
Cp = [1 0 0 0] % Output Matrix
% Dp = [0; 0];
Dp = zeros(size(Cp,1),size(Bp,2)) % Feedforward Matrix
%% Scaling the Matrices
% FACTS ON SCALING: Scaling affects the shape of singular value plots
%                   (i.e. multivariable frequency responses).
%                   It does not alter pole locations, zero locations.
%                   It does alter directionality information.
%
% Scaling Matrices
%
%   unew = su uold    (foot pounds, foot pounds)
%   xnew = sx xold    (degrees, degrees, degrees/sec, degrees/sec)
%   ynew = sy yold    (degrees, degrees)
%-
%
% 0.2247 lb/N         <--- Convert N to lb
% 4.45  N/lb
% 3.281 ft/m          <--- Convert m to ft
% 0.3048 m/ft
% 180 degrees/pi rad   <--- Convert rad to deg
%
r2d = 180/pi;                                       % Convert radians to degrees
su  = diag([1] );   % Convert N-m to lb-ft
```

```matlab
sx  = diag( [ 1, 1, r2d, r2d ]);                    % Convert radians to degrees
% sy = diag( [ 1 1,r2d,r2d ] );
sy  = diag( [1] );                  % Convert radians to degrees

%
% Scaled System Dynamics
%
%
% u = [ torque_1    (lb-ft)   torque_2     (lb-ft) ]
% x = [ theta_1     (deg)      theta_2      (deg)     dot_theta_1 (deg/sec)    dot_theta_2 (deg/sec) ]
% y = [ theta_1     (deg)      theta_2      (deg) ]
%
Ap = sx*Ap*inv(sx)
Bp = sx*Bp*inv(su)
Cp = sy*Cp*inv(sx)
Dp = sy*Dp*inv(su)
%% State Space Representation
states = {'x','\theta','x_dot','\theta_dot'}; % States
inputs = {'force'}; % Inputs
% outputs = {'x','theta_dot'}
outputs = {'x'}; % Outputs
sys_ss = ss(Ap,Bp,Cp,Dp,'statename',states,'inputname',inputs,'outputname',outputs) % State Space
sys_tf = zpk(sys_ss) % Transfer Function


%% pid search

% NOMINAL PARAMETERS
p  = max(pole(sys_tf));
z  = max(tzero(sys_tf));

numpoints = 2;
% Range of gi
gi_min = 10
gi_max = 15
gi_count = numpoints
gi_inc = (gi_max-gi_min)/gi_count

% Range of go
go_min = 0.8
go_max = 2.0
go_count = numpoints
go_inc = (go_max-go_min)/go_count

% Range of zi1
zi1_min = 0.1;
zi1_max = 0.3;
zi1_count = numpoints
zi1_inc = (zi1_max-zi1_min)/zi1_count

% Range of zi2
zi2_min = 2;
zi2_max = 3;
zi2_count = numpoints
zi2_inc = (zi2_max-zi2_min)/zi2_count

% Range of zo1
zo1_min = 0.01;
zo1_max = 0.05;
zo1_count = numpoints
zo1_inc = (zo1_max-zo1_min)/zo1_count

% Range of zo2
zo2_min = 0.01;
zo2_max = 0.05;
zo2_count = numpoints
zo2_inc = (zo2_max-zo2_min)/zo2_count


s = tf('s');

counter = 0;
for go = go_min:go_inc:go_max
    counter = counter + 1;
    go_values(:,counter) = go;
    counter1 = 0;
    for gi = gi_min:gi_inc:gi_max
        counter1 = counter1 + 1;
        gi_values(:,counter1) = gi;
        counter2 = 0;
        for zi1 = zi1_min:zi1_inc:zi1_max
            counter2 = counter2 + 1;
            zi1_values(:,counter2) = zi1;
            counter3 = 0;
            for zi2 = zi2_min:zi2_inc:zi2_max
                counter3 = counter3 + 1;
                zi2_values(:,counter3) = zi2;
                counter4 = 0;
                for zo1 = zo1_min:zo1_inc:zo1_max
                    counter4 = counter4 + 1;
                    zo1_values(:,counter4) = zo1;
                    counter5 = 0;
                    for zo2 = zo2_min:zo2_inc:zo2_max
                        counter5 = counter5 + 1;
                        zo2_values(:,counter5) = zo2;

                        % replace the following example by example_(number) to view example
                        % results
```

```
[Pi,Ki,Po,Ko]  = example_7(p,z,gi,go,zi1,zi2,zo1,zo2)
% transfer function analogy
Li_tf = series(Pi,Ki)



figure (1)
w  = logspace(-1,3, 2000);
[L_i_mag, L_i_phase] = bode(Li_tf,w);
semilogx(w, 20*log10(L_i_mag(1,:)))
ylabel('Magnitude (dB)')
xlabel('freq (rad/sec)')
title(['Inner Loop Magnitude |L_i| for z = ',num2str(z)...
    ,',p = ',num2str(p)])
grid on
%    hold on

figure (2)
w  = logspace(-1,3, 2000);
[L_i_mag, L_i_phase] = bode(Li_tf,w);
semilogx(w, L_i_phase(1,:))
ylabel('Angle (deg)')
xlabel('freq (rad/sec)')
title(['Inner Loop Phase \angle L_i for z = ',num2str(z)...
    ,',p = ',num2str(p)])
grid on


figure (3)
w  = logspace(-1,3, 2000);
sen_i = feedback(1,Li_tf)
[sen_i_mag, sen_i_phase] = bode(sen_i,w);
[hinf_sen_i,fpeak_sen_i] = hinfnorm(sen_i)
semilogx(w, 20*log10(sen_i_mag(1,:)))
ylabel('Magnitude (dB)')
xlabel('freq (rad/sec)')
title(['Inner Loop Sensitivity |S_i| for z = ',num2str(z)...
    ,',p = ',num2str(p)])
grid on
%    hold on

figure (4)
w  = logspace(-1,3, 2000);
comp_sen_i = feedback(Li_tf,1)
[comp_sen_i_mag, comp_sen_i_phase] = bode(comp_sen_i,w);
[hinf_comp_sen_i,fpeak_comp_sen_i] = hinfnorm(comp_sen_i)
semilogx(w, 20*log10(comp_sen_i_mag(1,:)))
ylabel('Magnitude (dB)')
xlabel('freq (rad/sec)')
title(['Inner Loop Complementary Sensitivity |T_i| for z = ',num2str(z)...
    ,',p = ',num2str(p)])
grid on
%    hold on

figure (5)
semilogx(w, 20*log10(comp_sen_i_mag(1,:)), 'r')
hold on
semilogx(w, 20*log10(sen_i_mag(1,:)), 'b')
ylabel('Magnitude (dB)')
xlabel('freq (rad/sec)')
title(['Inner Loop Sensitivity |S_i| & Complementary Sensitivity |T_i| for z = ',num2str(z),',p = ',num2str(p)])
grid on
legend({'Complementary Sensitivity','Sensitivity'},...
    'Location','Southwest','NumColumns',1)

% transfer function
Lo_tf = minreal(Po*Pi*Ko*(feedback(1,(Pi*Ki))));


% outer loop plots at output
figure (6)
w  = logspace(-1,3, 2000);
[L_o_mag, L_o_phase] = bode(Lo_tf,w);
semilogx(w, 20*log10(L_o_mag(1,:)))
ylabel('Magnitude (dB)')
xlabel('freq (rad/sec)')
title(['Loop Magnitude at Output |L_o| (g_i = ',num2str(gi)...
    ,'), for z = ',num2str(z),',p = ',num2str(p)])
grid on
hold on

figure (7)
w  = logspace(-1,3, 2000);
[L_o_mag, L_o_phase] = bode(Lo_tf,w);
semilogx(w, L_o_phase(1,:)-360)
ylabel('Angle (deg)')
xlabel('freq (rad/sec)')
title(['\angle L_o : Loop Phase at Output (g_i = ',num2str(gi)...
    ,'), for z = ',num2str(z),',p = ',num2str(p)])
grid on
hold on

figure (8)
w  = logspace(-1,3, 2000);
sen_o = feedback(1,Lo_tf)
[sen_o_mag, sen_o_phase] = bode(sen_o,w);
[hinf_sen_o,fpeak_sen_o] = hinfnorm(sen_o)
```

```
hinf_so(:,counter) = hinf_sen_o
semilogx(w, 20*log10(sen_o_mag(1,:)))
ylabel('Magnitude (dB)')
xlabel('freq (rad/sec)')
title(['|S_o|: Loop Sensitivity at Output (g_i = ',...
    ,num2str(gi),'), for z = ',num2str(z),',p = ',num2str(p)])
grid on
hold on


figure (9)
w  = logspace(-1,3, 2000);
comp_sen_o = feedback(Lo_tf,1)
[comp_sen_o_mag, comp_sen_o_phase] = bode(comp_sen_o,w);
[hinf_comp_sen_o,fpeak_sen_o] = hinfnorm(comp_sen_o)
hinf_cso(:,counter) = hinf_comp_sen_o
semilogx(w, 20*log10(comp_sen_o_mag(1,:)))
ylabel('Magnitude (dB)')
xlabel('freq (rad/sec)')
title(['|T_o|: Loop Complementary Sensitivity at Output (g_i = ',...
    num2str(gi),'), for z = ',num2str(z),',p = ',num2str(p)])
grid on
hold on


% Outer Loop at Controls

Lu_tf = minreal((Ki*Pi)+(Ko*Po*Pi));

% Outer Loop poles at controls
figure (10)
w  = logspace(-1,3, 2000);
[L_u_mag, L_u_phase] = bode(Lu_tf,w);
semilogx(w, 20*log10(L_u_mag(1,:)))
ylabel('Magnitude (dB)')
xlabel('freq (rad/sec)')
title(['|L_u|: Loop Magnitude at Controls (g_i = ',...
    num2str(gi),'), for z = ',num2str(z),',p = ',num2str(p)])
grid on
hold on


figure (11)
w  = logspace(-1,3, 2000);
[L_u_mag, L_u_phase] = bode(Lu_tf,w);
semilogx(w, L_u_phase(1,:))
ylabel('Angle (deg)')
xlabel('freq (rad/sec)')
title(['\angle L_u: Loop Phase at Controls (g_i = ',...
    num2str(gi),'), for z = ',num2str(z),',p = ',num2str(p)])
grid on
hold on


figure (12)
w  = logspace(-1,3, 2000);
sen_u = feedback(1,Lu_tf)
[sen_u_mag, sen_u_phase] = bode(sen_u,w);
[hinf_sen_u,fpeak_sen_u] = hinfnorm(sen_u)
hinf_su(:,counter) = hinf_sen_u
semilogx(w, 20*log10(sen_u_mag(1,:)))
ylabel('Magnitude (dB)')
xlabel('freq (rad/sec)')
title(['|S_u|: Loop Sensitivity at Controls (g_i = ',...
    num2str(gi),'), for z = ',num2str(z),',p = ',num2str(p)])
grid on
hold on

figure (13)
w  = logspace(-1,3, 2000);
comp_sen_u = feedback(Lu_tf,1)
[comp_sen_u_mag, comp_sen_u_phase] = bode(comp_sen_u,w);
[hinf_comp_sen_u,fpeak_comp_sen_u] = hinfnorm(comp_sen_u)
hinf_csu(:,counter) = hinf_comp_sen_u
semilogx(w, 20*log10(comp_sen_u_mag(1,:)))
ylabel('Magnitude (dB)')
xlabel('freq (rad/sec)')
title(['|T_u|: Loop Complementary Sensitivity at Controls (g_i = ',...
    num2str(gi),'), for z = ',num2str(z),',p = ',num2str(p)])
grid on
hold on

figure (14)
w  = logspace(-1,3, 2000);
comp_sen_u_o = (comp_sen_u/comp_sen_o)
[comp_sen_u_o_mag, comp_sen_u_o_phase] = bode(comp_sen_u_o,w);
semilogx(w, 20*log10(comp_sen_u_o_mag(1,:)))
ylabel('Magnitude (dB)')
xlabel('freq (rad/sec)')
title(['$\frac{|T_u|}{|T_o|} > 1 \; \forall \; \omega > 0 \; (p = ',...
    num2str(p),') \; for \; z = ',num2str(z),' $']...
    ,'Interpreter','latex')
grid on
hold on

% Control Response to Step Ref. Command at controls
figure (15)
tinc            = 0.0005;
```

```matlab
    tmax = 10;
    t               = 0:tinc:tmax;
    plot(t,step(sen_u,t));
    xlabel('Time (sec)')
    ylabel(['u(t)'])
    title(['Step S_u: Control Response u_p to Step Input Disturbance d_i (g_i = ',...
        num2str(gi),'), for z = ',num2str(z),',p = ',num2str(p)])
    grid on
    hold on

    % Output Response to Step Ref. Command at Controls
    figure (16)
    tinc            = 0.0005;
    tmax = 10;
    t               = 0:tinc:tmax;
    plot(t,step(comp_sen_u,t))
    xlabel('Time (sec)')
    ylabel(['y(t)'])
    title(['Step T_u: Control Response u to Step Input Disturbance d_i (g_i = ',...
        num2str(gi),'), for z = ',num2str(z),',p = ',num2str(p)])
    grid on
    hold on

    % Control Response to Step Ref. Command at Output
    figure (17)
    tinc            = 0.0005;
    tmax = 10;
    t               = 0:tinc:tmax;
    plot(t,step(sen_o,t));
    xlabel('Time (sec)')
    ylabel(['u(t)'])
    title(['Step S_o: Error Response e to Step Ref. Command r (g_i = ',...
        num2str(gi),'), for z = ',num2str(z),',p = ',num2str(p)])
    grid on
    hold on

    % Output Response to Step Ref. Command at Output
    figure (18)
    tinc            = 0.0005;
    tmax = 10;
    t               = 0:tinc:tmax;
    plot(t,step(comp_sen_o,t))
    xlabel('Time (sec)')
    ylabel(['y(t)'])
    title(['Step T_o: Output Response y to Step Ref. Command r (g_i = ',...
        num2str(gi),'), for z = ',num2str(z),',p = ',num2str(p)])
    grid on
    hold on


% impulse

% Control Response to Step Ref. Command at controls
    figure (19)
    tinc            = 0.0005;
    tmax = 10;
    t               = 0:tinc:tmax;
    plot(t,impulse(sen_u,t));
    xlabel('Time (sec)')
    ylabel(['u(t)'])
    title(['Impulse S_u: Control Response u_p to Impulse Input Disturbance d_i (g_i = ',...
        num2str(gi),'), for z = ',num2str(z),',p = ',num2str(p)])
    grid on
    hold on

    % Output Response to Step Ref. Command at Controls
    figure (20)
    tinc            = 0.0005;
    tmax = 10;
    t               = 0:tinc:tmax;
    plot(t,impulse(comp_sen_u,t))
    xlabel('Time (sec)')
    ylabel(['y(t)'])
    title(['Impulse T_u: Output Response y to Impulse Ref. Command r (g_i = '...
        ,num2str(gi),'), for z = ',num2str(z),',p = ',num2str(p)])
    grid on
    hold on

    % Control Response to Step Ref. Command at Output
    figure (21)
    tinc            = 0.0005;
    tmax = 10;
    t               = 0:tinc:tmax;
    plot(t,impulse(sen_o,t));
    xlabel('Time (sec)')
    ylabel(['u(t)'])
    title(['Impulse S_o: Error Response e to Impulse Ref. Command r (g_i = '...
        ,num2str(gi),'), for z = ',num2str(z),',p = ',num2str(p)])
    grid on
    hold on

    % Output Response to Step Ref. Command at Output
    figure (22)
    tinc            = 0.0005;
    tmax = 10;
    t               = 0:tinc:tmax;
    plot(t,impulse(comp_sen_o,t))
```

```
                         xlabel('Time (sec)')
                         ylabel(['y(t)'])
                         title(['Impulse T_o: Control Response u to Impulse Input Disturbance d_i (g_i = '...
                            ,num2str(gi),'), for z = ',num2str(z),',p = ',num2str(p)])
                         grid on
                         hold on

                       end
                    end
                 end
              end
           end
end
```