The Islamic University Of Gaza

Deanery of Higher Studies

Faculty of Engineering

Electrical Engineering Department

RIPPLE FREE DEADBEAT CONTROL FOR NONLINEAR SYSTEMS WITH TIME-DELAYS AND DISTURBANCES

Prepared By:

Safaa Ahmed El Najjar

Advisors:

Dr. Hatem Elaydi **Dr. Hala El Khozondar**

A Thesis Submitted to the Faculty of Engineering in Partial Fulfillment of the Requirements for the Degree of Master of Science in Electrical Engineering
The Islamic University of Gaza, Palestine

2013 - 1434

ABSTRACT

The design of controllers for the ripple-free deadbeat problem has long been investigated in the literature, but a need still exists to offer a better methodology for both performance and robustness for linear and nonlinear systems. This research proposes a new design methodology for ripple-free deadbeat control for nonlinear systems with time delays and disturbances. The proposed method combines two control laws, PID controller with Ripple-free Deadbeat controller. The new controller is applied to magnetic ball levitation CE 152 as a case study for nonlinear systems.

The deadbeat tracking and rejection formulation combines both the polynomial approach and the time domain approach, the time domain approach (PID control) is used to ensure the local behavior of the nonlinear system, while the polynomial approach (Diophantine equation) is used to provide deadbeat control to the nonlinear system. The nonlinear system is firstly controlled with PID law which tuned using Simulink Design Optimization™ software, then a second order linear model for the resulting response is estimated to model the nonlinear system. After that, a ripple-free deadbeat reference tracking and disturbance rejection control method for the system with time delays and disturbances is proposed.

The developed control is simulated using MATLAB software. Simulation results show that the output signal exactly tracks the input signal and rejects the disturbance in short settling time and at time-delay existence. The time domain specification for the output signal, control signal, and error signal are computed and satisfied the requirements and constraints.

ملخص البحث

منذ فترة طويلة والدراسات السابقة تسعى الى تصميم نظام التحكم المرهق الخالي من التموجات أو ما يسمى بنظام الضربة القاضية ، ولكن لا تزال ثمة حاجة لتقديم منهجية أفضل للأداء والمتانة على حد سواء وذلك للأنظمة الخطية وغير الخطية. يقدم هذا البحث منهجية جديدة لتصميم نظام تحكم مرهق خالي من التموجات لأنظمة غير خطية والتي تكون تحت تأثير التشويش و التأخير الزمني للموجات. هذا المتحكم الجديد يقوم بالدمج بين نظامين تحكم ألا وهما نظام مرهق خالي من التموجات و نظام تحكم بواسطة عوامل ضربيه لموجات الخطأ (PID). تم تطبيق المتحكم الجديد على جهاز رفع الكرة مغناطيسيا (CE 152) كحالة دراسية للأنظمة غير الخطية.

نظام التحكم المقترح يدمج بين نهجين: طريقة المجال الزمني (التحكم بواسطة PID) والأخرى بواسطة استخدام مجال المعادلات والتي تعتمد على حل معادلة الديوفانتاين (Diophantine). طريقة المجال الزمني ستستخدم لضمان الاستجابة الجزئية والمحلية (local behavior) للأنظمة غير الخطية، بينما طريقة مجال المعادلات ستستخدم لضمان الاستجابة التامة للنظام غير الخطي. بداية تم التحكم بالنظام غير الخطي بواسطة نظام تحكم (PID) والذي تم معايرته من خلال برمجية Simulink Design Optimization™، ثم تم تقدير موجة الاستجابة الناتجة عن المتحكم السابق كمعادلة من الدرجة الثانية وذلك كصيغة تقديرية خطية للنظام غير الخطي. بعد ذلك تم تقديم منهجية للتحكم بالنظام الخطي السابق تقديره باستخدام نظام التحكم المرهق الخالي من الموجات الذي يقوم بتتبع الاشارة المرجعية وتقليل التشويش في أقل زمن ممكن ، وذلك ضمن وجود التأخير الزمني.

تم محاكاة نظام التحكم الجديد باستخدام بيئة المحاكاة (MATLAB). أظهرت نتائج المحاكاة أن اشارة الاستجابة تتبع الاشارة المرجعية بدقة وتحذف التشويش في زمن قصير جدا وفي وجود التأخير الزمني . الخصائص الزمنية لإشارة الاستجابة واشارة التحكم واشارة الخطأ تم حسابها ووجدنا أنهم قد حققوا المطلوب واستوفوا الشروط.

DEDICATION

I dedicate this work to my parents, my brothers, my sisters, my husband and my lovely kid.

ACKNOWLEDGEMENT

First and foremost, all praise is due to Allah, the Almighty, who gave me the opportunity, strength, and patience to carry out this work.

I wish to express my deepest gratitude to my advisors, Dr. Hatem Elaydi and Dr. Hala El Khozondar for their professional assistance, support, advice and guidance throughout my thesis, and to my discussion committee, Dr. Basil Hamad and Dr. Iyad Abuhadrous for their acceptance to discuss my thesis.

Many thanks go to my husband, Dr. Muhammed, for his infinite support and patience. Words will not be enough to thank my family for their patience and encouragement during my thesis. Finally, thanks for everyone who has raised his hands and prayed ALLAH for my success.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABBREVIATIONS

ADC	Analog to Digital Converter
DAC	Digital to Analog Converter
RFDBC	Ripple-Free Deadbeat Controller
Maglev	Magnetic Levitation
Maglev CE152	Magnetic Ball Levitation CE152
PID	Proportional-Integral-Derivative
OS	Overshoot

CHAPTER 1 INTRODUCTION

The main topic of this thesis, optimal ripple-free deadbeat controllers for nonlinear systems with time-delays and disturbances, is introduced and investigated. An overview of the existing literature dealing with related topics is provided. The chapter ends with highlights of the contributions and the outline of this thesis.

1.1 General Introduction

Rapid advances in control theory have led to a rapid development in discrete-time nonlinear control systems[1]. Practically, every aspect of our day-to-day activities is affected by some type of control systems. Control systems are found in abundance in all sectors of industry, such as quality control of manufactured products, automatic assembly line, machine-tool control, computer control and many others[2].Today, almost all controllers are computer implemented meaning digital control [3]. Deadbeat controller is a type of digital controllers, which offers the fastest settling time. Therefore, deadbeat controller ensures that the error sequence vanishes at the sampling instants after a finite time.

Plants and processes are typically nonlinear; the most typical nonlinearity is saturation. Since, computer implemented controllers are a standard configuration, a theory for discrete-time nonlinear systems is very important in particular for control design purposes. Indeed, we cannot use linear control theory in cases where: large dynamic range of process variables is possible, multiple operating points are required, the process is operating close to its limits, small actuators cause saturation, etc. [4].

A control system is a device or set of devices (the controller) that manage the behavior of other devices(the plant).[5].

Systems with delays can be usually encountered in the real world. Time delay is defined as the required time between applying change in the input and notices its effect on the system output. When the system involves propagation and transmission of information on material, the delay is certain to occur. The presence of delays complicates the system analysis and the control design [6].

Systems under environment disturbances are commonly encountered. The problem of disturbance rejection arises in many fields [7].

1.2 Digital Control

Digital control is a branch of control theory that uses digital computers to act as system controllers. Depending on the requirements, a digital control system can be a microcontroller, DSP kit, FPGA kit, standard desktop computer and so on. Since a digital computer is a discrete system, the Laplace transform is replaced with the Z-transform. In addition, since a digital computer has finite precision, extra care is needed to ensure that the error in coefficients, A/D conversion, D/A conversion, etc. are not producing undesired or unplanned effects. For any digital controller, the output is a weighted sum of current and past input samples, as well as past output samples, this can be implemented by storing relevant values in any digital controller [3].

1.2.1 Features of Digital Controllers

- **Inexpensive**
- **Flexible:** easy to configure and reconfigure through software
- **Scalable:** programs can be scaled to the limits of the storage space without extra cost
- **Adaptable:** parameters of the program can be changed
- **Static operation:** digital computers are much less prone to environmental conditions than capacitors, inductors, etc.[7].

1.2.2 Digital Controller Requirements

- **A/D converter:** converts analog inputs to machine readable format (digital)
- **D/A converter:** converts digital output to a form that can be input to a plant
- **Software program:** that relates the outputs to the inputs [3]

1.3 Overview of the Literature:

The Deadbeat control, Time-delay handling, and Disturbance rejection are studied separately or partially in the following studies:

- H. Elaydi and R. A. Paz, (1998), proposed Optimal Ripple-free Deadbeat Disturbance Rejection Controllers for Systems with Time-delays. Affine parameterization of the Diophantine equation was used to solve this problem. Based on this parameterization, LMI conditions are used to provide optimal or constrained controllers for design quantities such as overshoot, undershoot and control amplitude [8]. However, they did not deal with nonlinear systems.

- Zongxuan Sun, Tsu-Chin Tsao, (2002), proposed a control design based on the internal model principle to track or reject nonlinear exogenous signal dynamics for nonlinear systems. Necessary condition to achieve asymptotic disturbance rejection based on the proposed control structure was first derived. It was shown that the necessary condition becomes sufficient for linear systems with linear disturbance dynamics. Inspired by the unique structure of the necessary condition, sufficient conditions were then proposed. Simulations of a nonlinear plant with chaotic disturbance showed the effectiveness of the proposed scheme [9]. But the presence of time-delays was not been considered.

- Steven Weidong Su, (2002), proposed Nonlinear Disturbance Rejection, studied the problem of robust disturbance rejection for nonlinear systems based on three different methods: H_∞ control, singular perturbation theory and multiple model adaptive control [7]. However, he did not handle the problem of time-delay presence. Also, his approach differs than ripple-free deadbeat control strategy.

- Al Batch, (2009), proposed Multi-rate Ripple-Free Deadbeat Control. Two degree of freedom controller for the fixed-order constrained optimization problem performance specifications utilizing the parameters of Diophantine equation to build a multi-rate ripple-free deadbeat control was presented. A combination between the concept of multivariable and robust single rate was utilized. A time-delay was also presented in simulation and was solved by using deadbeat controller based on solving Diophantine equation parameters [10]. However, it handled linear systems only.

- W. Sakallah, (2009), proposed Comparative Study For Controller Design Of Time-delayed Systems, which studied delay modeling using different approaches such as Pad’e approximation and Smith Predictor in continuous system and modified z-transform in discreet systems. Delays were assumed to be constant and known. The delays in the system are lumped in the plant model. This study showed the design of stable and optimal controller for time-delay systems using algebraic Riccati equation solutions and PID control [6]. However, ripple-free deadbeat strategy was not considered.

- M. Elammasie, (2011), proposed a ripple-free deadbeat controller of multi-rate nonlinear system. This controller was used to track a reference input with zero steady-state error in finite time and with minimum overshoot and minimum energy in the presence of a time-delay [11]. However, it did not handle systems under disturbances.

By contrast, this research will present a collective bunch of previous studies, that controls nonlinear systems providing time-delays and disturbances via a ripple-free deadbeat controller strategy.

1.4 Thesis Motivation

Advancement in control systems theory have progressed at enormous rates over the past four decades. Not long ago, issues such as stability and performance were the topics of the hour. Issues such as nonlinear systems control, minimum settling time, time delays and disturbance elimination could barely be addressed simultaneously.

There is a wide area of applications for control systems, where nonlinear phenomena cannot be avoided. These applications range from ship or submarine vehicle control, position control for robotic systems in a precision manufacturing process, autonomous vehicle systems, biochemical reactors, power plants and many others [12]. Therefore, control of nonlinear systems is an important area of control engineering with a range of potential applications.

Time-delay systems are unavoidable in many control systems. Most of the classical methods that deal the control system such as root locus and nyquist criterion, cannot deal with delay. Hence, a strong need to deal with them is exist.

A desire to develop new disturbance rejection methodologies that are simpler and more robust than existing methodologies is needed. Existing methodologies for rejecting known disturbances use complex paradigms. This thesis will use a simple and effective methodology to reject undesired disturbances.

Many processes is required to settle in minimum period, which is called in control system deadbeat systems, thus one of thesis aim is to achieve minimum settling time for these applications.

Hence, for all stated previous issues, this work is done. As seen from previous section, no complete dealt to all issues is investigated at once, so this thesis will deal all these topics at the same time, and with different and effective methodology.

1.5 Thesis Objectives

This Research presents an approach for the ripple-free deadbeat controller for nonlinear system in order to track random input signal in presence of time delays and known disturbance signals via tuning PID controller and solving Diophantine equation.

The control objectives are:

1. Designing Ripple-Free Deadbeat Controller that achieve good transient response in presence of time-delay for nonlinear system which makes the output signal y to track any random input signal with zero steady-state error in the smallest number of sampling instants (time steps)
2. Tuning PID controller
3. Minimize the effect of disturbances on the system, since disturbances may enter the system from many nodes such as actuators and sensors
4. Solving time-delayed nonlinear control problem under disturbance using ripple-free deadbeat controller

5. Studying the effect of time-delays and disturbances on the stability and performance
6. Realizing the developed controller using MATLAB software Toolbox.

1.6 Thesis Contribution

This thesis presents methodologies for designing internally stabilizing ripple-free deadbeat controllers for nonlinear systems to solve the tracking of an arbitrary reference signal and the attenuation of general disturbances. Nonlinearities on the system is handled by SIMULINK design optimization toolbox. Ripple–free deadbeat, disturbance attenuation, tracking problem is formulated based on the solution of the Diophantine equation. The approach also can handle systems with time delays, where the time delay is not an integer multiple of the sampling time.

1.7 Thesis Outlines

This thesis is organized as follow: The Second chapter gives backgrounds and details and explains the proposed methodology. The third chapter presents the magnetic ball levitation CE152. The fourth chapter presents the PID controller tuning of magnetic ball levitation CE152. The fifth chapter shows the methodology and approach. The sixth chapter shows simulations and results, and the final chapter concludes this thesis.

CHAPTER 2 BACKGROUND

The purpose of this chapter is to emphasis the importance of the concept of deadbeat control for nonlinear systems. This chapter gives background for the forthcoming chapters. Previous studies which mentioned in Chapter 1 have not treated ripple-free deadbeat controller for nonlinear systems, disturbance rejection and time delay simultaneously. Therefore, my work will be started with Elaydi's and Elammassie's results and apply Diophantine equations to the estimated linear plant after applying PID controller optimization. Therefore, this chapter will cover briefly the deadbeat controller for linear systems, nonlinear systems, PID optimization, time delay and disturbance systems, designing steps to deadbeat magnetic ball levitation CE152, and the necessary assumptions.

2.1 Nonlinear Systems

In mathematics, a nonlinear system is a system which does not satisfy the superposition principle, or whose output is not directly proportional to its input. Less technically, a nonlinear system is any problem where the variable(s) to be solved cannot be written as a linear combination of independent components [13] such as squared terms.

As an example for nonlinear systems, magnetic ball levitation CE152 will be considered in this thesis. Any other nonlinear system could be manipulated in the same manner as approached in this thesis. Motion equation of Magnetic ball levitation CE152 model can be described as:

$$m_k\ddot{x} + K_{fv}\dot{x} = \frac{i^2 K_c}{(x - x_0)^2} - m_k g \qquad (2.1)$$

where m_k is the mass ball, g is gravity, x is ball position, i is the coil current, K_c is coil constant, K_{fv} is viscous friction coefficient, and x_0 is position offset.

A more intelligent answer is that thorough understanding of linear phenomena and linear mathematics is an essential prerequisite for progress in the nonlinear area. Moreover, many important physical systems are "weakly nonlinear", in the sense that, while nonlinear effects do play an essential role, the linear terms tend to dominate the physics, and so, the system is essentially linear. As a result, such nonlinear phenomena are best understood as some form of perturbation of their linear approximations. The advent of powerful computer algorithms has excited a veritable revolution in our understanding of nonlinear problems.

Hence, the approach in this thesis is to estimate linear behaved system to apply it in further steps. This is done via applying PID feedback control law tuned by Simulink Design Optimization™ software, the behaved simulated output is estimated as a second order system:

$$\frac{{\omega_n}^2}{s^2 + 2\eta\omega s + {\omega_n}^2} \tag{2.2}$$

where ω_n is the natural frequency and η is the damping ratio.

2.2 Time-Delay Systems

Systems with delays can be usually encountered in the real world. Time delay is defined as the required time between applying change in the input and notices its effect on the system output. When the system involves propagation and transmission of information or material, the delay is certain to occur. The presence of delays complicates the system analysis and the control design.

In this thesis, delays will be lumped into a single delay in the feedback loop, representing delay in control action or delayed measurements.

In the continuous-time system case, the delay is expressed as infinite dimensions e^{-sh}. In the discrete-time system case, finite dimensional z^{-h} can be considered as part of the system. The transfer function of a delay can be represented in M-file using transfer function expression "tf(numertator,denominator,'ioDelay',T_d)" or using SIMULINK block (Transport delay).

2.3 Systems with Disturbances

Tracking or rejection of known exogenous signals with known generating dynamics is of major concern in feedback control design. Linear feedback control based on the internal model principle achieves asymptotic performance for linear systems with linear exogenous signal dynamics. This thesis presents a control design based on the internal model principle to track or reject known exogenous signal dynamics for nonlinear systems. Necessary condition to achieve asymptotic disturbance rejection based on the proposed control structure is first derived. Simulations of a nonlinear plant with disturbance show the effectiveness of the proposed scheme.

2.4 Ripple-free Deadbeat Control

The study of deadbeat control of discrete systems dates back to the early 1950's. Deadbeat control makes the output of the systems coincide with the reference input signal in a finite period of time. Deadbeat control achieves exact settling after a finite number of discrete sampling instants.

However, there may exist ripple (non–zero deviation between the output response and the reference input signal) in the continuous plant between the discrete sampling instants. This inter-sample ripple is undesirable. There are two sources of inter–sample ripple: The first source is due to the failure of the design to cause the control signal to settle. This problem is due to the design allowing cancellation of the plant zeros. The second source of inter–sample ripple is due to the system being unable to track a moving reference between samples (lack of a continuous internal model).

To obtain a ripple–free deadbeat design, a continuous internal model and cancellation of no plant zeros are required, so that the response has the zero ripple property.

Two approaches for obtaining a deadbeat feedback controller for linear discrete time systems have been developed.

The first approach is called the ***time domain approach*** and is based on providing minimum control energy. This approach uses the controllability matrix and the state space realization in designing deadbeat controllers. This approach shows the trade-off between the sampling period, the number of steps to settle, and the magnitude of the control signal

The other approach is the ***polynomial method***. In this approach, the design procedure operates on the transfer function, here, the error between the system output and the reference signal is made to decay to zero in a finite number of sampling intervals. The appropriate controller involves the cancellation of both the plant poles and zeros.

One drawback of the polynomial approach has been that the control signal may not attain its steady state form in finite time, hence ripples may show up in the plant continuous time output (in between the sampling instants). This occurs due to cancellation of plant zeros by controller poles. This results in controller modes that may be excited by the reference signal but are not affected by feedback.

This problem is overcome by the design of ripple-free controller which is used in this thesis. Thus, the proposed controller combines two deadbeat controllers, one of them will use polynomial approach and the other will use time domain approach.

An important point to emphasis is that the proposed deadbeat control is based on ***internal model principle*** which could combine time delay and disturbance dynamics in the system to be controlled, and hence all part of thesis topic are taken into account on problem formulation.

2.5 Designing steps to deadbeat Magnetic ball levitation

Magnetic ball levitation CE152 will be used as a case study for nonlinear system. The proposed approach is completed following these steps

- Deriving the input/output relation of maglev sub-models (D/A converter, Power amplifier, ball & coil subsystem, Position sensor, A/D converter)
- Applying PID controller tuning via Simulink Design Optimization™ software

- Evaluating second order estimation to the step response of maglev with PID controller

- Formulating the plant to be deadbeat controlled in presence of time delay and disturbance
- Evaluating deadbeat controller for estimated model

- Simulating the developed control using MATLAB software

2.6 Design Constraints

The necessary and sufficient conditions for the ripple-free continuous response are that [8]:

- The continuous system is controllable with discrete input at period T

- The plant plus the Hold plus the Controller must have a continuous internal model of the reference input that is observable from the output

- The closed-loop system be internally stable

CHAPTER 3 MAGNETIC BALL LEVITATION MODEL CE152

Levitation is the process by which an object is suspended by a physical force against gravity, in a stable position without solid physical contact. A number of different techniques have been developed to levitate matter, including the aerodynamic, magnetic, acoustic, electromagnetic, electrostatic, gas film, and optical levitation methods [14]. Magnetic levitation systems have many varied uses such as in frictionless bearings, high-speed maglev passenger trains, levitation of wind tunnel models, vibration isolation of sensitive machinery, levitation of molten metal in induction furnaces and levitation of metal slabs during manufacturing. These systems have nonlinear dynamics that are usually open loop unstable and, as a result, a high performance feedback controller is required to control the position of the levitated object. Due to inherent nonlinearities associated with electromechanical dynamics, the control problem is usually quite challenging to the control engineers, since a linear controller is valid only about a small region around a nominal operating point [15]. This chapter will talk about magnetic ball levitation CE152 as one of Magnetic levitation systems.

3.1 Introduction to Magnetic Ball Levitation CE152

The Magnetic Levitation Apparatus shows control problems with nonlinear, unstable systems. The apparatus consists of a steel ball held in a magnetic field produced by a current-carrying coil. At equilibrium, the downward force on the ball due to gravity (its weight) is balanced by the upward magnetic force of attraction of the ball towards the coil. Any imbalance, the ball will move away from the set-point position. The basic control task is to control the vertical position of the freely levitating ball in the magnetic field of the coil. The Magnetic Levitation Apparatus is a nonlinear, dynamic system with one input (set point) and two outputs (ball position and coil current) [16].

The CE 152 Magnetic Levitation Model, shown in Figure (3.1) and its block diagram, shown in Figure (3.2) is an unstable system designed for studying system dynamics and experimenting with number of different control algorithms [17].

Figure (3.1): CE152 magnetic ball levitation.

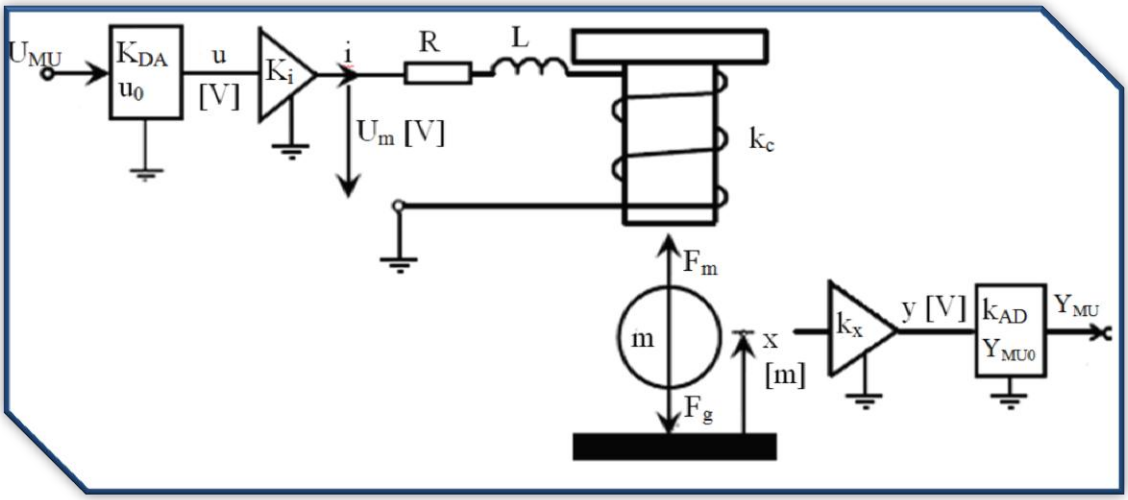


Figure (3.2): Principal scheme of the magnetic levitation model.

The scheme shows, that the model interface can be considered at two different levels:

- physical level - input and output voltage to the coil power amplifier and from the ball position sensor.
- logical level - voltage converted by the data acquisition card and scaled to ± 1 machine unit [MU]. Because simulation is in MATLAB environment, the later convention is used.

3.2 Model Analysis

The CE152 model, shown in Figure (3.2) consists of the following sub models [18]:

- D/A converter.
- Power amplifier.

- Ball & coil subsystem.
- Position sensor.
- A/D converter.

3.2.1 D/A Converter

D/A Converter, shown in Figure (3.3) has model output voltage u, where the input is U_{MU}. The digital to analog converter gain K_{DA}, with offset U_0. Its output is defined by Eq. (3.1). (Note: this block is added for real time application)

$$u = K_{DA} * U_{MU} + U_0 \quad (3.1)$$

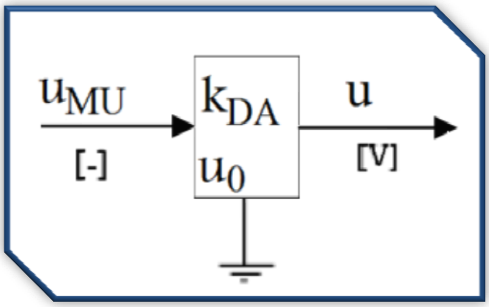
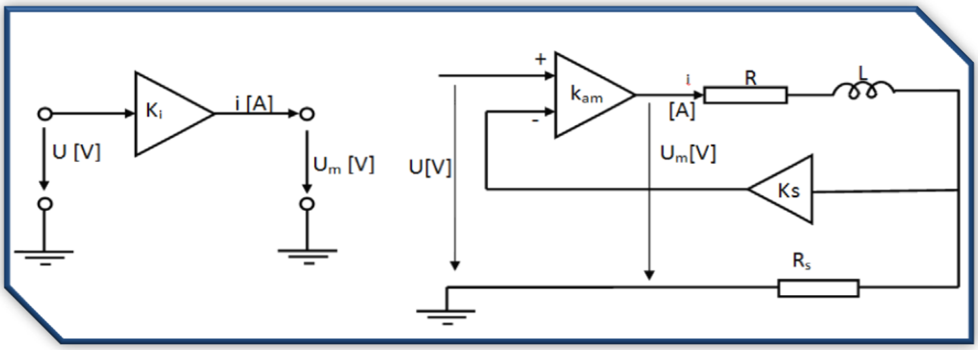


Figure (3.3): D/A Converter

3.2.2 Power Amplifier:

The power amplifier is designed as a source of constant current with the feedback current stabilization, as shown in Figure (3.4).



(a) Power amplifier **(b) Internal structuree**

Figure (3.4): The power amplifier and its internal structure.

Relation between input current and output voltage from power amplifier will be found as following:

From internal structure, Figure (3.4.a):

$$u_m = iR + L\frac{di}{dt} + R_s i \, , \quad (3.2)$$

$$u_m = K_{am}\left(u - K_s(iR_s)\right) \qquad (3.3)$$

from Eq. (3.2) and (3.3):

$$\therefore \; iR + L\frac{di}{dt} + R_s i \; = K_{am}\left(u - K_s(iR_s)\right)$$

$$\Rightarrow IR + LIS + R_s I = K_{am}\,U - K_{am}\,K_s R_s I$$

$$\Rightarrow \frac{I}{U} = \frac{K_{am}}{R}\left(\frac{1}{\frac{L}{R}S + 1 + \frac{R_s + K_{am}\,K_s R_s}{R}}\right)$$

And if $R \gg R_s + K_{am}\,K_s R_s$, as the case in this system,

$$\therefore \frac{I}{U} = \frac{K_{am}}{R}\left(\frac{1}{\frac{L}{R}S + 1}\right) = K_i\left(\frac{1}{T_a\,S + 1}\right) \qquad (3.4)$$

Where K_i is Gain, and T_a is the time constant.

Equation (3.4) will be used in the next chapter in the derivation of the state space of linearized model of magnetic ball levitation CE 152

3.2.3 Ball and Coil Subsystem:

The motion equation is based on the balance of all forces acting on the Ball. We have two forces: gravity force F_g, electromagnetic force F_m and the difference is the acceleration force F_a , as shown in Figure (3.5), equation of free body diagram will be derived where I: the coil current, K_c : coil constant, x_0: position offset, and K_{fv}: damping constant.

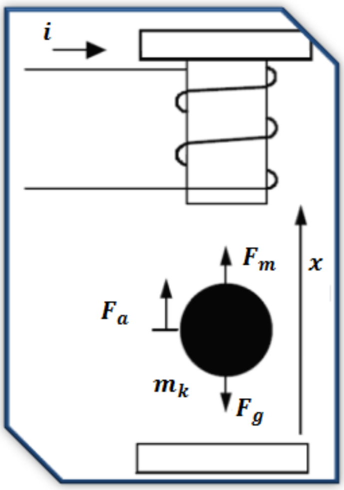


Figure (3.5): Free diagram of the ball and the forces.

According to Newton's second law of motion, the acceleration of an object as produced by a net force is directly proportional to the magnitude of the net force, in the same direction as the net force, and inversely proportional to the mass of the object [19]. By mathematics equations:

The net force is

$$F_a = F_m - F_g \tag{3.5}$$

where F_a: Acceleration force,

$$F_a = m_k \ddot{x} \tag{3.6}$$

F_m: Magnetic force,

$$F_m = \frac{i^2 K_c}{(x - x_0)^2} \tag{3.7}$$

F_g: Gravitational force,

$$F_g = m_k g \tag{3.8}$$

Substituting Eq. (3.6), (3.7) and (3.8) into (3.5) results in:

$$m_k \ddot{x} = \frac{i^2 K_c}{(x - x_0)^2} - m_k g \tag{3.9}$$

Limits of the ball movements and ball damping is taken into account. So, to model the damping, the term K_{fv} is introduced into the equation

$$m_k \ddot{x} + K_{fv} \dot{x} = \frac{i^2 K_c}{(x - x_0)^2} - m_k g \tag{3.10}$$

Equation (3.10) will be used in the next chapter in the derivation of the state space of linearized model of magnetic ball levitation CE 152.

3.2.4 Position Sensor.

The position sensor, shown in Figure (3.6) which used to measure the ball position x, has model output voltage Y, gain K_x and offset Y_0.

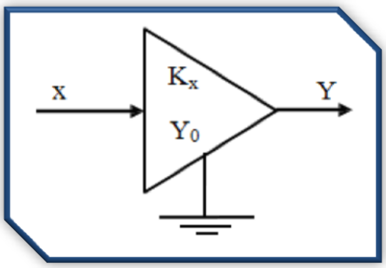


Figure (3.6): Position sensor subsystem.

The output is defined in Eq. (3.11)

$$Y = K_x\ x + Y_0 \tag{3.11}$$

3.2.5 A/D Converter.

The A/D Converter, shown in Figure (3.7) has model output voltage Y_{MU}, input Y, gain K_{AD}, and offset Y_{MU0}. (Note: this block is added for real time application)

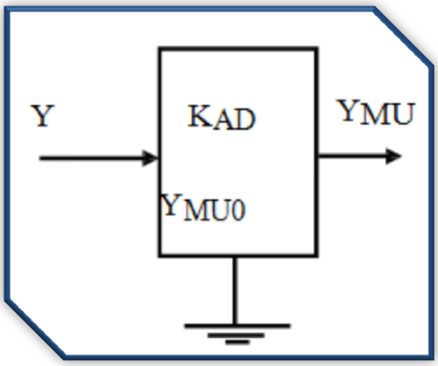


Figure (3.7): Position sensor subsystem.

The output is defined in Eq. (3.12)

$$Y_{MU} = K_{AD}\ Y + Y_{MU0} \tag{3.12}$$

3.2.6 Magnetic constant K_c

Coil Energy:

$$W_m = \frac{1}{2} L i^2 \tag{3.13}$$

where L: inductance,

$$L = \frac{N^2}{R} \tag{3.14}$$

and R: Resistance,

$$R = \frac{l}{\mu \text{A}} \tag{3.15}$$

where l: length of the coil, N: number of coil turns, μ: permeability of the coil core, A: Cross section area

Substituting Eq. (3.14) and (3.15) into (3.13) results in:

$$W_m = \frac{1}{2}\left(\frac{\mu A N^2}{l}\right) i^2 = \frac{\mu A N^2 i^2}{2l} \qquad (3.16)$$

By taking the derivative of last equation with respect to l , we get the magnetic force, that's:

$$F_m = \frac{dW_m}{dl} = \frac{\mu A N^2 i^2}{2l^2} = \frac{K_c\, i^2}{l^2}$$

Hence,

$$K_c = \frac{\mu A N^2}{2} \qquad (3.17)$$

3.3 Complete Modeling:

The final block diagram of the magnetic levitation model CE 152 is given in SIMULINK model as shown in Figure (3.8):

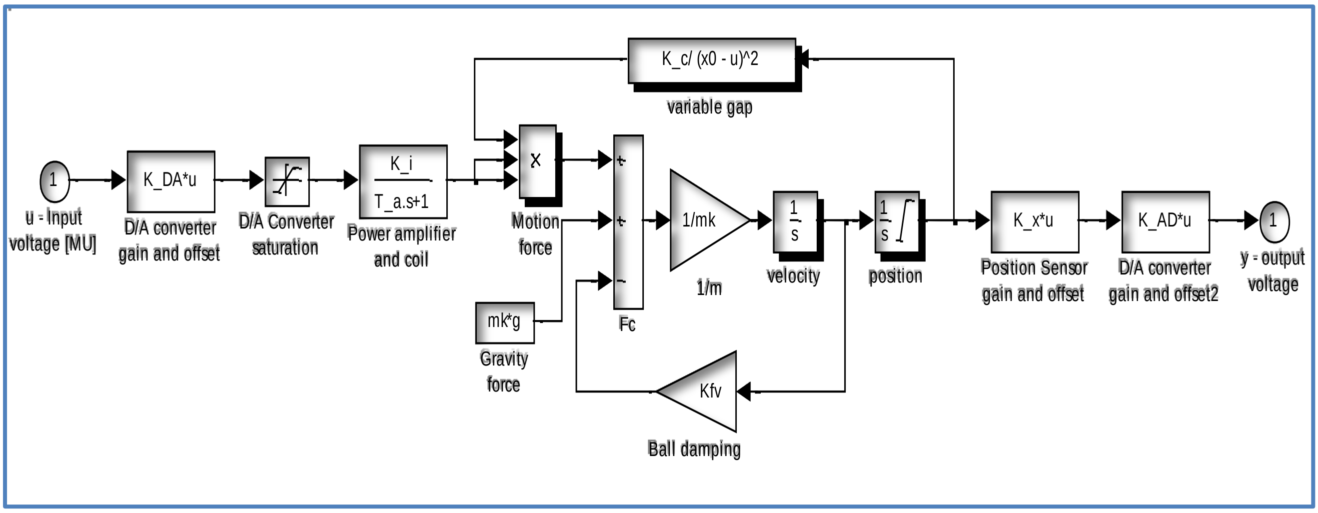


Figure (3.8): The complete model of magnetic levitation CE152

Table (3.1) shows CE152 model parameters that used in simulating the system.

Table (3.1): Parameters of magnetic ball levitation CE 152 [16]

Parameter	Symbol	Value
ball diameter	D_k	12.7x10-3 m
ball mass	m_k	0.0084 kg
distance from the ground and the edge of the magnetic	T_d	0.019 m
distance of limits= 0.019 - D_k	L	0.0063 m
gravity acceleration constant	g	9.81 m.s^-2
maximum DA converter output voltage	U_{DAM}	5 V

coil resistance	R_c	3.5 Ω
coil inductance	L_c	30 x10^{-3} H
current sensor resistance	R_s	0.25 Ω
current sensor gain	K_s	13.33
power amplifier gain	R_{am}	100
maximum power amplifier output current	I_{am}	1.2 A
amplifier time constant= $L_c/\{(R_c + R_s) + R_s * K_s * K_{am}\}$	T_a	3x10^{-3} s
amplifier gain= $K_{am}/\{(R_c + R_s) + R_s * K_s * K_{am}\}$	K_i	0.3
viscose friction	K_{fv}	0.02 N.s/m
converter gain	K_{DA}	10
Digital to Analog converter offset	U_0	0 V
Analog to Digital converter gain	K_{AD}	0.2
Analog to Digital converter offset	Y_{MU0}	0 V
position sensor constant	K_x	797.4603
coil bias	x_0	6 x10^{-3} m
Aggregated coil constant	K_f	0.606 x10^{-6} N/V
coil constant = K_f / K_i^2	K_c	2.16 x10^{-6} N/V

3.4 Open Loop System:

Using SIMULINK, the open loop step response of the model is shown in Figure (3.9)

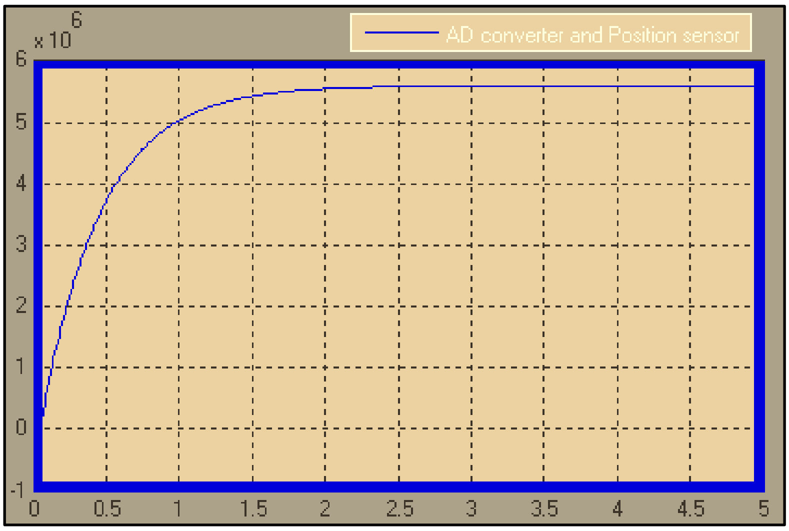


Figure (3.9): Open loop response of magnetic levitation CE152

From last figure, we deduce that this system needs a controller in order to prevent saturation besides performance modification. This aim is achieved in the next two chapters.

CHAPTER4 PID OPTIMIZATION

A proportional–Integral–Derivative (PID) controller is a generic control loop feedback mechanism widely used in industrial control systems. A PID controller calculates an "error value" as the difference between a measured process variable and a desired setpoint. The controller attempts to minimize the error by adjusting the process control inputs [20].

Simulink Design Optimization™ software provides interactive tools, functions, and Simulink blocks for tuning design parameters in a Simulink model to meet objectives such as improved system performance and minimized energy consumption. Using design optimization techniques, we can meet both time- and frequency-domain constraints such as overshoot and phase margin [21].

Since the developed RFDBC algorithm which depends on polynomial approach is applied only to linear systems; firstly, we seek to obtain a linear expression for the process. Also, most processes in real world are unstable. Hence, the purpose of this chapter is to solve stability and performance issues, and at the same time obtain an equivalent linear expression for the plant.

4.1 Nonlinear Systems:

Nonlinearity is ubiquitous in physical phenomena. Fluid and plasma mechanics, gas dynamics, elasticity, relativity, chemical reactions, combustion, ecology, biomechanics, and many, many other phenomena are all governed by inherently nonlinear equations. For this reason, an ever increasing proportion of modern mathematical research is devoted to the analysis of nonlinear systems [22].

Nonlinear systems are vastly more difficult to analyze. In the nonlinear problems, many of the most basic questions remain unanswered: existence and uniqueness of solutions are not guaranteed; explicit formula are difficult to come by; linear superposition is no longer available; numerical approximations are not always sufficiently accurate; etc. [23].

A more intelligent answer is that thorough understanding of linear phenomena and linear mathematics is an essential prerequisite for progress in the nonlinear area.

Moreover, many important physical systems are "weakly nonlinear", in the sense that, while nonlinear effects do play an essential role, the linear terms tend to dominate the physics, and so, the system is essentially linear. As a result, such nonlinear phenomena are best understood as some form of perturbation of their linear approximations.
The advent of powerful computer algorithms has excited a veritable revolution in our understanding of nonlinear problems.

4.2 Stability and performance issues:

Given a physical system, the ultimate goal in control engineering is to achieve control specifications that mostly are stated in two criteria: stability and performance.

Internal stability is usually the most fundamental requirement in systems. For unstable plants; stabilization, tracking problem and disturbance rejection led to the most used structure in control theory: Feedback structure. One of these structures is PID configuration.

Performance defines the exact way the designer desires the system to operate. Performance cannot always be easily described. Several restrictions can be put on the behavior of the system. Such restrictions include overshoot, undershoot, settling time, rise time and amplitude of control signal.

4.3 PID controller theory:

The PID controller calculation (algorithm) involves three separate constant parameters, and is accordingly sometimes called three-term control: the proportional, integral and derivative values, denoted P, I, and D. Heuristically, these values can be interpreted in terms of time: P depends on the present error, I on the accumulation of past errors, and D is a prediction of future errors, based on current rate of change. The weighted sum of these three actions is used to adjust the process as shown in Figure (4.1) [20].

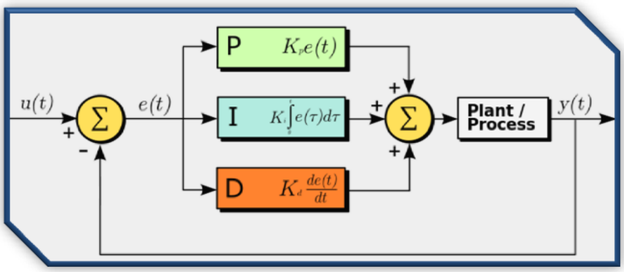


Figure (4.1): Block diagram of PID controller

Defining $u(t)$ as the controller output, the final form of the PID algorithm is:

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau)d\tau + K_d \frac{d}{dt} e(t)$$

where

K_p: Proportional gain, a tuning parameter
K_i: Integral gain, a tuning parameter
K_d: Derivative gain, a tuning parameter
t: Time or instantaneous time
τ: Variable of integration; takes on values from time 0 to the present

4.3.1 PID Tuning

Tuning a control loop is the adjustment of its control parameters to the optimum values for the desired control response. PID tuning is a difficult problem, because it must satisfy complex criteria within the limitations of PID control.

There are several methods for tuning a PID loop. The most effective methods generally involve the development of some form of process model, then choosing P, I, and D based on the dynamic model parameters. Manual tuning methods can be relatively inefficient.

Thus, PID optimization software is used to ensure consistent results. Software packages will gather the data, develop process models, and suggest optimal tuning. Simulink Design Optimization™ software is used for this purpose.

4.4 PID Optimization:

Simulink Design Optimization™ software optimizes model response by formulating the requirements into a constrained optimization problem. It then solves the problem using optimization methods.

For time-domain requirements, the software simulates the model during optimization, compares the current response with the requirement and uses gradient methods to modify design variables to meet the objectives [21].

4.5 Magnetic Ball Levitation CE152 (Nonlinear System) Tuning:

We model a CE 152 Magnetic Levitation system where the controller is used to position a freely levitating ball in a magnetic field. The control structure for this model is fixed and the required controller performance can be specified in terms of an idealized time response. Controller parameters are tuned via Design optimization tool.

4.5.1 Control Problem Description:

As shown at the end of chapter 3, open loop nonlinear system is unstable and its specifications are not adequate. Thus, stability and performance criteria will be solved. Then, an equivalent, linear expression of the resulting system will be derived and used at the next chapter.

Hence, the requirement of the controller is positioning the ball at any arbitrary stable location in the magnetic field and moving it from one position to another.

Simulink Design Optimization™ and numerical optimization is ideally suited to tune the PID coefficients because:

- The system dynamics are complex enough to require effort and time for analysis if we approach the problem using conventional control design techniques
- The controller structure is fixed
- The step response we require from the system is known.

4.5.2 Model Structure:

The model includes the following blocks shown in Figure (4.2):

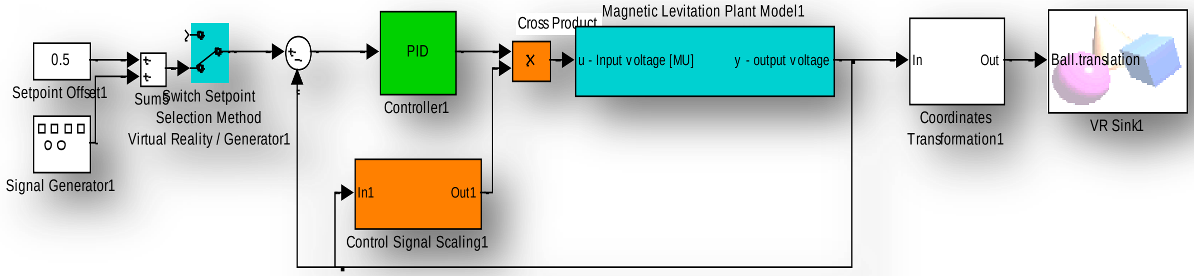


Figure (4.2): CE152 with PID controller

- **Controller block,** which is a PID controller. This block controls the output of the Plant subsystem
- **Control Signal Scaling block,** Figure (4.3),

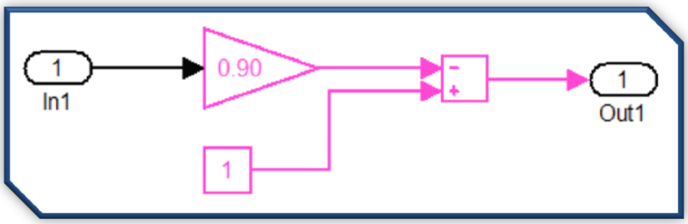


Figure (4.3): Control Signal Scaling Block

is included because as the force from the coil decays according to an inverse square law, larger voltages are required, the further the ball is from the coil. Hence, the control signal is scaled to account for this action

- **Input Signal block,** applies a reference signal to the plant
- **Plant subsystem**, is the Magnetic Ball Levitation CE152. In fact, Simulink Design Optimization™ tool linearizes the nonlinear plant in each iteration, then computes tuned gains K_p, K_i, K_d.

4.5.3 Design Requirements

The plant output must meet the following step response requirements:

- **Rise time** less than 0.2 seconds
- **Settling time** less than 0.7 seconds
- **Overshoot** less than 5%

4.5.4 Specify Step Response Requirements

1) Add a Signal Constraints block to the model at signal position we seek to optimize. We select **View > Library Browser > Simulink Design Optimization > Signal Constraints** as shown in Figure (4.4)

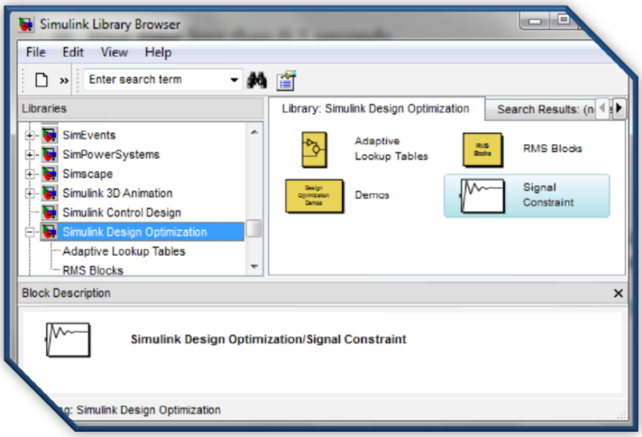


Figure (4.4): Control Signal Scaling Block

2) Connect the block to the output signal, Figure (4.5)

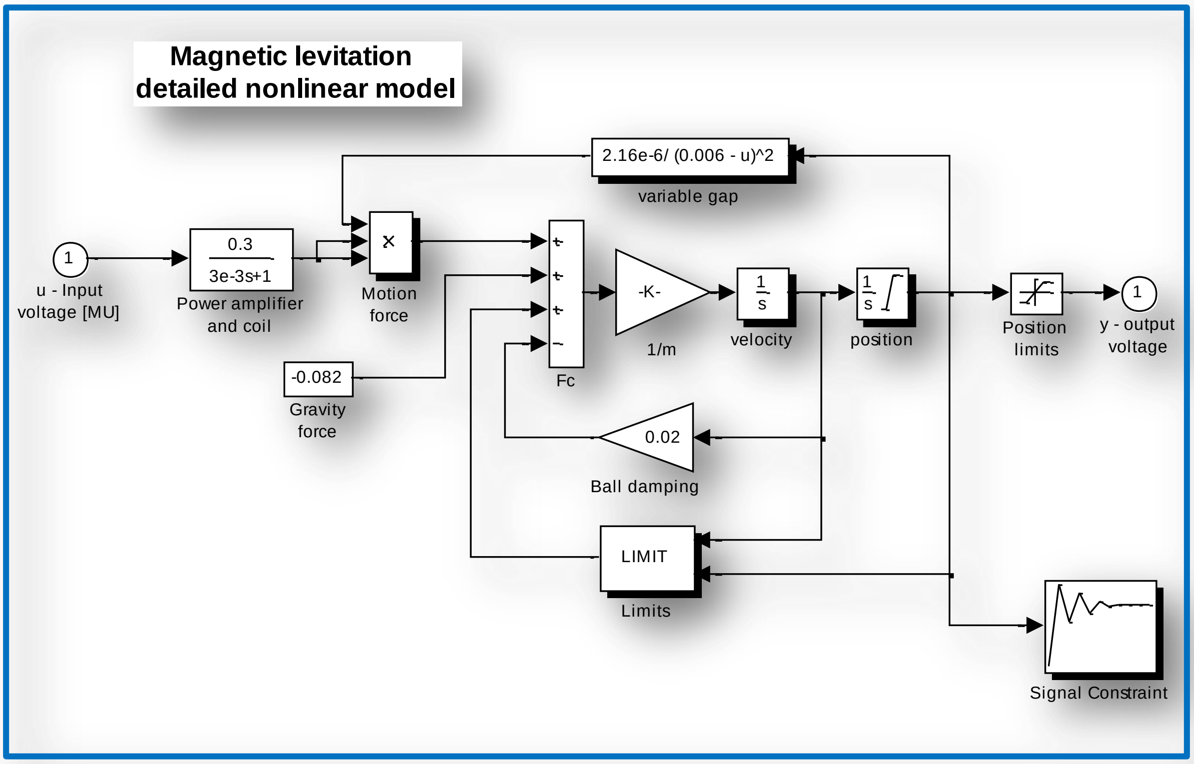


Figure (4.5): CE152 with PID controller Tuning

3) Double click on signal **constraint block** > **Goals** > **Specify step response characteristics**, as shown in Figure (4.6) and Figure (4.7)

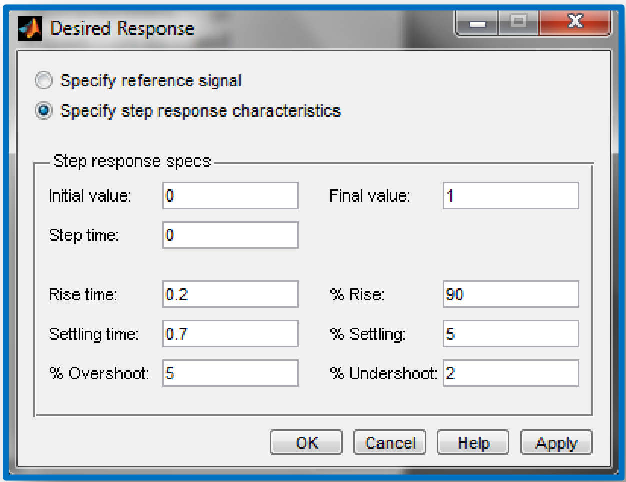


Figure (4.6): Step Response Characteristics

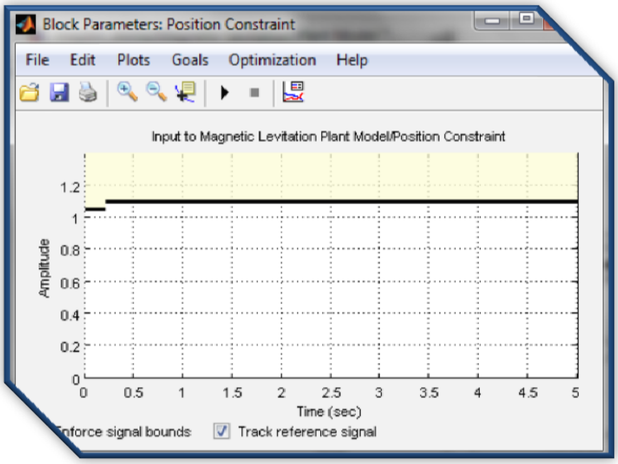


Figure (4.7): Step Response

4.5.5 Defining Tuned Parameters

1) Double click on **signal constraint** block **> Optimization** > **Tuned Parameters**
2) Add desired system parameters to be optimized

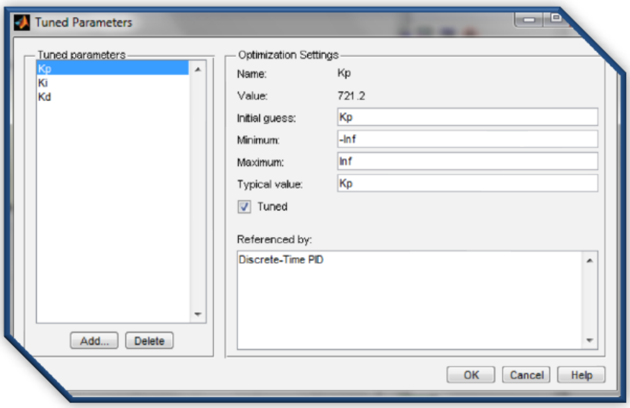


Figure (4.8): Design parameters

4.5.6 Running the optimization

1) Double click on **signal constraint** block **> Optimization** > **Start,** An optimization progress window opens. Default optimization solver Gradient descent modifies the design variables to reduce the distance between the simulated response and the design requirement line segments. After the optimization completes, the optimization progress window, Figure (4.9) resembles the tuned parameters:

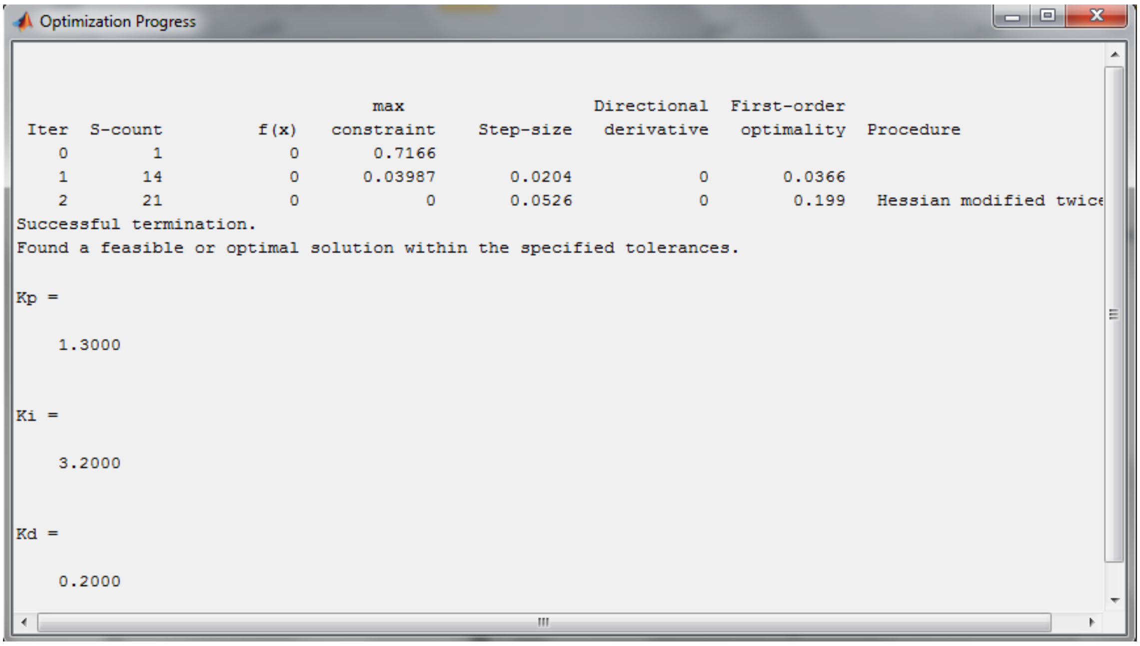


Figure (4.9): Optimization progress window

Hence:

$$K_p = 1.3, K_i = 3.2, \ K_d = 0.2 \qquad (4.1)$$

2) Plot current response,

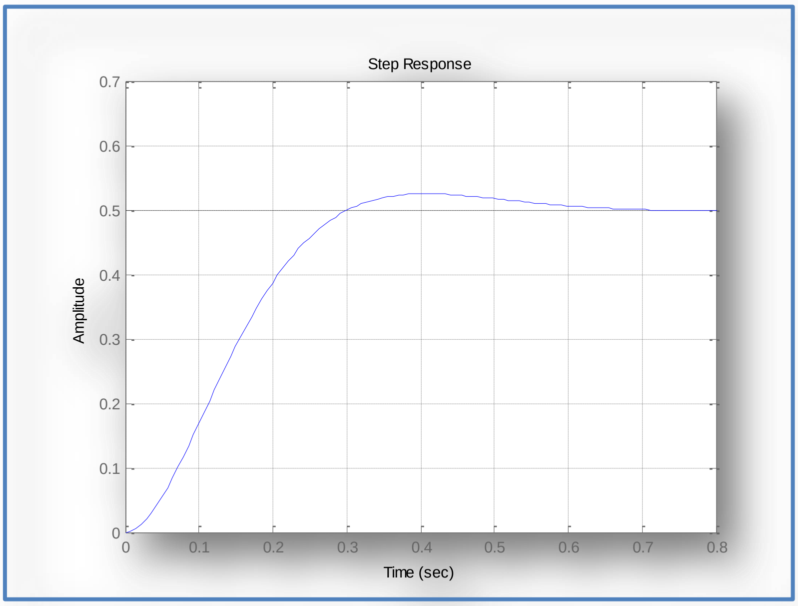


Figure (4.10): Output signal after PID optimization

From Figure (4.10), notice that: T_{rise}= 0.2 sec, $T_{settling}$= 0.7 sec, %Os = 5%

4.6 Magnetic Ball Levitation CE152 (Linear System) Tuning:

To obtain an equivalent, linear plant that needed in the following chapter, depending on mechanism of PID optimization algorithm, an equivalent system is derived in this section.

When Simulink Design Optimization™ toolbox is used for optimizing nonlinear plants, optimization algorithm is based on finding firstly an equivalent linear plant at each iteration, and then computes tuned gains $\mathrm{K_p}$, $\mathrm{K_i}$, $\mathrm{K_d}$.

Thus, to accomplish the purpose of this section, a linearized plant is derived, then using obtained PID tuned parameters, a closed loop system is computed to use it in the next chapter.

The magnetic ball levitation CE152, shown in Figure (4.11) is characterized by third order differential equation (since it has three states), as shown in Eq.(3.10):

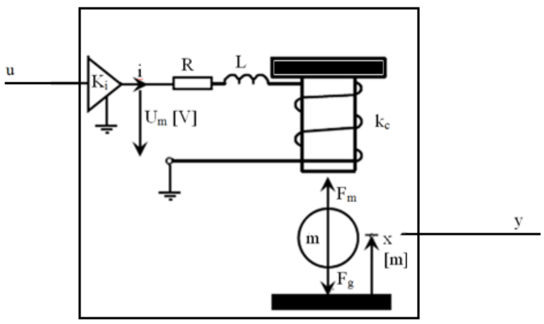


Figure (4.11): Plant of magnetic Ball levitation CE152

$$m_k\ddot{x} = \frac{i^2 K_c}{(x - x_0)^2} - m_k g - K_{fv}\dot{x}$$

To obtain a transfer function expression $G(s)$,

$$G(s) = C(SI - A)^{-1}B + D \qquad (4.2)$$

we must find firstly the state space model (A, B, C, D),

∴ let

$$x_1 = x \qquad (4.3)$$

$$x_2 = \dot{x}_1 = \dot{x} \qquad (4.4)$$

$$x_3 = i \qquad (4.5)$$

$$\Longrightarrow \dot{x}_2 = \ddot{x},\ \dot{x}_3 = \dot{i}(t) \qquad (4.6)$$

By substituting Eq.(4.3),(4.4), (4.5) and (4.6) into Eq.(3.10):

$$\therefore\ \dot{x}_2 = \frac{{x_3}^2 K_c}{m_k (x_1 - x_0)^2} - g - \frac{K_{fv}}{m_k} x_2 \qquad (4.7)$$

From Eq.(3.4, we have $(T_a s + 1) I = K_i U$, and by converting s-domain to time-domain implies that:

$$T_a \dot{\imath}(t) + i(t) = K_i u(t)$$

$$\Rightarrow \imath(t)^{\dot{}} = \frac{K_i u(t) - i(t)}{T_a}$$

$$\dot{x}_3 = \frac{K_i u(t) - x_3}{T_a} \qquad (4.8)$$

Thus

$$\begin{bmatrix} \dot{x}_1 = x_2 \\ \dot{x}_2 = \dfrac{{x_3}^2 K_c}{m_k (x_1 - x_0)^2} - g - \dfrac{K_{fv}}{m_k} x_2 \\ \dot{x}_3 = \dfrac{K_i u(t) - x_3}{T_a} \end{bmatrix} \qquad (4.9)$$

Notice that Eq. (4.9) is nonlinear (nonlinear system); thus we need to linearize it, that the term

$$\frac{i^2 K_c}{m_k (x_1 - x_0)^2} = f(x_1, i)$$

around certain operating point (a,b). Using Taylor series expansion for function of two variables (two terms are sufficient description),

$$\therefore\ \dot{x}_2 = \frac{b^2 K_c}{m_k (a - x_0)^2} + \left[\frac{-2b^2 K_c}{m_k (a - x_0)^3}\right] x_1 + \left[\frac{2bK_c}{m_k (a - x_0)^2}\right] x_3 - g - \frac{K_{fv}}{m_k} x_2 \quad (4.10)$$

$$= \left(\frac{b^2 K_c}{m_k (a - x_0)^2} - g\right) + \left[\frac{-2b^2 K_c}{m_k (a - x_0)^3}\right] x_1 - \frac{K_{fv}}{m_k} x_2 + \left[\frac{2bK_c}{m_k (a - x_0)^2}\right] x_3 \quad (4.11)$$

Evaluating the constant term in Eq.(4.15), using ball position at $x = 0$ (the ball is freely at the ground of CE152), means no applied coil current ($i(t)$) nor motion velocity ($\dot{x}$) or acceleration ($\ddot{x}$) that's:

$$x_1 = 0 \Rightarrow x_2 = x_3 = 0 \qquad (4.12)$$

Hence, substituting Eq.(4.12) into Eq.(4.11) results in:

$$0 = \left(\frac{b^2K_c}{m_k(a-x_0)^2} - g\right) + \left[\frac{-2b^2K_c}{m_k(a-x_0)^3}\right](0) - \frac{K_{fv}}{m_k}(0) + \left[\frac{2bK_c}{m_k(a-x_0)^2}\right](0)$$

$$0 = \left(\frac{b^2K_c}{m_k(a-x_0)^2} - g\right) \tag{4.13}$$

Substituting Eq.(4.13) into Eq.(4.11) implies:

$$\dot{x}_2 = \left[\frac{-2b^2K_c}{m_k(a-x_0)^3}\right]x_1 - \frac{K_{fv}}{m_k}x_2 + \left[\frac{2bK_c}{m_k(a-x_0)^2}\right]x_3 \tag{4.14}$$

This equation is linear and thus the system is now linear by substituting Eq.(4.18) into Eq.(4.8):

$$\begin{bmatrix} \dot{x}_1 = x_2 \\ \dot{x}_2 = \left[\frac{-2b^2K_c}{m_k(a-x_0)^3}\right]x_1 - \frac{K_{fv}}{m_k}x_2 + \left[\frac{2bK_c}{m_k(a-x_0)^2}\right]x_3 \\ \dot{x}_3 = \frac{K_i u(t) - x_3}{T_a} \end{bmatrix} \tag{4.15}$$

Plant output equation is Eq. (3.11)

Notice that Eq.(4.15) describes CE152 model without ADC, DAC and position sensor models shown in Figure (4.10). Hence, modifying it to obtain the total system shown in Figure (3.2) is done as following:

$$\begin{bmatrix} \dot{x}_1 = x_2 \\ \dot{x}_2 = \left[\frac{-2b^2K_c}{m_k(a-x_0)^3}\right]x_1 - \frac{K_{fv}}{m_k}x_2 + \left[\frac{2bK_c}{m_k(a-x_0)^2}\right]x_3 \\ \dot{x}_3 = \frac{K_i K_{DA} U_{MU}(t) - x_3}{T_a} \end{bmatrix} \tag{4.16}$$

System output equation is Eq.(3.12).

Hence, state space model of Linearized CE152 model around certain operating point (a,b) is written as:

$$\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\\ \dot{x}_3\end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ \frac{-2b^2K_c}{m_k(a-x_0)^3} & \frac{-K_{fv}}{m_k} & \frac{2bK_c}{m_k(a-x_0)^2} \\ 0 & 0 & \frac{-1}{T_a}\end{bmatrix}\begin{bmatrix}x_1\\ x_2\\ x_3\end{bmatrix} + \begin{bmatrix}0\\ 0\\ \frac{K_iK_{DA}}{T_a}\end{bmatrix}U_{MU}(t) \qquad (4.17)$$

$$[Y_{MU}] = [K_xK_{AD} \quad 0 \quad 0]\begin{bmatrix}x_1\\ x_2\\ x_3\end{bmatrix} + [0]U_{MU}(t) \qquad (4.18)$$

In order to compute Eq.(4.17) and (4.18) at specified operating point (a,b), we must find variables a,b. System parameters are substituted directly from Table(3.1).
From Eq.(4.13):

$$g = \frac{b^2K_c}{m_k(a-x_0)^2}$$

$$\Rightarrow b^2 = \frac{m_kg\,(a-x_0)^2}{K_c}$$

$$\therefore b = \pm(a-x_0)\sqrt{\frac{m_kg}{K_c}} \qquad (4.19)$$

Notice that state space model differs as (a,b) is changed. Let us use the center position as typical case for equilibrium point with following properties:

- ☒ Ball position at the center of gap distance L, i.e. $x = a = \frac{L}{2} = \frac{0.019}{2} = 0.0095\,m$
- ☒ Equilibrium state $\Rightarrow velocity, \dot{x} = accelaration, \ddot{x} = 0, i.e. \dot{x}_1 = \dot{x}_2 = 0$
- ☒ Current has a constant value $i(t) = x_3 = b$, Thus,current change (derivative), $\dot{I} = \dot{x}_3 = 0$

In this case $(a = 0.0095)$,

$$b = \pm(a-x_0)\sqrt{\frac{m_kg}{K_c}}$$

$$= \pm\left(0.0095 - 8.26e(-3)\right)\sqrt{\frac{0.0084 * 9.81}{6.8823e(-6)}}$$

$$= \pm 0.13568 \quad (-ve\ is\ unreilable\ )$$

$\therefore b = 0.13568$

$\therefore$ The resulting state space model of linearized magnetic ball levitation CE152 around $(a,b) = (0.0095, 0.13568)$ is:

$$\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\\ \dot{x}_3\end{bmatrix}=\underbrace{\begin{bmatrix}0 & 1 & 0\\ -15821.6191 & -2.381 & 144.5962\\ 0 & 0 & -53493.0994\end{bmatrix}}_{A}\begin{bmatrix}x_1\\ x_2\\ x_3\end{bmatrix}+\underbrace{\begin{bmatrix}0\\ 0\\ 158714.0259\end{bmatrix}}_{B}U_{MU}(t)$$

$$[Y_{MU}]=\underbrace{[159.4921\quad 0\quad 0]}_{C}\begin{bmatrix}x_1\\ x_2\\ x_3\end{bmatrix}+\underbrace{[0]}_{D}U_{MU}(t)$$

Hence, from Eq.(4.2), transfer function expression $P(s)$ is given by (using MATLAB command window):

```
>>A=[0,1,0;-15821.6191,-2.381,144.5962;0,0,-53493.0994];
>> B=[0;0;158714.0259];
>> C=[159.4921,0,0];
>> D=0;
>> [num,den]=ss2tf(A,B,C,D);
>> Ps=tf(num,den)              % Plant

Transfer function:
        1.301e-008 s + 3.66e009
------------------------------------------------------------
s^3 + 5.35e004 s^2 + 1.432e005 s + 8.463e008
```

Thus

$$\mathrm{P(s)}=\left.\frac{1.301\mathrm{e}-008\ \mathrm{s}\ +\ 3.66\mathrm{e}009}{\mathrm{s}^3\ +\ 5.35\mathrm{e}004\mathrm{s}^2\ +\ 1.432\mathrm{e}005\ \mathrm{s}\ +\ 8.463\mathrm{e}008}\right|_{(a,b)=(0.0095,0.13568)} \quad (4.20)$$

From Eq. (4.1), PID optimal parameters for x at center are used to obtain controller transfer function as following [25]:

$$C(s)=K_p+\frac{K_i}{s}+K_d s=\frac{K_d s^2+K_p s+K_i}{s} \quad (4.21)$$

$$\therefore C(s)|_{(a,b)=(0.0095,0.13568)}=\frac{0.2s^2+1.3s+3.2}{s} \quad (4.22)$$

Hence, from Figure (4.2) and using Eq.(4.24) and Eq.(4.26), closed loop transfer function expression $G(s)$ is given by (using MATLAB command window):

```
>> Cs=tf([0.2 1.3  3.2],[1 0]);   % PID Controller
>> Gs=feedback(Cs*Ps,1)

Transfer function:
 3.903e-010 s^3 + 1.098e008 s^2 + 3.294e009 s + 4.319e010
--------------------------------------------------------------------------------
s^4 + 5.35e004 s^3 + 1.1e008 s^2 + 4.141e009 s + 4.319e010

>> step(0.5*Gs)
```

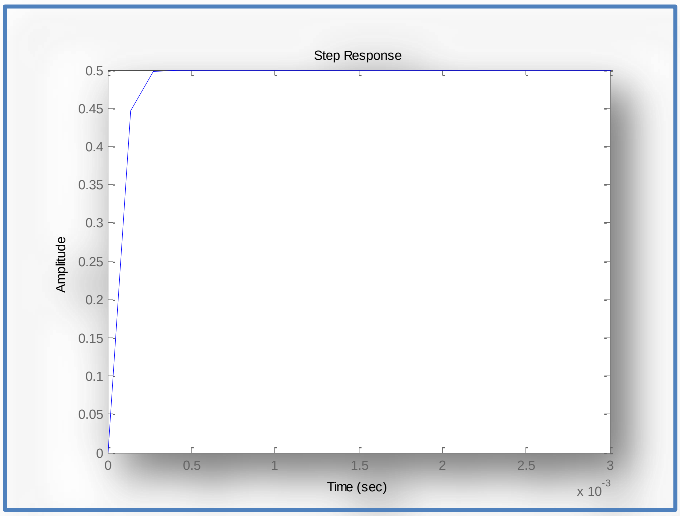


Figure (4.12): Output signal from linearized, tuned plant loop (approximately)

Note that using the statement "Gs=feedback(Cs*Ps,1)" is not exact equivalent for the loop in model shown in Figure (4.2) (cross product results in nonlinear expression in R,Y; so I ignore this inner loop, but unfortunately, its effects is not seen in the response shown in Figure (4.12). Hence, this response is not accurate, and another approach will be developed.

4.7 Approximated Tuned Magnetic Ball Levitation CE152 (Nonlinear System):

As shown in previous section, when modeling the system via linearizing plant and approximating model (Figure (4.5)) results in a response nearly far from the original response, Figure (4.9). Hence, a need to obtain accurate equivalent model is necessary. One method to accomplish this, is to expect the total system that has the desired response.

Target response is shown in Figure (4.10), notice that we can model it using second order prototype, Eq. (2.2) as following:

$$G(s)_{estimated} = \frac{{\omega_n}^2}{s^2 + 2\eta\omega_n s + {\omega_n}^2} \qquad (4.23)$$

As seen in subsection (4.4.7), from Figure (4.9), system specifications are:

$T_{settling}$= 0.7 sec, %OS = 5%

In order to obtain the transfer function, we need to translate the system specifications to η, ω_n as following:

$$\eta = \sqrt{\frac{(\ln(OS))^2}{(\ln(OS))^2 + \pi^2}} = 0.7797 \qquad (4.24)$$

$$T_{settling} = \frac{4}{\eta \omega_n} \quad \Rightarrow \quad \omega_n = 25.6509 \qquad (4.25)$$

Substituting Eq.(4.24) and Eq.(4.25) into Eq.(4.23), using MATLAB command window:

```
>> OS=5;
>> Ts=0.7;
>> zeta=(-log(OS/100))/(sqrt(pi^2+log(OS/100)^2));
>> wn= 4/(zeta*Ts);
>> num=wn^2;
>> den=[1  2*zeta*wn  wn^2];
>> Gs_estimated=tf(num,den)
Transfer function:
     12
--------------------
s^2 + 5 s + 12
>> step(0.5*Gs_estimated)
```

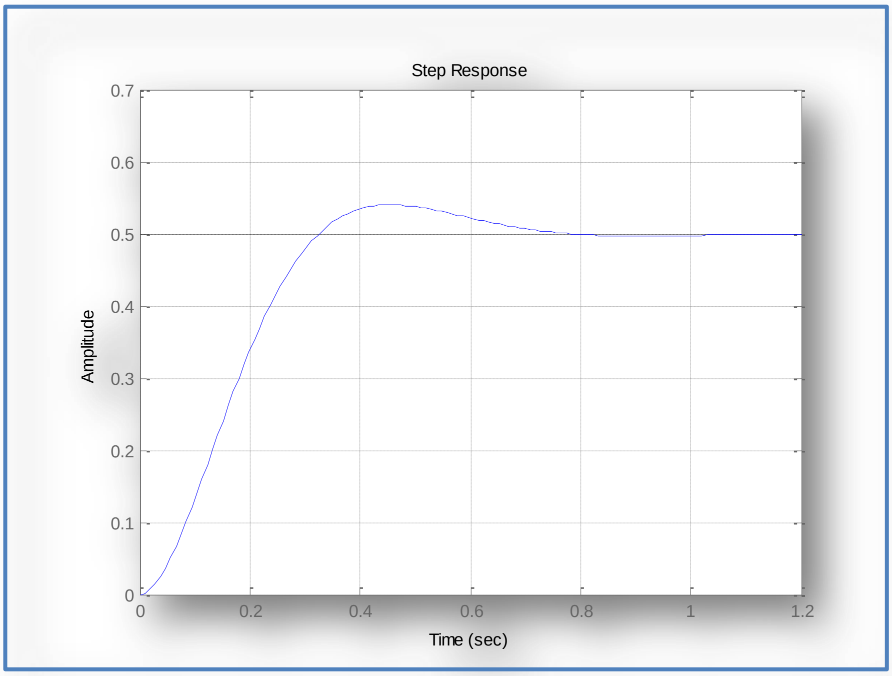


Figure (4.13): Output signal of estimated tuned CE152 system

Comparing Figure (4.13) with Figure (4.10),

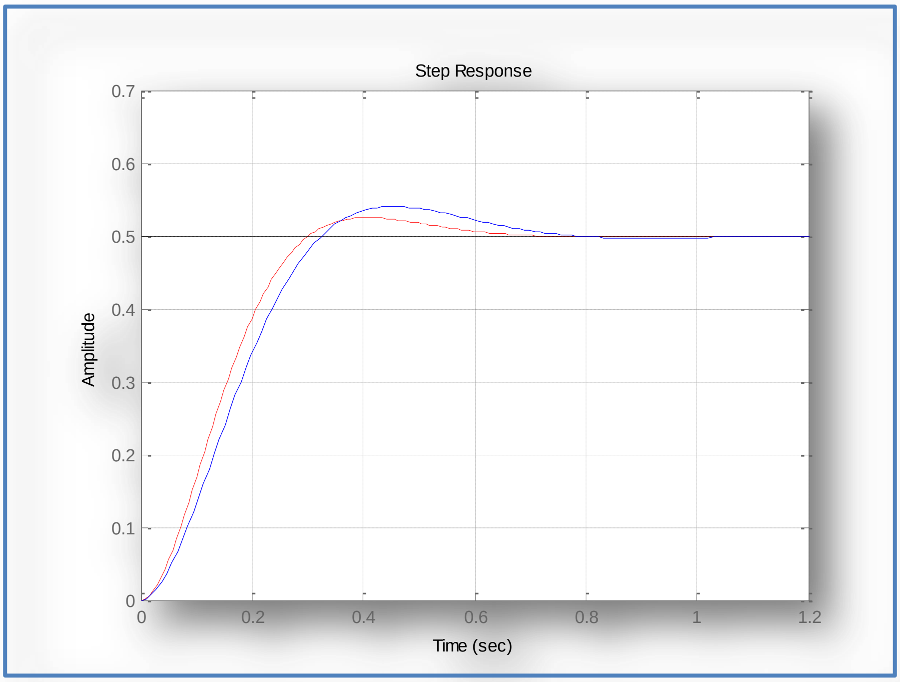


Figure (4.14): Output signal of tuned CE152 system (red) and its estimation (blue)

we notice that they are roughly the same, i.e. estimated tuned plant, Eq.(4.26),

$$G(s)_{estimated} = \frac{12}{s^2 + 5s + 12} \qquad (4.26)$$

is so accurate and coincides with the origin system. Hence, it will be used in following chapters as a linear mirror for the target system.

CHAPTER 5 RIPPLE-FREE DEADBEAT CONTROL

The purpose of this chapter is to state useful definitions and used assumptions. In addition, a complete description for the problem of ripple-free deadbeat control is given. Also, the approach to solve this problem for linear and nonlinear systems is developed.

5.1 Introduction

Time delay, fast response without oscillation, disturbance attenuation, and tracking of a general reference input signal problems are considered here via Ripple-free Deadbeat control.

Mathematical model of a physical system plays an important role in controller design. The synthesis of a controller is often based on the model of the system, therefore, the construction of an effective controller relies heavily on the accuracy of this model. Thus, the first step is to develop a method that makes it easier to understand and incorporate the model in the controller design procedure. Section 5.3 is designated for this aim.

Time delays are frequently encountered in industrial processes. Time delays limit the achievable bandwidth and the allowed maximum. Also, time delays often significantly complicate the analysis and computation analysis in system design. Hence, for simplicity, time delay is assumed to be lumped and incorporated in system transfer function.

Disturbances are undesirable inputs to the system. Disturbances may enter the system from many nodes such as actuators and sensors. The effect of disturbances on the system should be minimized using control. The error signal shows the degree of success in minimizing the effect of the disturbances on the system.

Ripple-free deadbeat control can be solved by two approaches: Time Domain approach which solves the problem in state space setting (depends on minimum energy solution), and Polynomial approach which solves the problem in Transfer Function settings (depends on the solution of Diophantine equation), leading to optimization problem that deal with robustness and performance objectives. Nonlinear input-output map can be realized by polynomial model. Hence, the used approach in this thesis is the second one, whereas it can handle nonlinear systems with time delays, beside disturbances, all these details are included in the problem because Diophantine equation depends on the internal structure of the system.

The time optimal control problem (Deadbeat control) is defined as finding a finite control sequence such that the error sequence is short as possible. This often produces a violent transient response and may drive the system into saturation. To avoid this problem, a restriction is imposed on the amplitude of the control sequence at the expense of the settling time. The constrained time optimal control problem has many solutions. The polynomial approach can specify all stabilizing optimal control sequences, thus it is helpful to have an algorithm that finds the control sequence that satisfies all the constraints.

5.2 Time-Delay Systems

Systems with delays can be usually encountered in the real world. Time delay is defined as the required time between applying change in the input and notices its effect on the system output. When the system involves propagation and transmission of information or material, the delay is certain to occur. The presence of delays complicates the system analysis and the control design.

In this thesis, delays will be lumped into a single delay in the feedback loop, representing delay in control action or delayed measurements.

In continuous time-delay system, the delay is modeled by using Pad'e approximation approach. In discrete time-delay systems, a modified *z*-transform is used to model constant delays, which are expressed as non-integer multiples of the sampling periods.

In the continuous-time system case, the delay is expressed as infinite dimensions e^{-sh}. In the discrete-time system case, finite dimensional z^{-h} can be considered as part of the system. The transfer function of a delay can be represented in M-file using transfer function expression " tf(numertator,denominator,'ioDelay',T_d) " or using SIMULINK block shown in Figure (5.1).

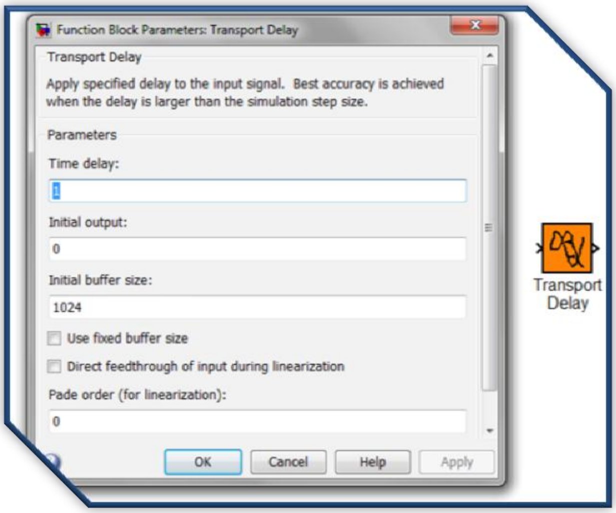


Figure (5.1): Time-Delay representation in SIMULINK models

5.3 System Formulation

The approach considers a single-input, single-output (SISO), nonlinear continuous-time plant.

Deadbeat control problem is solved here by polynomial approach that requires the plant to be linear. Estimated equivalent linear plant in section 4.7 is improved to accomplish this requirement.

Figure (5.2) shows the deadbeat controller C(z) for linear plant P(s),

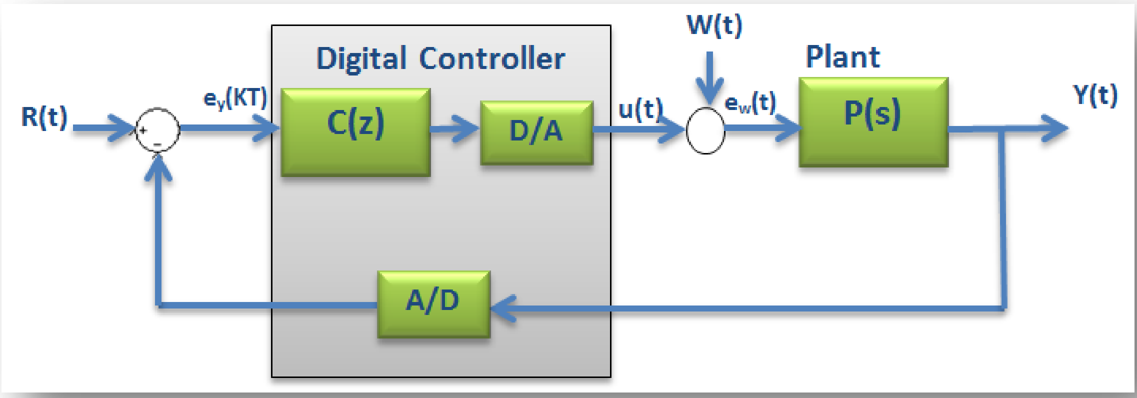


Figure (5.2): Closed loop control system

Now, the plant is assumed to be linear, minimal, and strictly proper, and having transfer function of the form:

$$\frac{y(s)}{e_w(s)} = (e^{-T_d s})P(s) = (e^{-T_d s})\frac{N_p(s)}{D_p(s)} \qquad (5.1)$$

where $T_d \geq 0$ is the value of the pure time delay of the system. The delay arise in a number of ways in continuous-time systems (e.g. a transport delay, communication delay etc.). It may also arises as a computational delay associated with the time required for the computer to calculate the control value based on the present data. Here, T_d is assumed to be lumped of all delay types that can be occurred in the system.

Digital controller, Ripple-free Deadbeat controller C(z) is designed in computer, thus firstly a digital form of plant should obtained to be compatible with C(z), i.e. P(z).

The delay element z^{-T_d} can be defined in terms of q variable, where $q = z^{-1}$, in order to simplify mathematical computations only. Hence, we can write the following transfer functions in q-domain (by replacing z^{-n} by q^n, where n is integer number):

- ***Plant:***

$$\frac{y(q)}{e_w(q)} = \frac{N_p(q)}{D_p(q)} \qquad (5.2)$$

where $N_p(q)$: is the plant numerator polynomial of m^{th} degree order, $N_p(0)$=0

$D_p(q)$: is the plant denominator polynomial of n^{th} degree order, $N_p(q)$ is assumed to be coprime (no common factors) with $D_p(q)$.

- ***Reference signal***: is assumed to be arbitrary and has the following representation:

$$r(q) = \frac{N_r(q)}{D_r(q)} \tag{5.3}$$

where $\mathrm{N_r(q)}$ is assumed to be coprime with $\mathrm{D_r(q)}$, and σ is the order of $\mathrm{D_r(q)}$.

- ***Disturbance signal***: is assumed to be arbitrary and has known representation:

$$w(q) = \frac{N_w(q)}{D_w(q)} \tag{5.4}$$

where $\mathrm{N_w(q)}$ is assumed to be coprime with $\mathrm{D_w(q)}$.

- ***Tracking error, $e_y(q)$*** is defined as:

$$e_y(q) = r(q) - y(q) \tag{5.5}$$

- ***Control error, $e_w(q)$*** is defined as:

$$e_w(q) = u(q) + w(q) \tag{5.6}$$

❖ **Definition 5.1 [26]:**

The Ripple-Free Deadbeat Control (RFDBC) problem is defined as finding a controller

$$C(q) = \frac{u(q)}{e_y(q)} = \frac{N_c(q)}{D_c(q)} \tag{5.7}$$

where $N_c(q)$ and $D_c(q)$ are coprime and $D_c(0)$=1 (monic; to obtain a prototype transfer function form), such that:

i. Closed-loop system is internally stable ⇒ no cancellations of unstable poles or zeros.
ii. Tracking error settles to zero in N discrete steps ⇒ $e_y(q)$ is a polynomial of N degree at most.
iii. $u(q), y(q)$ settle to their final form in N discrete steps ⇒ $u(q)$ is a rational function whose poles are a subset of the poles of $r(q)$ and $w(q)$, and $y(q)$ is a

rational function whose poles are a subset of the poles of $r(q)$. It also implies that $D_c(q)$ and $N_p(q)$ are coprime.

5.4 Approach to Solve Ripple-free Deadbeat Problem:

The used deadbeat approach in this thesis depends on internal model principle. Internal model principle states that in order for a plant to track an input with steady state error, the internal forward-path must have a model of the input as a factor. Thus either the plant or the controller must have this as a factor in the denominator.

Systematic steps is followed to complete the design, which are:

1) Step1: Factor plant and reference transfer functions
2) Step2: Obtain controller structure
3) Step3: Solve Diophantine equation
4) Step4: Optimize the solution of Diophantine equation

Step1:Factoring plant and reference transfer functions

Factoring plant denominator polynomial and reference numerator polynomial into stable and unstable polynomial,

$$D_p(q) = D_{ps}(q)D_{pu}(q) \tag{5.8}$$

$$N_r(q) = N_{rs}(q)N_{ru}(q) \tag{5.9}$$

Where:

$D_{ps}(q)$: plant denominator stable factors

$D_{pu}(q)$: plant denominator unstable factors

$N_{rs}(q)$: reference numerator stable factors

$N_{ru}(q)$: reference numerator unstable factors

Note: In q-domain notation, stable factor means having a root outside the unit circle.

Thus, it is important to compute system dynamics (poles) which contain plant, reference and disturbance signal poles. The approach depends on factoring these polynomials as following:

$$D_r(q) = \widetilde{W}(q)\tilde{X}(q)\tilde{Y}(q)\widetilde{D}_r(q) \tag{5.10}$$

$$D_w(q) = \widetilde{W}(q)\tilde{X}(q)\tilde{Z}(q)\widetilde{D}_w(q) \tag{5.11}$$

$$D_{pu}(q) = \widetilde{W}(q)\tilde{Y}(q)\tilde{X}(q)\widetilde{D}_{pu}(q) \tag{5.12}$$

Where:

- $\widetilde{W}(q)$: a polynomial resulting from multiplying common roots between $D_r(q), D_w(q), D_{pu}(q)$.
- $\tilde{X}(q)$: a polynomial resulting from multiplying common roots between $D_r(q), D_w(q)$.
- $\tilde{Y}(q)$: a polynomial resulting from multiplying common roots between $D_r(q), D_{pu}(q)$.
- $\tilde{Z}(q)$: a polynomial resulting from multiplying common roots between $D_w(q), D_{pu}(q)$.

Steps 2, 3 and 4: are discussed by details in the following sections as follows.

5.5 Obtaining controller structure

In a hybrid system, the necessary and sufficient conditions for the ripple-free continuous response are that:

1) The continuous system is controllable with discrete input at period T, and
2) The plant plus the Hold plus the Controller must have a continuous internal model of the reference input that is observable from the output, and
3) The closed-loop system be internally stable

5.5.1 Deadbeat Tracking Problem

❖ **Assumptions 5.1:**

i. Assume that $D_p(q) = D_{pu}(q), i.e. D_{ps}(q) = 1$. This assumption will increase the settling time one sample for every additional pole in $D_{pu}(q)$ but increases the accuracy of approach results
ii. To assure that the plant will be able to track the desired input signal and reject the disturbance; $N_p(q)$ and $D_r(q)$ are assumed to be coprime. This means that plant transfer function has no transmission zeros.

❖ **Definition 5.2 [26]: Ripple-free Deadbeat Tracking Problem**

The Ripple-free deadbeat control problem has a solution if and only if the polynomials $N_p(q)$ and $D_r(q)$ coprime. Moreover, all solutions are of the form

$$N_c(q) = D_{ps}(q)\, Q_n(q) \qquad (5.13)$$

$$D_c(q) = \widetilde{D}_r(q)\, Q_d(q) \tag{5.14}$$

where $Q_n(q)$ *and* $Q_d(q)$ *are polynomial solutions of the Diophantine equation*

$$[N_p(q)]\, Q_n(q) + [\widetilde{D}_r(q) D_{pu}(q)]\, Q_d(q) = N_{rs}(q) \tag{5.15}$$

such that $Q_d(0) = 1$

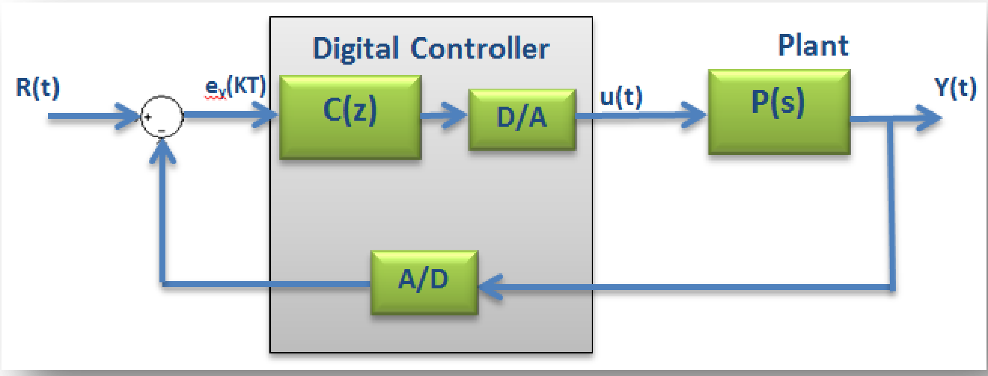


Figure (5.3): Ripple-free deadbeat tracking model

In order to provide ripple-free tracking, we should examine tracking error, control and output signals in q-domain as following:

Output signal should track exactly reference signal after N steps at most. Output error must go to zero at finite number of steps. To obtain deadbeat (fastest) response, all poles should be at the origin, that's the denominator in q-domain equals one. The following equations illustrate these properties:

- ***Tracking error signal calculation:***

From Eq.(5.5),

$$e_y(q) = r(q) - y(q)$$

$$\Rightarrow \quad e_y(q) = r(q) - \left\{\frac{C(q) * P(q)}{1 + C(q) * P(q)}\right\} r(q)$$

$$= r(q)\left\{1 - \frac{C(q) * P(q)}{1 + C(q) * P(q)}\right\}$$

$$= r(q)\left\{\frac{1 + C(q) * P(q) - C(q) * P(q)}{1 + C(q) * P(q)}\right\} = r(q)\left\{\frac{1}{1 + C(q) * P(q)}\right\}$$

$$= \frac{N_r(q)}{D_r(q)} \left\{ \frac{1}{1 + \frac{N_c(q)}{D_c(q)} * \frac{N_p(q)}{D_p(q)}} \right\}$$

$$= \frac{N_r(q)}{D_r(q)} \left\{ \frac{D_c(q)D_p(q)}{D_c(q)D_p(q) + N_c(q)N_p(q)} \right\} \tag{5.16}$$

- ***Control signal calculation:***

From Figure(5.3),

$$u(q) = C(q)e_y(q) \tag{5.17}$$

By substituting Eq.(5.7) and (5.16) into (5.17),

$$u(q) = \frac{N_c(q)}{D_c(q)} * \frac{N_r(q)}{D_r(q)} \left\{ \frac{D_c(q)D_p(q)}{D_c(q)D_p(q) + N_c(q)N_p(q)} \right\} \tag{5.18}$$

$$\therefore \quad u(q) = \frac{N_r(q)}{D_r(q)} \left\{ \frac{N_c(q)D_p(q)}{D_c(q)D_p(q) + N_c(q)N_p(q)} \right\} \tag{5.19}$$

- ***Output signal calculation:***

From Figure(5.3),

$$y(q) = P(q)u(q) = \frac{N_p(q)}{D_p(q)} u(q) \tag{5.20}$$

By substituting Eq.(5.19) into (5.20)

$$\therefore \quad y(q) = \frac{N_r(q)}{D_r(q)} \left\{ \frac{N_c(q)N_p(q)}{N_c(q)N_p(q) + D_c(q)D_p(q)} \right\} \tag{5.21}$$

Substituting Eq.(5.13) and (5.14) into (5.21) results in:

$$y(q) = \frac{N_r(q)}{D_r(q)} \left\{ \frac{D_{ps}(q)Q_n(q)N_p(q)}{\tilde{D}_r(q)\, D_p(q)Q_d(q) + D_{ps}(q)Q_n(q)N_p(q)} \right\} \tag{5.22}$$

From previous equation, Eq.(5.22):

- ☒ Desired response $y(q)$, will be equal reference signal $\frac{N_r(q)}{D_r(q)}$ (Well Tracking) if and only if

$$\{D_{ps}(q)Q_n(q)N_p(q)\} = \{\widetilde{D}_r(q)\, D_p(q)Q_d(q) + D_{ps}(q)Q_n(q)N_p(q)\} \qquad (5.23)$$

- Moreover, fastest response occurs when all z-domain poles are at the origin, this correspond to unity denominator in q-domain. This property happens if and only if the following condition satisfies,

$$\{\widetilde{D}_r(q)\, D_p(q)Q_d(q) + D_{ps}(q)Q_n(q)N_p(q)\} = 1 \qquad (5.24)$$

substituting Eq.(5.24) into (5.23) implies that:

$$D_{ps}(q)Q_n(q)N_p(q) \;= 1 \qquad (5.25)$$

and hence, $\widetilde{D}_r(q)\, D_p(q)Q_d(q) = 0 \qquad (5.26)$

- ***Output signal examination:***

Substituting Eq.(5.24) and (5.25) into (5.22) implies that:

$$\therefore \quad y(q) = \{1\}\frac{N_r(q)}{D_r(q)} \qquad (5.27)$$

- ***Tracking error signal examination:***

Substituting Eq.(5.24) and (5.25) into (5.16) implies that:

$$e_y(q) \;= \frac{N_r(q)}{D_r(q)}\left\{\frac{0}{1}\right\} = 0 \qquad (5.28)$$

- ***Control signal examination:***

Substituting Eq.(5.24) and (5.25) into (5.19) implies that:

$$\therefore u(q) = \frac{N_r(q)}{D_r(q)}\left\{\frac{N_c(q)D_p(q)}{1}\right\} = \frac{N_r(q)D_{ps}(q)\, Q_n(q)D_p(q)}{D_r(q)} = \frac{N_r(q)D_p(q)\, Q_n(q)}{D_r(q)} \qquad (5.29)$$

5.5.2 Deadbeat Disturbance Rejection Problem

For Ripple-free deadbeat disturbance rejection problem, the following assumptions are also assumed to solve it.

❖ **Assumptions 5.2:**

i. Assume that $D_p(q) = D_{pu}(q), i.e. D_{ps}(q) = 1$. This assumption will increase the settling time one sample for every additional pole in $D_{pu}(q)$ but increases the accuracy of approach results
ii. $D_w(q), D_{pu}(q)$ are coprime
iii. $N_w(q), D_w(q)$ are coprime.

❖ **Theorem [27]: The Ripple-free Deadbeat Tracking and Disturbance rejection problem**

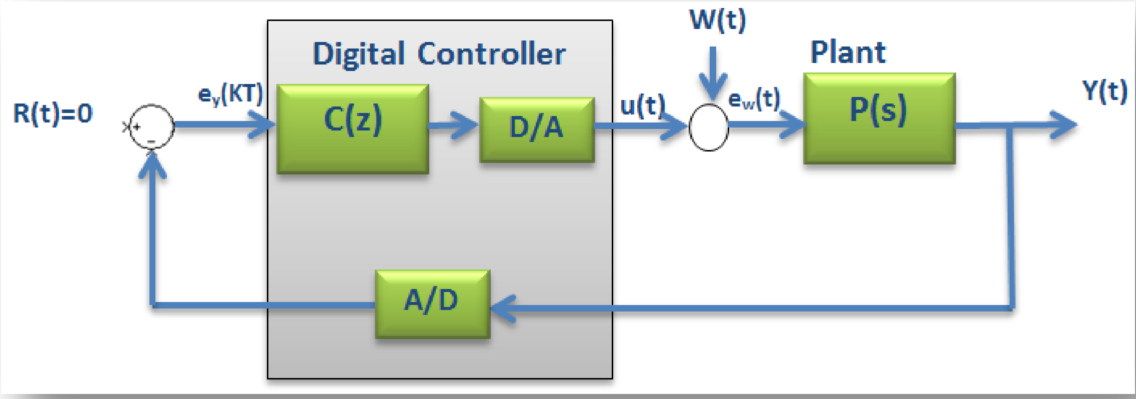


Figure (5.4): Ripple-free deadbeat disturbance rejection model

The Ripple-free deadbeat tracking and disturbance rejection problem has a solution if and only if the polynomial $N_p(q)$ *is coprime with both* $D_r(q)$ *and* $D_w(q)$. *Moreover, all solutions are of the form*

$$N_c(q) = D_{ps}(q)\, Q_n(q) = Q_n(q) \tag{5.30}$$

$$D_c(q) = \tilde{X}(q)\tilde{D}_w(q)\tilde{D}_r(q)\, Q_d(q) \tag{5.31}$$

where $Q_n(q)$ *and* $Q_d(q)$ *are polynomial solutions of the Diophantine equation*

$$\Rightarrow \quad [N_p(q)]\, Q_n(q) + [\tilde{X}(q)\tilde{D}_w(q)\tilde{D}_r(q)D_p(q)]\, Q_d(q) = 1 \tag{5.32}$$

such that $Q_d(0) = 1$

In order to provide ripple-free tracking and disturbance rejection, tracking property is proved previously. Disturbance property is proved alone by applying superposition principle, assuming zero reference signal $(r = 0)$, we should examine control error, control and output signals in q-domain as following:

Control error signal should eliminate disturbance after finite number of steps. Output signal also must go to zero after N steps at most. To obtain deadbeat (fastest) response, all poles should be at the origin, that's the denominator of output in q-domain equals one. The following equations illustrate these properties:

- ***Control error signal calculation:***

From Eq.(5.6) and Figure(5.4),

$$e_w(q) = w(q) + u(q)$$
$$= w(q) - C(q)\, y(q)$$
$$= w(q) - C(q)\, P(q) e_w(q)$$
$$= \frac{w(q)}{1 + C(q) * P(q)}$$

$$= \left\{\frac{D_c(q)D_p(q)}{D_c(q)D_p(q) + N_c(q)N_p(q)}\right\} \frac{N_w(q)}{D_w(q)} \tag{5.33}$$

$$\therefore\ e_w(q) = \left\{\frac{\tilde{D}_r(q)D_p(q)\, Q_d(q)}{\tilde{D}_w(q)\tilde{D}_r(q)D_p(q)\, Q_d(q) + N_p(q)Q_n(q)}\right\} N_w(q) \tag{5.34}$$

where $\tilde{X}(q) = 1$

- ***Control signal calculation:***

From Figure(5.4),

$$u(q) = C(q)y(q) = C(q)P(q)e_w(q) \tag{5.35}$$

By substituting Eq.(5.33) into (5.35),

$$u(q) = \frac{N_c(q)}{D_c(q)} * \frac{N_p(q)}{D_p(q)} \left\{\frac{D_c(q)D_p(q)}{D_c(q)D_p(q) + N_c(q)N_p(q)}\right\} \frac{N_w(q)}{D_w(q)} \tag{5.36}$$

By substituting Eq.(5.30), (5.31) into (5.36),

$$\therefore\ u(q) = \left\{\frac{N_p(q)\, Q_n(q)}{\tilde{D}_w(q)\tilde{D}_r(q)D_p(q)\, Q_d(q) + N_p(q)\, Q_n(q)}\right\} \frac{N_w(q)}{D_w(q)} \tag{5.37}$$

- ***Output signal calculation:***

From Figure(5.4),

$$y(q) = P(q)e_w(q) = \frac{N_p(q)}{D_p(q)} e_w(q) \tag{5.38}$$

By substituting Eq.(5.34) into (5.38)

$$\therefore\ y(q) = \frac{N_p(q)}{D_p(q)} \left\{\frac{\tilde{X}(q)\tilde{D}_w(q)\tilde{D}_r(q)D_p(q)\, Q_d(q)}{\tilde{D}_w(q)\tilde{D}_r(q)D_p(q)\, Q_d(q) + N_p(q)Q_n(q)}\right\} \frac{N_w(q)}{D_w(q)} \tag{5.39}$$

$$y(q) = N_w(q)N_p(q)\left\{\frac{\widetilde{D}_r(q)\,Q_d(q)}{\widetilde{D}_w(q)\widetilde{D}_r(q)D_p(q)\,Q_d(q) + N_p(q)Q_n(q)}\right\} \qquad (5.40)$$

- ☒ From Eq.(5.34), Control error signal equal *zero* signal (Well Disturbance rejection) if and only if

$$\widetilde{D}_r(q)D_p(q)\,Q_d(q) = 0 \qquad (5.41)$$

- ☒ Moreover, from Eq.(5.40), fastest response occurs when all z-domain poles are at the origin (s-domain poles are at infinity since $z = e^{-sT}$), this correspond to unity denominator in q-domain. This property happens if and only if the following condition satisfies,

$$\{\widetilde{D}_w(q)\widetilde{D}_r(q)D_p(q)\,Q_d(q) + N_p(q)Q_n(q)\} = 1 \qquad (5.42)$$

substituting Eq.(5.41) into (5.42) implies that:

$$Q_n(q)N_p(q) \;= 1 \qquad (5.43)$$

- ***Error signal examination:***

Substituting Eq. (5.42) into (5.34) implies that:

$$\therefore\;\; e_w(q) = \left\{\frac{\widetilde{D}_r(q)D_p(q)\,Q_d(q)}{\widetilde{D}_w(q)\widetilde{D}_r(q)D_p(q)\,Q_d(q) + N_p(q)Q_n(q)}\right\} N_w(q)$$

$$= \left\{\frac{\widetilde{D}_r(q)D_p(q)\,Q_d(q)}{1}\right\} N_w(q)$$

$$= \;\widetilde{D}_r(q)D_p(q)\,Q_d(q)N_w(q) \qquad (5.44)$$

Previous equation, Eq. (5.44), implies that the error signal function is a polynomial which vanishes to zero after finite steps, which equal the order of this polynomial.

- ***Output signal examination:***

Substituting Eq. (5.42) into (5.40) implies that:

$$y(q) = N_w(q)N_p(q)\left\{\frac{\widetilde{D}_r(q)\,Q_d(q)}{1}\right\}$$

$$y(q) = N_w(q)N_p(q)\widetilde{D}_r(q)\,Q_d(q) \qquad (5.45)$$

A polynomial approaches to zero after finite steps

Combining results obtained from previous manipulations, the system will to settle, reject disturbance and eliminate the error within the smallest time.

5.6 Solving Diophantine Equation

At this moment, we can find the controller transfer function by solving Diophantine equation, i.e. compute the unknowns, $Q_n(q)$ and $Q_d(q)$ in Eq. (5.32) systemically as following steps:

1. For simplicity, rewrite Eq. (5.32) as

$$\alpha(q)\, Q_n(q) + \beta(q)\, Q_d(q) - 1 = 0 \qquad (5.46)$$

   where

$$\alpha(q) \stackrel{\text{def}}{=} N_p(q)$$

   and

$$\beta(q) \stackrel{\text{def}}{=} \tilde{X}(q)\widetilde{D}_w(q)\widetilde{D}_r(q)D_p(q)$$

   It is clear that in order to compute $\alpha(q)$ and $\beta(q)$, we need to factorize plant, reference and disturbance transfer functions as stated in section (5.4). Then, we can easily evaluate $order(\alpha(q)), order(\beta(q))$.

2. Evaluate the minimum order of $Q_n(q)$ and $Q_d(q)$ from the relation:

$$Minimum\ order\ of\ Q_n(q) \text{ and } Q_d(q) = max\{order(\alpha(q)), order(\beta(q))\} - 1 \qquad (5.47)$$

$$Minimum\ Settling\ time\ N_{min} = order\ of\ Q_n(q) + order\ of\ Q_d(q) + 1$$

3. Assume two polynomials $Q_n(q)$ and $Q_d(q)$ with unknown coefficients, and of degree computed from Eq. (5.47).

4. After processing the previous three steps, now Diophantine equation; Eq. (5.46) has at most $\{2 * (result\ of\ Eq.(5.47) + 1)\}$ unknowns which evaluated from equating terms of similar degrees in the two sides of Eq. (5.46). As the number of unknown coefficients increases, manipulation complexity is also increases. Hence, M-file function is programmed to simplify computations as shown in Appendix A.

5. Substituting $Q_n(q), Q_d(q)$ in Eq. (5.30) and Eq.(5.31) respectively, obtain the Ripple-free deadbeat controller is found via Eq. (5.7).

5.7 Optimizing the Solution of Diophantine Equation

Noticing Diophantine equation, it has an infinite number of solutions that all provide an internally stabilizing controller. Hence, an optimization is needed. Such parameters to be optimized are as settling time, control signal magnitude. A tradeoff is needed between them. Control power shouldn't be ignored in real application, hence if the infinity norm of control signal is larger than acceptable value, we must compute reliable one.

The importance of the sampling period selection in the deadbeat controller design is clear. In many real life problem, the sampled closed-loop step responses have intersample oscillations cannot be avoided completely with previous approach. To avoid intersample oscillations, we maintain the control variable restricted or minimized, but this is achieved at the expense of obtaining minimum settling time. This section is designated for this option [25].

To accomplish this purpose, we examine control signal equation, Eq. (5.29),

$$u(q) = \frac{N_r(q)D_p(q)\ \boldsymbol{Q_n(q)}}{D_r(q)} \tag{5.29}$$

The control signal will depend only on denominator of plant, numerator and denominator of the reference signal, and the obtained polynomial $Q_n(q)$, so to minimize the absolute value of control signal; $u(q)$, it can be done by minimizing the numerator, $Q_n(q)$, by evaluating another polynomial instead of previous $Q_n(q)$. The following procedure illustrates this task. Let us called the previous $Q_n(q)$ by $Q_{nmin}(q)$,

Adding and subtracting the term $\{\alpha(q)\beta(q)V(q)\}$ from Eq.(5.46), where $V(q)$: *is the polynomial optimization vector*, results in:

$$\alpha(q)Q_n(q) + \beta(q)\ Q_d(q) + \ \alpha(q)\beta(q)V(q) - \ \alpha(q)\beta(q)V(q) - 1 = 0$$

$$\alpha(q)\underbrace{\{\ Q_{nmin}(q) - \beta(q)V(q)\}}_{Q_{nnew}(q)} + \beta(q)\underbrace{\{\ Q_{dmin}(q) + \alpha(q)V(q)\}}_{Q_{dnew}(q)} - 1 = 0 \tag{5.48}$$

$$\therefore Q_{nnew}(q) \stackrel{\text{def}}{=} \ Q_{nmin}(q) - \beta(q)V(q) \tag{5.49}$$

$$, Q_{dnew}(q) \stackrel{\text{def}}{=} \ Q_{dmin}(q) + \alpha(q)V(q) \tag{5.50}$$

And hence by substituting Eq.(5.49) into Eq.(5.29),

$$\therefore\ u(q) = \frac{N_r(q)D_p(q) * \boldsymbol{\{Q_{nmin}(q) - \beta(q)V(q)\}}}{D_r(q)} \tag{5.51}$$

It's obvious that minimizing the term $\{Q_{nmin}(q) - \beta(q)V(q)\}$ will also minimizes control signal magnitude, $u(q)$. One approach and the simplest way to do this, is by equating it by zero, that's

$$Q_{nmin}(q) - \beta(q)V(q) = 0 \tag{5.52}$$

where $Q_{nmin}(q)$ has known parameters from previous section; $\beta(q)$ also has known parameters results from plant, reference and disturbance factorization; and $V(q)$ is the unknown polynomial which should be computed. This is done by following steps:

1. Assume optional *order* for the polynomial $V(q)$. Then $V(q)$ now is a polynomial of (*order*+1) unknown coefficients.

2. Now Eq. (5.52) has only $V(q)$ unknowns which evaluated from equating terms of similar degrees in the two sides of Eq. (5.52). This step as the same as evaluating $Q_n(q)$ in Diophantine equation (5.46) and hence the developed function in section (5.6) can be used to find the unknowns.

3. Notice that the order of control signal is now increased and hence the settling time increases at the expense of minimum control magnitude.

4. The same procedure is followed to evaluate $Q_{dmin}(q)$

5. Substituting $Q_{nmin}(q), Q_{dmin}(q)$ in Eq. (5.48), Diophantine equation is evaluated to obtain the controller, Eq. (5.7).

CHAPTER 6   SIMULATION AND RESULTS

This chapter demonstrates the design methods that have been developed in the two previous chapters. MATLAB is used to perform the calculations, Simulink Design Optimization™ is used to perform PID optimization and SIMULINK for virtual simulation. The M-file codes that are used in obtaining the solutions of the examples are given in appendix A. Section 6.1 discusses designing PID controller for Magnetic ball levitation CE152 to solve stability and performance issues, and then evaluating second order estimation for the design. The results of section 6.1 is used in section 6.2 to evaluate a ripple-free deadbeat control for SIMULINK model of magnetic ball levitation CE152 with presence of time-delay and disturbance. Finally, section 6.3 compares between the results of this thesis approach and the results of other previous studies such as Elaydi [27] and Elammassie [11] results.

6.1 Magnetic Ball Levitation CE152

In this section, second order estimation transfer function of magnetic ball levitation CE152 will be simulated.

Following the procedure covered in Chapter 4, Stability and performance problem of the magnetic ball levitation are solved using PID controller optimization, as shown in Figure (6.1), with $K_p = 1.3, K_i = 3.2, K_d = 0.2$

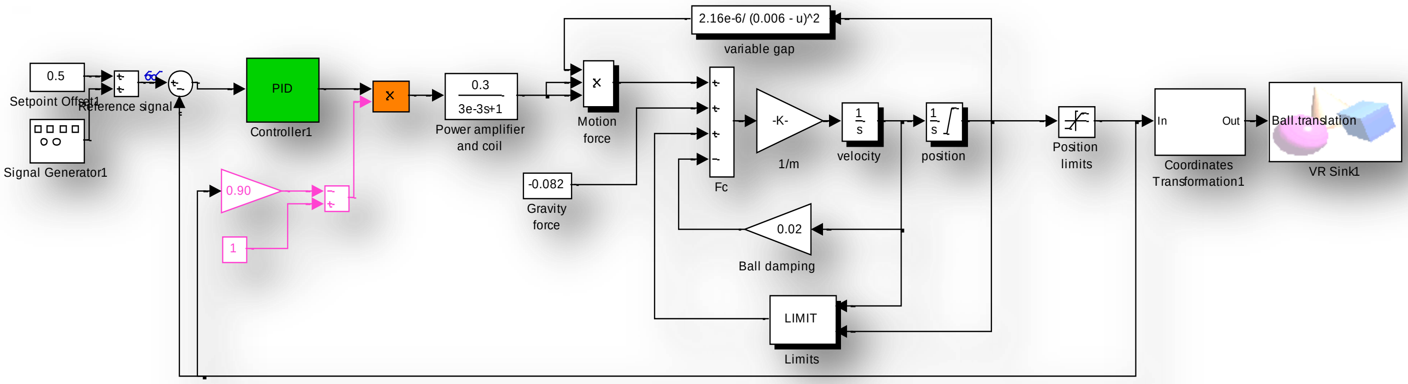


Figure (6.1): Model of maglev CE152 with PID controller

The step response of the magnetic Ball levitation with PID controller is shown in Figure (6.2).

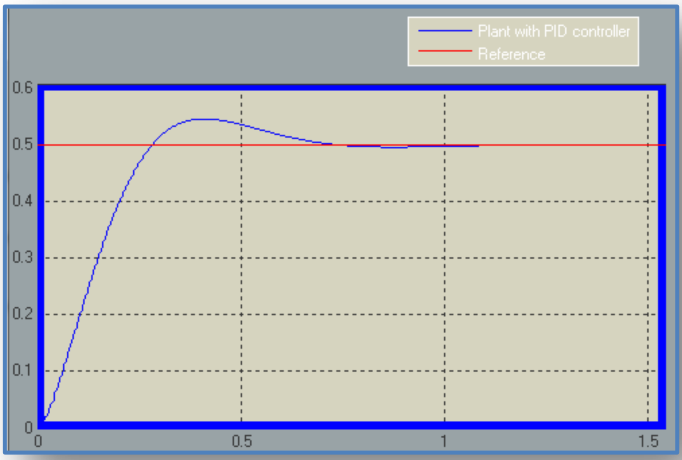


Figure (6.2): Step response of maglev CE152 with PID controller

Figure (6.2) shows that T_{rise}= 0.2 sec, $T_{settling}$= 0.7 sec, %Os = 5%**.**

As done in section (4.7), we have obtained a linear plant transfer function with certain specifications by estimating the step response of the magnetic Ball levitation with PID controller shown in Figure (6.2) as stated in Eq. (4.25), that's

$$\therefore \ Plant: \quad P(s) = \frac{12}{s^2 + 5s + 12}$$

with $T_{settling}$= 0.7 sec, %OS = 5% .To ensure that this system behaves in a similar manner as our original system, their step responses are shown in the same figure, Figure (6.3) and (6.4)

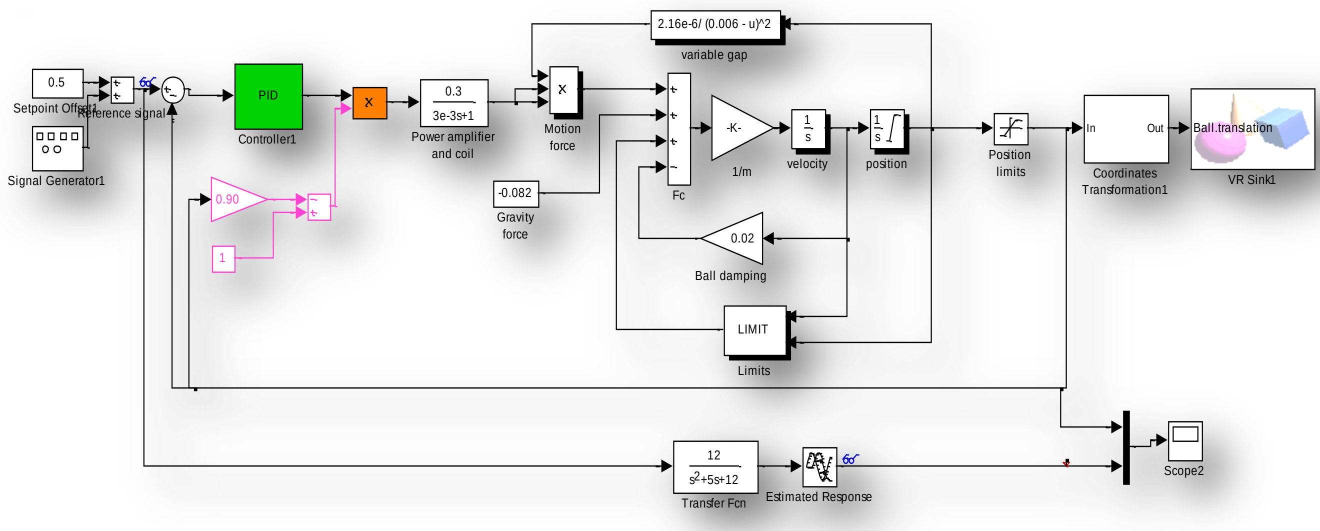


Figure (6.3): Maglev CE152 with PID controller and its second order estimation

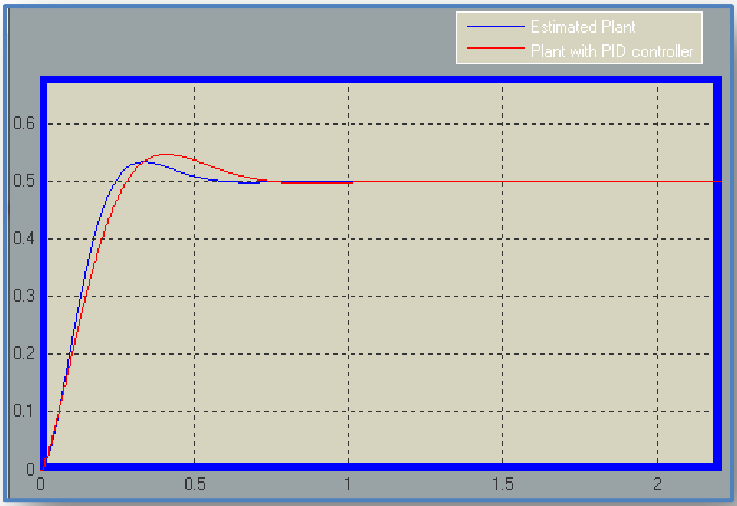


Figure (6.4): Step response of maglev CE152 with PID controller and the estimated model

Thus, the two responses are nearly the same, so this model will be used to design the deadbeat controller for the magnetic ball levitation.

6.2 Ripple-free Deadbeat Controller for Magnetic Ball Levitation CE152

Magnetic ball Levitation CE152 is chosen as a case study for nonlinear systems, since this apparatus is placed at IUG laboratory.

Consider this SISO continuous time plant with a time delay, T_d. Ripple-free deadbeat control is desired to track a reference signal, $r(t)$, in the presence of a disturbance signal, $w(t)$. It is desired to be used to meet certain specifications such as rising time (or settling time), steady state error and overshoot.
Many cases should be considered here, such as:

- Developing the controller without restrictions on control magnitude, the corresponding case is with restriction
- In both cases reference signal could have arbitrary form such as step, sinusoidal,…etc.
- In the presence of disturbance, controller performance should be examined developed by definition 5.1 and theorem 5.1 respectively
- Is the developed controller capable of time-delay handling?

6.2.1 Tracking problem without restriction on control signal magnitude

Case 1: $\boldsymbol{No\ disturbance\ w(t) = 0, No\ control\ magnitude\ constraints}$

Deadbeat controller for magnetic ball levitation with PID controller will be found by evaluating deadbeat controller for its estimated model, thus deadbeat controller will be found for $P(s)_{estimated}$ to follow reference signal $r(t) = 0.5\ (at\ center) \Rightarrow R(s) = \frac{0.5}{s}$. Consider also a time-delay, $T_d = 0.03\ sec$

Solution: Assume that the sampling time is $T = 0.1sec$

- Using MATLAB code, Convert plant transfer function from $s - domain$ to $z - domain$, then from $z - domain$ to $q - domain$, then factorize $P(q)$ to numerator and denominator, also find denominator roots as following

```
                   0.00043605 (z+0.9688)
P(z)= z^(-3) * -----------------------------
                    (z-0.9894) (z-0.9191)

         0.0004225 q^2 + 0.000436 q
P(q)= ---------------------------------------
           0.9094 q^2 - 1.909 q + 1

Np(q)= 0.0004225 q^2 + 0.000436 q

Dp(q)= 0.9094 q^2 - 1.909 q + 1

Dpq_root =
   1.0880
   1.0107
```

- Repeat step1 for reference transfer function,

```
       0.005
R(z)= ------
       (z-1)

       -0.005 q
R(q)= ----------
        q - 1

Nr(q) =0.0050q

Dr(q) = -q+1

Drq_root =  1
```

- Factorize plant and reference denominators as stated in Eq. (5.10) and (5.11), depending on denominator roots computed above, this results in:

$$\widetilde{W}(q) = 1, \widetilde{Y}(q) = 1$$

$$\Rightarrow \widetilde{D}_w(q) = D_w(q), \widetilde{D}_{pu}(q) = D_{pu}(q), \widetilde{D}_r(q) = D_r(q)$$

- Forming α(q) and β(q) as defined in Eq. (5.46), hence $\alpha(q) = N_p(q), \beta(q) = D_r(q)D_p(q)$, thus

beta = -0.9094 q^3+ 2.8179 q^2 -2.9085 q+ 1.0000

alpha = 0.0004225 q^2 + 0.000436 q

- Evaluate the minimum order of $Q_n(q)$ and $Q_d(q)$ from Eq. (5.47), thus

$$Minimum\ order\ of\ Q_n(q) \text{ and } Q_d(q) = max\{3, 2\} - 1 = 2$$

- Defining two polynomials $Q_n(q)$ and $Q_d(q)$, each with order equal 2 and total unknown coefficients equal 6.

$Q_n(q)$ = a q^2 + b q + c

$Q_d(q)$ = d q^2 + e q + f

- By substituting $\alpha(q), \beta(q), Q_n(q)$ and $Q_d(q)$ into Diophantine equation Eq. (5.46)

$$\therefore (0.0004225\, q^2 + 0.000436\, q)(a\, q^2 + b\, q + c) + (-0.9094q^3 + 2.8179 - -2.9085\, q + 1.0000)(d\, q^2 + e\, q + f) - 1 = 0$$

$$\therefore(-0.9094d)q^5 + (0.0004a + 2.8179d - 0.9094e)q^4 + (0.0004a + 0.0004b - 2.9085d + 2.8179e - 0.9094f)q^3 + (0.0004b + 0.0004c + d - 2.9085e + 2.8179f)q^2 + (0.0004c + e - 2.9085f)q + (f) = 0$$

Now solve unknowns by making all coefficients of previous equation equal zero as following:

$$\begin{bmatrix} 0 & 0 & 0 & -0.9094 & 0 & 0 \\ 0.0004 & 0 & 0 & 2.8179 & -0.90940 & 0 \\ 0.0004 & 0.0004 & 0 & -2.9085 & 2.8179 & -0.9094 \\ 0 & 0.0004 & 0.0004 & 1 & -2.9085 & 2.8179 \\ 0 & 0 & 0.0004 & 0 & 1 & -2.9085 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ d \\ e \\ f \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

By using MATLAB function (*pinv*) and the developed code to compute the product, we obtain:

$$\Rightarrow \begin{bmatrix} a \\ b \\ c \\ d \\ e \\ f \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & -0.9094 & 0 & 0 \\ 0.0004 & 0 & 0 & 2.8179 & -0.90940 & 0 \\ 0.0004 & 0.0004 & 0 & -2.9085 & 2.8179 & -0.9094 \\ 0 & 0.0004 & 0.0004 & 1 & -2.9085 & 2.8179 \\ 0 & 0 & 0.0004 & 0 & 1 & -2.9085 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}^{\dagger} \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

$\therefore Q_n(q) =$ 1.825 q^2 − 5.386 q + 4.726

$\therefore Q_d(q) =$ 0.8478 q + 1

$$\Rightarrow Minimum\ Settling\ time\ N_{min} = 2 + 1 + 1 = 4\ steps$$

- Substituting $Q_n(q), Q_d(q)$ in Eq. (5.30) and Eq.(5.31) respectively to obtain the Ripple-free deadbeat controller Eq. (5.7).

$$\therefore N_c(q) = Q_n(q),\ D_c(q) = D_w(q) D_r(q)\ Q_d(q)$$

$\therefore N_c(q) =$ 1.825 q^2 − 5.386 q + 4.726

$\therefore D_c(q) = -0.8478\ q^2 - 0.1522q + 1.0000$

Hence,

$$C(z) = \frac{4.726z^2 - 5.386z + 1.825}{z^2 - 0.1522z - 0.8478}$$

SIMULINK model for the plant for case 1 and case 2 is shown in Figure (6.5). Step response of the system with and without the controller is shown in Figure (6.6). Time response specifications are summarized in Table (6.1).

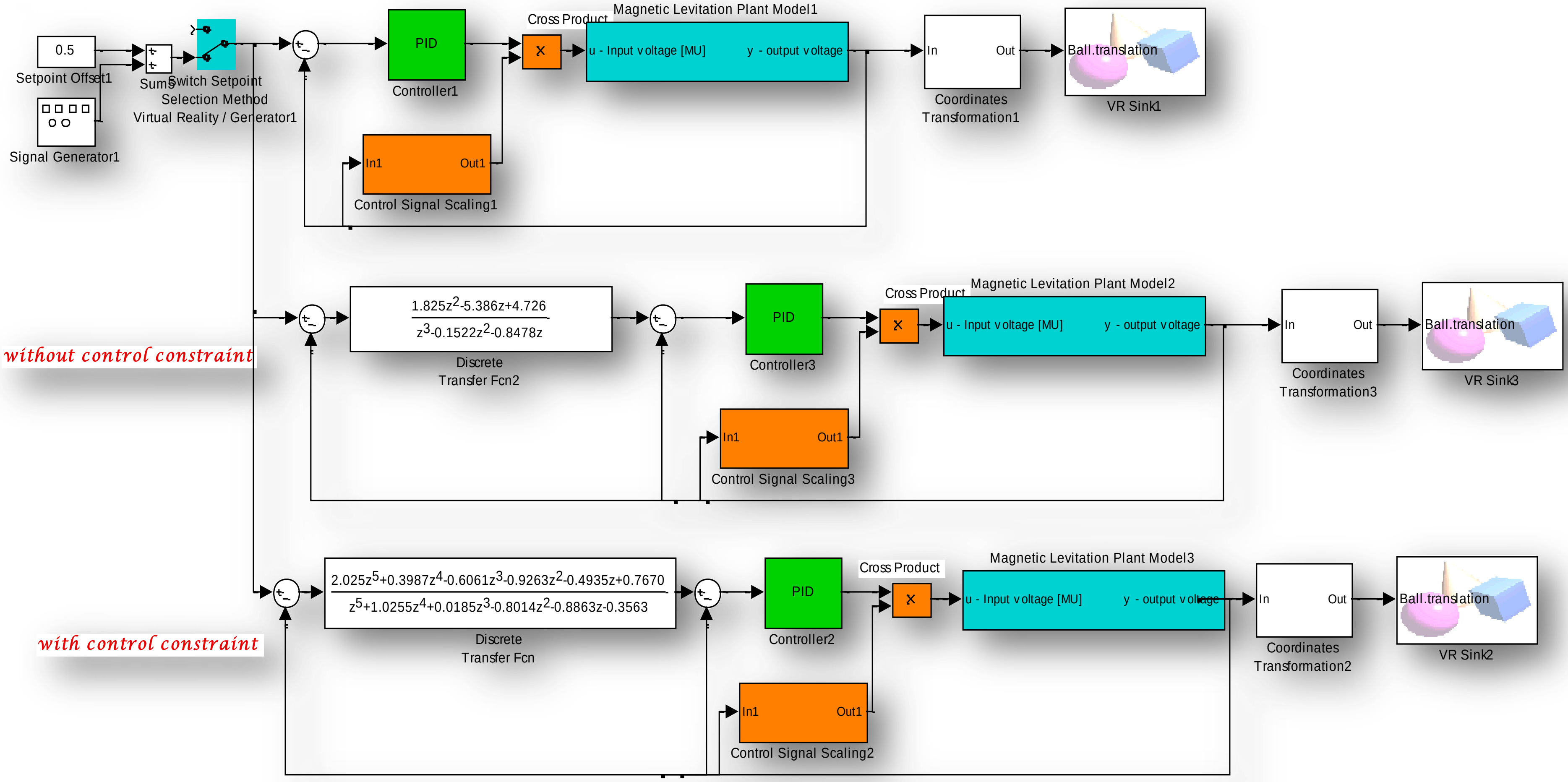


Figure (6.5): Maglev CE152 with PID, without and with deadbeat controller (Case 1 and Case2)

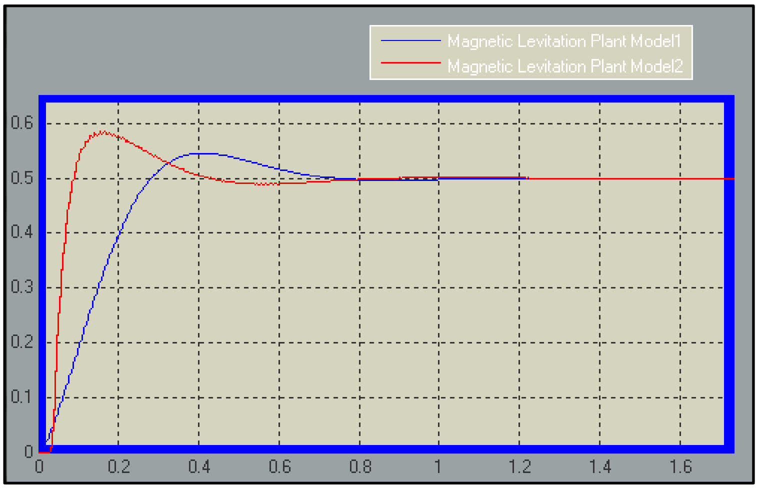


Figure (6.6): Step response of Maglev CE152 with PID, without and with deadbeat controller (Case 1)

Table (6.1): Time response specification summarization of Figure (6.6)

Figure (6.6)	**Without controller (Blue)**	**With controller, Case 1 (Red)**
Overshoot	5%	15%
Rising time	0.2	0.15
N_{min}	-	4 steps
Sampling time	0.1	0.1
Delay time	0	0.03
Settling time	0.7	0.43

From Figure (6.6) and Table (6.1), a clear minimization for the settling time is achieved, but surly this on the expense of larger system overshoot. There is a delay of 0.03, hence, approximately, the settling time is after 0.03+(4*0.1) $\simeq$ 0.43 sec.

6.2.2 Tracking problem with restriction on control signal magnitude

Case 2: $\boldsymbol{No\ disturbance\ w(t) = 0, with\ control\ magnitude\ constraints}$

Following the procedure developed in section 5.7, and repeating previous steps, we obtain the following results:

- Suppose that the length of the vector $V(q)\ is\ L = 3$, i.e. settling time will increase by three steps, $\Rightarrow N_{min} = 4 + 2 = 6\ steps$

$$,thus\ \ V(q) = \ 0.8434\ q^2\ +\ 2.0709\ q\ +\ 2.7009, \qquad using\ M - file\ directly$$

⇒ Hence,

$$C(z) = \frac{2.025z^5 + 0.3987z^4 - 0.6061z^3 - 0.9263z^2 - 0.4935z + 0.7670}{z^5 + 1.0255z^4 + 0.0185z^3 - 0.8014z^2 - 0.8863z - 0.3563}$$

SIMULINK model of the plant for case 2 is shown in Figure (6.5). Step response of the system with and without the controller is shown in Figure (6.7). Time response specifications are summarized in Table (6.2).

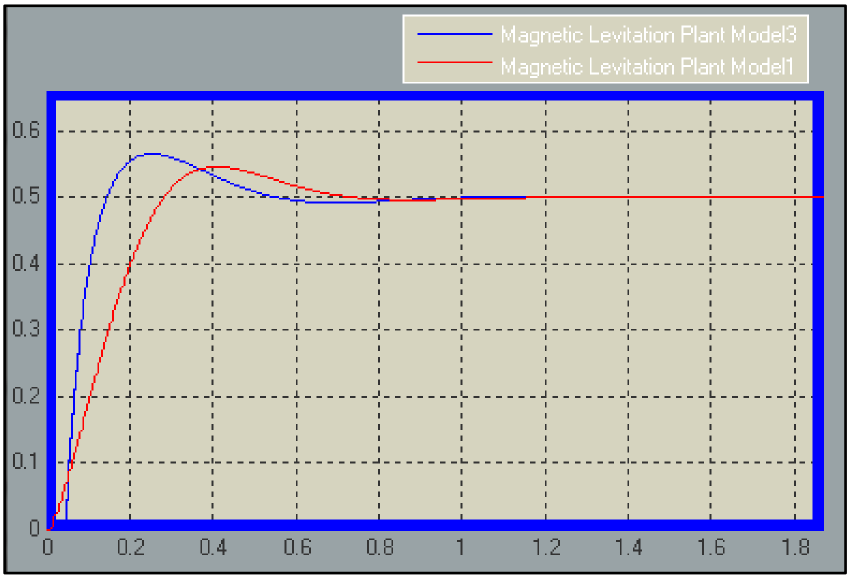


Figure (6.7): Step response of Maglev CE152 with PID, with and without deadbeat controller(Case 2)

Table (6.2): Time response specification summarization of Figure (6.7)

Figure (6.7)	**Without controller (Red)**	**With controller, Case 2 (Blue)**
Overshoot	5%	12%
Rising time	0.2	0.15
N_{min}	-	6 steps
Sampling time	0.1	0.1
Delay time	0	0.01
Settling time	0.7	0.61

From Figure (6.7) and Table (6.2), the control signal magnitude (and thus the system overshoot) is minimized, but this occurs at the expense of settling time. Approximately, settling time is after 0.01+(6*0.1) ≃ 0.61 sec.

To compare between the performance of the system without deadbeat controller and with two deadbeat controllers (Case1 and Case2), their responses are plotted simultaneously at Figure (6.8), and time response specifications are summarized in Table (6.3).

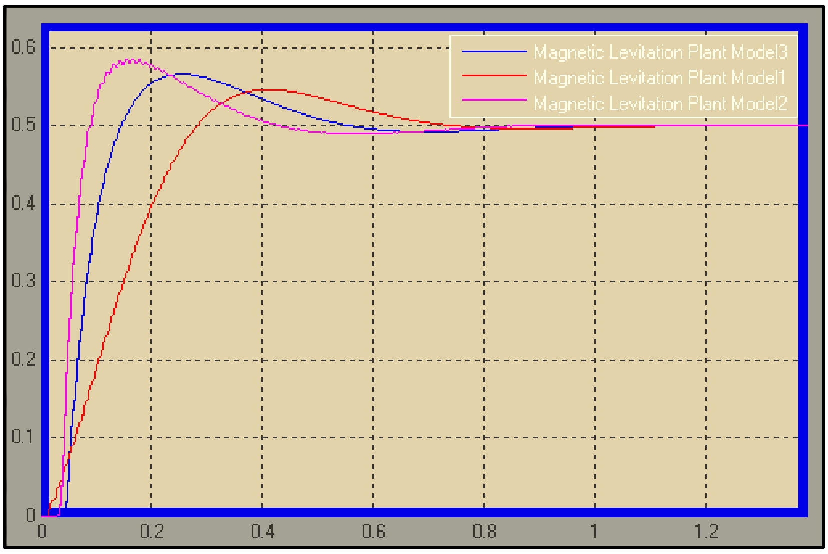


Figure (6.8): Step response of Maglev CE152 with PID, without and with deadbeat controller (Case1, Case2)

Table (6.3): Time response specification summarization of Figure (6.8)

Figure (6.8)	**Without Deadbeat controller (Red)**	**With Deadbeat controller, Case 1 (Pink)**	**With Deadbeat controller, Case 2 (Blue)**
Overshoot	5%	15%	12%
Rising time	0.2	0.15	0.15
N_{min}	-	4 steps	6 steps
Sampling time	0.1	0.1	0.1
Delay time	0	0.03	0.01
Settling time	0.7	0.43	0.61

6.2.3 Significance of Results (6.1):

Thus we conclude from Figure (6.8) and Table (6.3) that the developed two controllers for a plant with certain overshoot and settling time result in the following improvements:

1) After applying the PID controller, the time response is behaved but a still need to minimize the settling time is required. Also, note that after applying this controller, a time delay in the response is happened which added to the total resulting settling time.

2) After applying the PID controller, and the Ripple-free Deadbeat controller and without any optimization on control signal amplitude, Case1, we really obtain a clear minimization of the settling time at the expense of having higher overshoot.

3) After applying the PID controller, and the Ripple-free Deadbeat controller but with optimization for the control signal amplitude, Case2, we really obtain a clear minimization on the overshoot (resulting from minimization of the control signal magnitude) but surly this is at the expense of having longer settling step time over than previous case that results from the added optimization vector $v(q)$. The settling step time is increased by the order of $v(q)$.

4) The controller is capable of time delay handling.

6.2.4 Disturbance rejection problem without restriction on control signal magnitude

Case 3: Without input, With disturbance $w(t) = 0.1\, sin(2t)$, Without Control signal optimization

Repeating the same steps done in case 1 as following:

- Disturbance model and factorization:

```
         0.00099667 (z+1)
W(z)= -----------------------
         (z^2 - 1.96z + 1)

        0.0009967 q^2 + 0.0009967 q
W(q)= ------------------------------------
           q^2 - 1.96 q + 1

Nw(q) = 0.0009967 q^2 + 0.0009967 q

Dw(q) = q^2 - 1.96 q + 1

Dwq_root =
  0.9801 + 0.1987i
  0.9801 - 0.1987i
```

- $\therefore\ \widetilde{W}(q) = 1, \widetilde{X}(q) = 1, \widetilde{Y}(q) = 1, \tilde{Z}(q) = 1$
  $\Rightarrow\ \widetilde{D}_w(q) = D_w(q),\ \widetilde{D}_{pu}(q) = D_{pu}(q), \widetilde{D}_r(q) = D_r(q)$
  $\therefore N_c(q) = Q_n(q),\ D_c(q) = D_w(q)D_r(q)\, Q_d(q)$

- Forming α(q) and β(q) where $\alpha(q) = N_p(q), \beta(q) = D_r(q)D_w(q)D_p(q)$, thus

> beta = -0.6065 q^5 +3.3084 q^4 -7.2742 q^3 + 8.0454 q^2 -4.4732 q+ 1.0000
>
> alpha = 0.0429 q^2 + 0.0506 q

- Evaluating the minimum order of $Q_n(q)$ and $Q_d(q)$ from Eq. (5.47), thus

$$\textit{Minimum order of } Q_n(q) \text{ and } Q_d(q) = max\{5, 2\} - 1 = 4$$

- Defining two polynomials $Q_n(q)$ and $Q_d(q)$ with order equal '4' and '10' unknown coefficients

> $Q_n(q)$ = a q^4 + b q^3 + c q^2 + d q + e
>
> $Q_d(q)$ = f q^4 + g q^3 + h q^2 + k q + l

- By substituting $\alpha(q), \beta(q), Q_n(q)$ and $Q_d(q)$ into Diophantine equation Eq. (5.46) using MATLAB code, we obtain

$$\Rightarrow \begin{bmatrix} a \\ b \\ c \\ d \\ e \\ f \\ g \\ h \\ k \\ l \end{bmatrix} = \begin{bmatrix} 11.6562 \\ -63.2025 \\ 137.2813 \\ -147.1052 \\ 72.0658 \\ 0 \\ 0 \\ 0 \\ 0.8235 \\ 1 \end{bmatrix}$$

and hence

> $Q_n(q)$ = 11.6562 q^4 -63.2025 q^3 + 137.2813 q^2 - 147.1052 q + 72.0658
>
> $Q_d(q)$ = 0.8235 q^2 + 1

$$\Rightarrow Minimum\ Settling\ time\ N_{min} = 3 + 1 + 1 = 5\ steps$$

and

$$\therefore N_c(q) = 11.66\,q^4 - 63.2\,q^3 + 137.3\,q^2 - 147.1\,q + 72.07$$

$$\therefore D_c(q) = 0.8235\,q^4 - 1.438\,q^3 - 0.5223\,q^2 + 2.137\,q - 1$$

Hence,

$$C(q) = \frac{11.66\,q^4 - 63.2\,q^3 + 137.3\,q^2 - 147.1\,q + 72.07}{0.8235\,q^4 - 1.438\,q^3 - 0.5223\,q^2 + 2.137\,q - 1}$$

SIMULINK model of the plant for case 3 is shown in Figure (6.9). Step response of the system with and without the controller is shown in Figure (6.10). Time response specifications are summarized in Table (6.4).

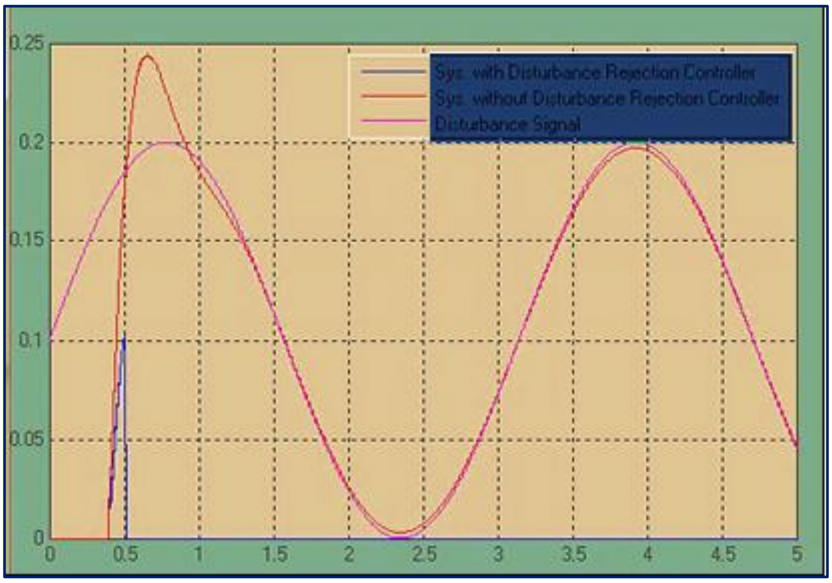


Figure (6.10): Step response of Maglev CE152 with PID, without and with deadbeat disturbance rejection controller (Case 3)

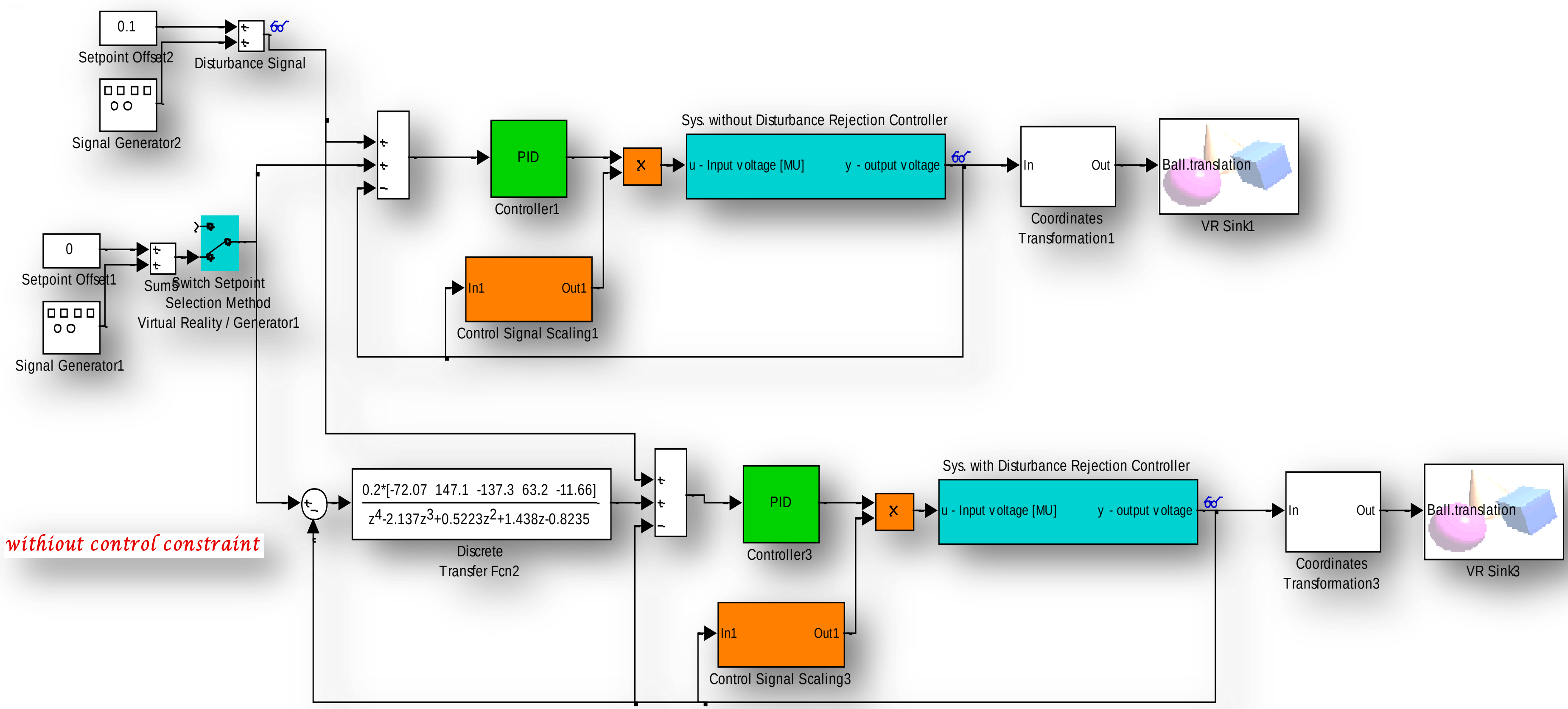


Figure (6.9): Maglev CE152 with PID, without and with deadbeat disturbance rejection controller (Case 3)

(Note: numerator of the controller is modified by a scaling factor (0.2) to limit saturations)

Table (6.4): Time response specification summarization of Figure (6.10)

Figure (6.10)	Without Disturbance rejection deadbeat controller (Red)	With Deadbeat controller, Case 1 (Blue)
Overshoot	25%	10%
Rising time	0.5	0.4
N_{min}	-	5 steps
Sampling time	0.1	0.1
Delay time	0	0.03
Settling time	1	0.5

From Figure (6.10) and Table (6.4), a clear elimination of pure disturbance problem is achieved within nearly 0.03+(5*0.1)=0.53 ≈ 0.5 sec and hence time is deadbeat also.

6.2.5 Disturbance Rejection problem with restriction on control signal magnitude

Case 4: Without input, With disturbance $w(t) = 0.1\, sin(2t)$, With Control signal optimization

Following the procedure developed in section 5.7, and repeating steps done in section 6.2.2, we can obtain the controller with restriction on control signal magnitude. Since this method applies longer settling time, it is not applied on this system in order to not increasing the settling time more than 0.5 sec. Also, here, overshoot is not important as in tracking problem so it hasn't been simulated. But if one want to apply it, it is straightforward as in section 6.2.2 as stated and surly overshoot will decreased at the expense of settling time.

6.2.6 Significance of Results (6.2):

Thus we conclude from Figure (6.10) and Table (6.4) that the developed disturbance rejection deadbeat controller results in the following improvements:

Purely disturbance rejection problem is achieved in minimum settling time. A problem of input tracking and disturbance rejection problem are simultaneously solved as in section 6.3.2

6.3 Improving Deadbeat Controller for Linear Systems

In this section, results obtained via this thesis approach are compared with that obtained in some references such as [11] and [27].

6.3.1 Example (1)

The system that is given in [27] and is shown in equation (6.1) is considered. It is desired to track a unit step signal in the presence of a sinusoidal signal as disturbance, $w(t)$. It is desired to control this plant with ripple-free deadbeat manner. The plant is

$$P(s) = \frac{50e^{-0.14s}}{s^2 + s + 1} \tag{6.1}$$

and the disturbance is

$$w(t) = cos(12.556t)$$

It is desired to minimize the overshoot.

Repeating the same steps done in previous section we have the controller :

$$C(q) = \frac{1.139\ q\^{}4\ -\ 4.527\ q\^{}3\ +\ 8.138\ q\^{}2\ -\ 9.116\ q\ +6.469}{-0.04618\ q^4 - \ 0.3546\ q^3 + 0.6209\ q^2 - 0.6495\ q + \ 1}$$

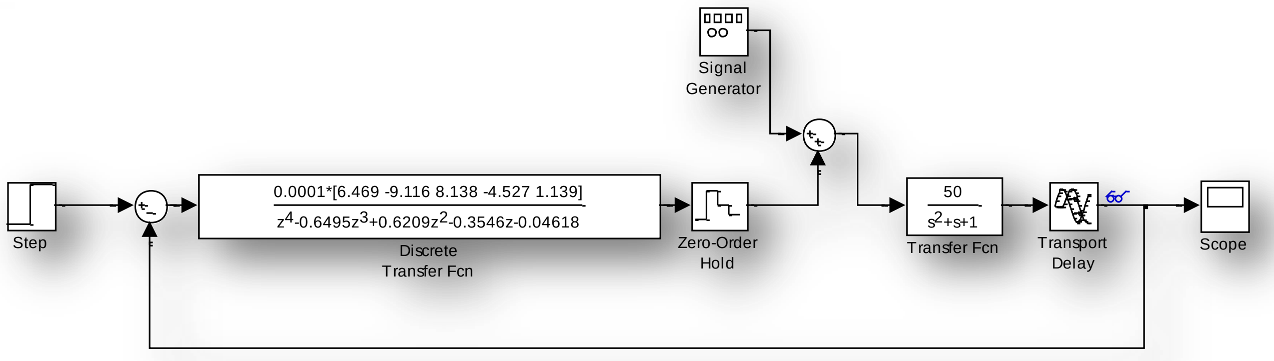


Figure (6.11): System in Example (1) with deadbeat controller

and a step response shown in Figure (6.12):

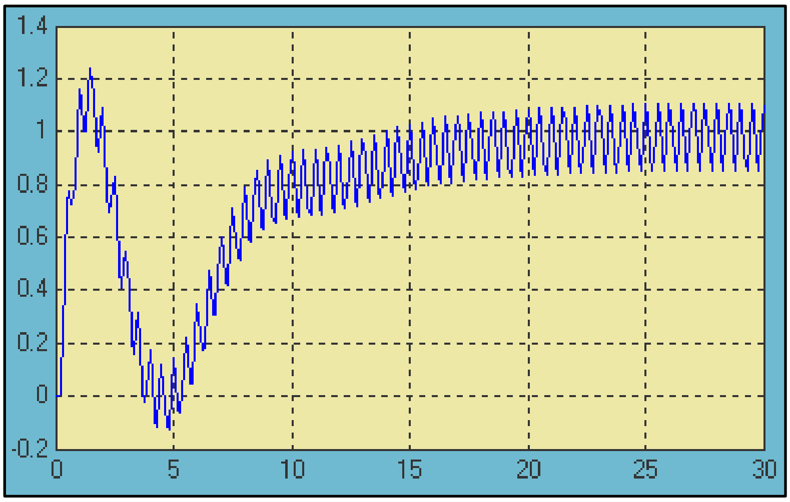


Figure (6.12): Step response of the system under disturbance and time-delay after applying deadbeat tracking and disturbance rejection controller

Comparing this figure with that in [27, page 119]:

Table(6.5): Comparison between Elaydi [27], this thesis results

	Overshoot	Controller order	Settling time
Elaydi [27]	20%	5	7 sec
This thesis	10%	4	7 sec

we notice that disturbance rejection is achieved here with overshoot=10% is less than 20% obtained in [27] at the same settling time = 7 sec. Also, this response is achieved by a controller of order 4 which is less than order 5 used in [27]. Thus, the deadbeat problem in this thesis is achieved with less both overshoot and controller order structure. Moreover, this approach can deal nonlinear systems.

6.3.2 Example (2)

The system that is given in [11] and is shown in equation (6.2) is considered.

$$P(s) = \frac{200e^{-0.2s}}{s^2 - 2s + 2} \tag{6.2}$$

It is desired to track the sinusoid signal

$$r(t) = \sin(2t + pi/5)$$

with ripple-free deadbeat manner and minimizes the infinity norm of control energy. Let the sampling time = 0.1 sec and vector $v(q)$ length is 3.

Repeating the steps done in previous section, then we have:

- Decreasing control signal using equation (5.51), we obtain

$$v(q) = 0.2608\ q^2 + 0.7736\ q + 1.1675,$$

$$using\ M - file\ directly\ given\ in\ AppendixA$$

where the vector obtained in [11, page 64] is

$$v(q) = 0.3082\ q^2 + 0.6752\ q + 1.0847,$$

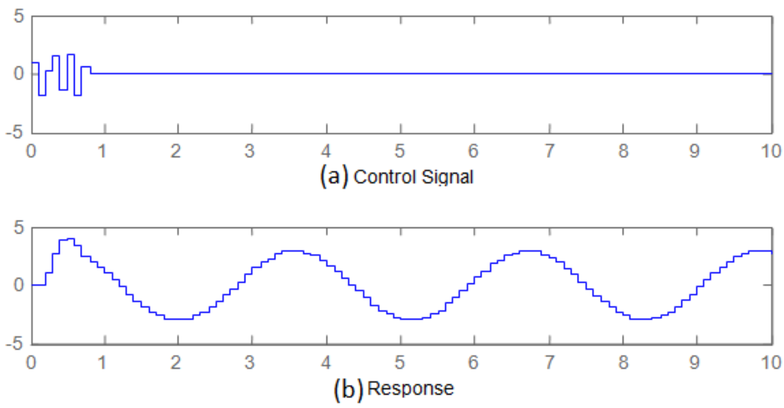


Figure (6.13): Time response of vector v(q) which was produced by thesis approach.

Figure (6.13) is a result of M-file code given in appendix A. Figure (6.13) shows control, and output signals which settled after 10 sampling times with overshoot = 38.68%, settling time = 10*0.1=1sec.

Results of step responses to the system using Elammassie approach [11, page 65] and the developed approach in this thesis are summarized in Table (6.5)

Table(6.6): Comparison between Elammassei [11], this thesis results

Vector	Norm(u,inf)	Norm(E,inf)	Norm(E,2)	Overshoot	Settling time
Elammassei app.	1.8764	2.4491	4.0026	35.79%	1 sec
My app.	0.4018	0.6545	0.8761	38.68%	1 sec

For the same settling time, the controller design based on thesis approach have minimum infinity norm of control signal, minimum 2-norm, and minimum infinity-norm of the error signal.

CHAPTER 7 CONCLUSION

Rapid advances in control theory have led to a rapid development in digital control systems. Practically, every aspect of our day-to-day activities is affected by some type of digital control. Deadbeat controller is a type of digital controllers, which offers the fastest settling time. Therefore, deadbeat controller ensures that the error sequence vanishes at the sampling instants after a finite time. Moreover, real plants and processes are typically nonlinear which are mostly affected by external disturbances and time-delays.

There is a wide area of applications for control systems, where nonlinear phenomena cannot be avoided. Therefore, control of nonlinear systems is an important area of control engineering. Many processes is required to settle in minimum period, which is called in control system deadbeat systems Time-delay systems are unavoidable in many control systems. Hence, a strong need to deal with them is exist. A desire to develop new disturbance rejection methodologies that are simpler and more robust than existing methodologies is needed. This issue as a whole was never dealt with before; thus, this thesis proposed a complete solution using an effective methodology.

Thus, in this thesis, the following contribution was made in the field of ripple-free deadbeat controllers for nonlinear systems. Constrained controllers are obtained that satisfy the properties of internal stability, performance, tracking of arbitrary references and attenuation of known disturbances. The thesis approach can also handle systems with time delays.

This thesis presented an approach for the ripple-free deadbeat controller for nonlinear system in order to track random input signal in presence of time delays and known disturbance signals via tuning PID controller and solving Diophantine equation. The control approach has many aims such as: designing Ripple-free deadbeat controller that achieve good transient response in presence of time-delay for nonlinear system which makes the output signal y to track any random input signal with zero steady-state error in the smallest number of sampling instants; tuning PID controller; solving time-delayed nonlinear control problem under disturbance using ripple-free deadbeat controller; studying the effect of time-delays and disturbances on the stability and performance; and finally realizing the developed controller using MATLAB software Toolbox.

The control methodology combined two controllers to control nonlinear systems, PID controller and Diophantine equation which depended on Polynomial approach. These two feedbacks were used to stabilize the nonlinear system and to make the response of the nonlinear system closely tracking the reference signal. The Diophantine equation depends on the internal model principle that was utilized to track a reference and reject disturbance dynamics for a plant with time delays. The results shown that, the response of nonlinear system tracked the reference signal with zero steady state after very small rising time.

A systematic method for the design of a controller that can track arbitrary reference input signal and reject a known disturbance was described. The models of the reference and disturbance signals were included in the Diophantine design equation.
The system was assumed to be nonlinear plant. In order to handle nonlinearities and saturations, a PID control tuning using Simulink Design Optimization™ software was applied, the resulting well behaved response was estimated as a linear second order transfer function which dealt directly with Diophantine polynomial approach. The approach can handle systems with time delays, where the time delay is not an integer multiple of the sampling time. The approach also dealt with disturbances, where the disturbance signal is assumed to be of a known linear form. A method for designing ripple-free deadbeat controller was developed. The controller allowed for minimizing the settling time and control signal magnitude.

The proposed controller was applied using SIMULINK model of magnetic ball levitation CE 152 as a case study for nonlinear systems, simulation results shown that the controller performed fine with simulated plant. Simulation results showed that the output signal exactly tracked the input signal and reject the disturbance signal in short settling time. The time domain specification for the output signal, control signal, and error signal were computed and satisfied the requirement and constraints. A time delay was also presented with simulation and was solved by using deadbeat controller based on solving Diophantine equation parameters.

Further directions for research include considering deadbeat nonlinear control method without using separated linearizing controller. Also, the general tracking problem for systems with multiple time delays, where the delays could show up anywhere in the system could be investigated. Moreover, the effect of changing working points, changing the sampling time, and changing the frequency of input and disturbance signals can be also studied.

REFERENCES

[1] H. Ahn, Y. Chen, and K. Moore (2007), "Iterative Learning Control: Brief Survey and Categorization", IEEE Systems, Man, and Cybernetics Society, VOL. 37, NO. 6.

[2] F. Orfano, Why are Control Systems Used?, Edited & published by Rebecca Scudder, accessed on Dec. 25, 2011, online: http://www.brighthub.com/engineering/electrical/articles/72133.aspx

[3] Wikipedia, the free encyclopedia, Digital control, accessed on Dec. 25, 2011, online: http://en.wikipedia.org/wiki/Digital_control.

[4] D. Nešić, (1996), "Dead-Beat Control for Polynomial Systems", Ph.D. thesis, Australian National University.

[5] Dictionary and Encyclopedia Directory, Control system – Definition, accessed on Dec. 25, 2011, online: http://www.wordiq.com/definition/Control_system

[6] W. Sakallah, (2009), "Comparative Study for Controller Design of Time-delay Systems", M.S. thesis, Department of Electrical Engineering. Islamic University Of Gaza, Gaza-strip, Palestine.

[7] S. Weidong, (2002), "Nonlinear Disturbance Rejection", Ph.D. thesis, Department of Systems Engineering. Australian National University, Australia.

[8] H. Elaydi and R. A. Paz, (1998), "Optimal Ripple-Free Deadbeat Controllers for Systems with Time Delays", Proceedings of the American Control Conference Philadelphia, Pennsylvania.

[9] L. Jetto and S. Longhi, (1999), "Parameterized solution of the deadbeat ripple-free control problem for multi-rate sampled-data systems", Proceedings of the 38th Conference on Decision & Control Phoenix, Arizona USA.

[10] F. Albatsh, (2009), "Multirate Ripple-Free Deadbeat Control", M.S. thesis, Department of Electrical Engineering. Islamic University Of Gaza, Gaza-strip, Palestine

[11] M. El Ammassie, (2011), "Deadbeat for nonlinear multirate systems via LMI", M.S. thesis, Department of Electrical Engineering. Islamic University Of Gaza, Gaza-strip, Palestine.

[12] D. Shona, (2003), "Design and Analysis of Nonlinear Sampled-Data Control Systems", Ph.D. thesis, Department of Electrical and Electronic Engineering, The University of Melbourne.

[13] Wikipedia, the free encyclopedia, Nonlinear system, accessed on Jan 26, 2013, online: http://en.wikipedia.org/wiki/ Nonlinear_system

[14] Wikipedia, the free encyclopedia, Scientific Techniques of Levitation, accessed on Jan 24, 2013, online: http://en.wikipedia.org/wiki/Levitation.

[15] Marcio S. de Queiroz and Darren M. Dawson (1996), " Nonlinear Control of Active Magnetic Bearings: A Backstepping Approach," IEEE transactions on control system technology, Vol. 4, NO 5.

[16] HUMUSOFT, (1996-2008), CE 152 Magnetic Levitation Model, Technical Manual, Czech Republic.

[17] F. Gazdoš , P. Dostál, and R. Pelikán (2009), "Polynomial approach to control system design for a magnetic levitation system," Cybernetic Letters: Informatics, Cybernetics, and Robotics, PP. 1-19

[18] Magnetic Levitation Model (CE 152), Advanced-models-laboratory, laboratories VSB -Technical University of Ostrava, accessed on Jan 26, 2013, online: http://www.352.vsb.cz/english/laboratories/advanced-models-laboratory/levitation

[19] K. Ali, M. Abdelati, M. Hussein (Jan.2010), "Modeling, Identification and Control of a Magnetic Levitation CE152", Al-Aqsa University Journal (Natural Sciences Series), Vol.14, No.1, PP 42-68.

[20] Wikipedia, the free encyclopedia, PID controller, accessed on Jan 4, 2013, online: http://en.wikipedia.org/wiki/PID_controller

[21] Math Works, Inc. (2012). Simulink® Design Optimization™, Getting Started Guide, Available: http://www.mathworks.com

[22] P. Olver, Nonlinear Systems, Available: http://www.math.umn.edu/~olver/am_/ne.pdf

[23] H. Khalil, Nonlinear Systems, 3rd Edition , Prentice-Hall, Englewood Cliffs, NJ.

[24] Ang, K.H., Chong, G.C.Y., and Li, Y. (2005). PID control system analysis, design, and technology, IEEE Trans Control Systems Tech, 13(4), pp.559-576, Available: http://eprints.gla.ac.uk/3817/

[25] M. Fadali, Digital Control Engineering. Burlington, MA: Academic Press, 2009

[26] H. Sirisena. (1985). Ripple-free Deadbeat Control of SISO Discrete Systems. IEEE Transactions on Automatic Control, AC-30(2):168-170

[27] H. Elaydi, "Complex controller design using linear matrix inequalities" Ph.D. thesis, Electrical Department, New Mexico state University, Las Cruces, New Mexico, 1998

APPENDIX A MATLAB M-FILE CODES

a. MATLAB main code

This code uses many user defined functions which attached on the thesis CD

```
clear all
close all
clc
T=0.01;
%% Plant information
Ps=tf([12],[1 5 12],'iodelay',0.025)    %% plant in s-domain
c2d(Ps,T)
Pz=zpk(c2d(Ps,T))
[Npz Dpz]=tfdata(Pz,'v')    %% plant in z-domain
Pq  = SQs( Ps,T )
[Npq Dpq]=tfdata(Pq,'v')     %% plant in q-domain
Dpq_root=roots(Dpq)
Dpq=removeZeros(Dpq)
OrderP=length(Dpq)-1         %% plant order
%% Reference information
Rs=tf([0.5],[1 0])           %% Reference signal in s-domain
Rs=zpk(Rs)
Rz=zpk(c2d(Rs,T))
[Nrz Drz]=tfdata(Rz,'v')     %% Reference signal in z-domain
Rq=SQs(Rs,T)
[Nrq Drq]=tfdata(Rq,'v')     %% Reference signal in q-domain
Drq_root=roots(Drq)
%% Symbolized plant and reference signals by variable x instead of q
Dp_sym=poly2sym(Dpq);
Dr_sym=poly2sym(Drq);
%% Solving Diophantine by computing polynomials Qn, Qd
DrqdotDpq_sym=Dp_sym*Dr_sym;
DrqdotDpq=sym2poly(DrqdotDpq_sym);
alpha=Npq
beta=DrqdotDpq
alpha_sym=poly2sym(alpha);
beta_sym=poly2sym(beta)
[MOQnQd]=Mini(alpha,beta)
[Qn_sym Qd_sym]=QnQdsym(MOQnQd)
DE=(collect(simplify((alpha_sym *Qn_sym)+(beta_sym*Qd_sym))-1))
[COs CO0 order]=COPs(DE)
[VB ConsK]=getVBs(COs,CO0,order)
QnQd=pinv(VB)*ConsK
Qn=QnQd(1:1:MOQnQd+1)
Qd=QnQd(MOQnQd+2:1:2*MOQnQd+2)
Qn_q=tf(Qn',1,T,'variable','q')
```

```
Qd_q=tf(Qd',1,T,'variable','q')
%% Check if the Diophantine equations are OK or  not
[Qn,DQn]=tfdata(Qn_q,'v')
[Qd,DQd]=tfdata(Qd_q,'v')
DC=DioChs(alpha,Qn,beta,Qd)
%% Compute controller C(q)= Nc(q)/Dc(q)
Qdq_sym=poly2sym(Qd)
Dc_sym=Qdq_sym*Dr_sym
Dcq=sym2poly(Dc_sym)
Cq=tf(Qn,Dcq,T,'variable','q')
%% Minimizing Control Signal by Computing vector V
L=3
V=QnQdsym(L-1)
Qn_sym=poly2sym(Qn)
Qd_sym=poly2sym(Qd)
Qn_new_sym=Qn_sym -(beta_sym*V)
Qd_new_sym=Qd_sym+(alpha_sym*V)
%Qn_new
[COs21 CO021 order21]=COPs(Qn_new_sym)
[VB21 ConsK21]=getVBs(COs21,CO021,order21)
V=pinv(VB21)*ConsK21
V_sym=poly2sym(V')
Qn_new_sym=Qn_sym -(beta_sym*V_sym)
Qn_new_final=sym2poly(Qn_new_sym)
%Qd_new
Qd_new_sym=Qd_sym +(alpha_sym*V_sym)
Qd_new_final=sym2poly(Qd_new_sym)
%% Compute controller C(q)= Nc(q)/Dc(q)
Dc_new_sym=Qd_new_sym*Dr_sym
Dcq_new=sym2poly(Dc_new_sym)
Cq=tf(Qn_new_final,Dcq_new,T,'variable','q')
%%
%% ---------------- WITH DISTURBANCES-----------------
%% Disturbance information
Ws=tf([0.2],[1 0 4])          %% Reference signal in s-domain
Ws=zpk(Ws)
Wz=zpk(c2d(Ws,T))
[Nwz Dwz]=tfdata(Wz,'v')    %% Reference signal in z-domain
Wq=SQs(Ws,T)
[Nwq Dwq]=tfdata(Wq,'v')    %% Reference signal in q-domain
Dwq_root=roots(Dwq)
%% Symbolized plant, disturbance and reference signals by variable x
instead of q
Dp_sym=poly2sym(Dpq);
Dr_sym=poly2sym(Drq);
Dw_sym=poly2sym(Dwq);
%% plant, disturbance and reference Factorization
Wtelda=1;
Xtelda=1;
Ytelda=1;
Ztelda=1;
DrTelda=Drq;
DwTelda=Dwq;
DpTelda=Dpq;
DrTelda_sym=poly2sym(DrTelda);
DwTelda_sym=poly2sym(DwTelda);
DpTelda_sym=poly2sym(DpTelda);
%% Forming alpha_q and beta_q
beta_sym=DrTelda_sym*DwTelda_sym*DpTelda_sym;
```

```
beta=sym2poly(beta_sym)
alpha=Npq
alpha_sym=poly2sym(alpha);
%% Solving Diophantine by computing polynomials Qn, Qd
[MOQnQd]=Mini(alpha,beta)
[Qn_sym Qd_sym]=QnQdsym(MOQnQd)
DE=(collect(simplify((alpha_sym *Qn_sym)+(beta_sym*Qd_sym))-1))
[COs CO0 order]=COPs(DE)
[VB ConsK]=getVBs(COs,CO0,order)
QnQd=pinv(VB)*ConsK
Qn=QnQd(1:1:MOQnQd+1)
Qd=QnQd(MOQnQd+2:1:2*MOQnQd+2)
Qn_q=tf(Qn',1,T,'variable','q')
Qd_q=tf(Qd',1,T,'variable','q')
%% Check if the Diophantine equations are OK or  not
[Qn,DQn]=tfdata(Qn_q,'v')
[Qd,DQd]=tfdata(Qd_q,'v')
DC=DioChs(alpha,Qn,beta,Qd)
%% Compute controller C(q)= Nc(q)/Dc(q)
Qdq_sym=poly2sym(Qd)
Dc_sym=Qdq_sym*Dr_sym*Dw_sym
Dcq=sym2poly(Dc_sym)
Cq=tf(Qn,Dcq,T,'variable','q')
%%
%% ---------------- WITH DISTURBANCES, CONTROL CONSTRAINT------------
%% Minimizing Control Signal by Computing vector V
L=3
[x Qn_new]=performance2s(Qn_q,Pq,Rq,Ps,L)
x=x(end:-1:1);
Qn_q_New=tf(Qn_new(end:-1:1),1,T,'variable','q')
%-------------------------------
% Compute controller C(q)= Nc(q)/Dc(q)
Cq_new=tf(Qn_new(end:-1:1),Dcq,T,'variable','q')
```

b. Developing control and response signals in the case of tracking problem via M-file

```
%% Generate time domain input
Amplitude=0.5;
t=0:0.001:5;
u=Amplitude*ones(1,length(t));
 %% Without Control Signal Minimizing
[sys1,Ter1,Tur1,Tyr1]= RFsyssthesis(Cq,Pq)
%--------------------
figure(1)        % Step Response of minimum order solutions
subplot(3,1,1)
lsim(Tyr1,u,t)
title(['Response , when R is step with Amplitude = 0.5']);

C=lsim(Tyr1,u,t);
CC=norm(C,inf);
OS=((CC-Amplitude)*100)/Amplitude
```

```
% --------------
C=lsim(Tur1,u,t);
subplot(3,1,2)
lsim(Tur1,u,t)
title(['Control Signal , when R is step with Amplitude = 0.5']);

Inf_Norm_U_N1_NO=norm(C,inf)
%---------------
C=lsim(Ter1,u,t);
subplot(3,1,3)
lsim(Ter1,u,t)
title(['Error Signal , when R is step with Amplitude = 0.5'])

Inf_Norm_Error_N1_NO=norm(C,inf)
Sec_Norm_Error_N1_NO=norm(C,2)
```

APPENDIX B USED SOFTWARE

a. Windows XP

b. MATLAB R2010a (matrix laboratory) is a numerical computing environment and fourth-generation programming language. Developed by MathWorks, MATLAB allows matrix manipulations, plotting of functions and data, implementation of algorithms, creation of user interfaces, and interfacing with programs written in other languages, including C, C++, and FORTRAN. It is used to simulate thesis M-file codes

c. SIMULINK 7.4, developed by MathWorks, is a commercial tool for modeling, simulating and analyzing multi-domain dynamic systems. Its primary interface is a graphical block diagramming tool and a customizable set of block libraries. It offers tight integration with the rest of the MATLAB environment and can either drive MATLAB or be scripted from it. SIMULINK is widely used in control theory and digital signal processing for multi-domain simulation and design. It is used to simulate thesis models and obtaining virtual reality simulation.