

University of Southern Queensland
Faculty of Engineering and Surveying

**ROBUST DEADBEAT CONTROL OF THE NON-
LINEAR MAGNETIC LEVITATION SYSTEM USING
ADAPTIVE CONTROL TECHNIQUES**

A dissertation submitted by

Adam James Clarke

in fulfilment of the requirements of

Courses ENG4111 and 4112 Research Project

Towards the degree of

Bachelor of Electrical Engineering

Submitted: October, 2012

Abstract

The modern push toward faster and more efficient transportation systems has caused many new techniques to be developed. One of which is a Magnetically Levitated train, or Maglev, where an electro-magnetic control system is used to hover and propel a train at high speeds and efficiency. This has been proven to work with the 'Transrapid' Maglev train in Germany, and the 'Shanghai Maglev Train' in China. The project studied a similar dynamic control system which is applied to operate these trains. This system provides many advantageous control characteristics in a control system sense.

An ECP (Educational Control Products) Magnetic Levitation plant was used in the development of this control system. The robust deadbeat controller was designed and simulated firstly, in order to prove its viability. The initial simulations found that a switch controller was required, due to the nonlinear dynamics of the plant. The designed switch controller was further investigated in simulations, and the results showed that the controller was viable and effective, and could be applied to the plant.

While applying the designed switch controller to the plant, some real-world problems such as noise and control errors were encountered. Digital signal processing techniques were employed to remove these issues. In the final testing, results were obtained and evaluated from the designed controller and a traditional PID controller. These results showed that the objectives of the designed controller were well achieved, and its performance is consistently 60% better in settling time than that of a traditional PID controller. Other performance results include a 66% better disturbance rejection, and a 30% better bandwidth. Analysis showed that controller designed was a viable alternative to available controllers in the industry.

University of Southern Queensland
Faculty of Engineering and Surveying

ENG4111 Research Project Part 1 & ENG4112 Research Project Part 2
--

Limitations of Use

The Council of the University of Southern Queensland, its Faculty of Engineering and Surveying, and the staff of the University of Southern Queensland, do not accept any responsibility for the truth, accuracy or completeness of material contained within or associated with this dissertation.

Persons using all or any part of this material do so at their own risk, and not at the risk of the Council of the University of Southern Queensland, its Faculty of Engineering and Surveying or the staff of the University of Southern Queensland.

This dissertation reports an educational exercise and has no purpose or validity beyond this exercise. The sole purpose of the course pair entitled “Research Project” is to contribute to the overall education within the student's chosen degree program. This document, the associated hardware, software, drawings, and other material set out in the associated appendices should not be used for any other purpose: if they are so used, it is entirely at the risk of the user.



Professor Frank Bullen

Dean

Faculty of Engineering and Surveying

Certification

I certify that the ideas, designs and experimental work, results, analyses and conclusions set out in this dissertation are entirely my own effort, except where otherwise indicated and acknowledged.

I further certify that the work is original and has not been previously submitted for assessment in any other course or institution, except where specifically stated.

Student Name: ADAM JAMES CLARKE

Student Number: 0050101263

Signature

Date

Acknowledgements

This project was supervised by Dr. Paul Wen. I would like to thank him for his support and technical help throughout the duration of the project. I would also like to extend a special thanks to Mr Shahab Abdulla for his guidance, without which I would not have been able to complete this project.

I would also like to thank my family and friends for their continuing support.

ADAM CLARKE

University of Southern Queensland

October 2012

Contents

Abstract	i
Certification	iii
Acknowledgements	iv
List of Figures	x
List of Tables	xiii
Chapter 1 Introduction	1
1.1. Introduction	1
1.2. System and Background	5
1.2.1. Control system perfection in the modern world.....	5
1.2.2. Magnetically Levitated Trains	6
1.2.3. ECP Model 730 Maglev Machine.....	7
1.2.4. Nonlinear control background.....	9
1.3. Nonlinear Control Systems	10
1.3.1. Design of Robust Deadbeat Controllers.....	11

1.3.2.	Adaptive solution to non-linear systems	12
1.3.3.	ECP Model 730 Maglev control system	13
1.4.	Assessment of Consequential Effects and Ethical Responsibilities	16
1.4.1.	Consequential Effects.....	16
1.4.2.	Ethical Responsibilities.....	17
1.5.	Methodologies, Resources and Timelines	17
1.5.1.	Task Execution Methodology	17
1.5.2.	Resource Analysis	18
1.5.3.	Time Usage Guidelines	21
Chapter 2	ECP Model 730 Maglev System.....	23
2.1.	ECP model 730 Maglev Machine	23
2.2.	Nonlinear Actuator (coil) Characteristics.....	26
2.3.	Linearization using Taylor' Series expansion	28
Chapter 3	Robust Deadbeat Control Technique.....	31
3.1.	Robust Deadbeat Control Theory	31
3.1.1.	Introduction to Robust Deadbeat Control	31
3.1.2.	Design Method	32
3.2.	Design procedure for the Maglev system.....	34
3.2.1.	Maglev design method using the linearized plant.....	34
3.2.2.	Matlab script design for fast deadbeat control calculations	35

3.3.	Simulation Analysis	36
3.3.1.	Plant Simulation	36
3.3.2.	Controller simulation	37
3.3.3.	Simulation examples	38
3.4.	Performance Optimisation using Numerical Methods	39
3.4.1.	Demonstration of the user-defined parameters	39
3.4.2.	Optimisation of the plant model using numerical methods	40
3.4.3.	Simulation Performance Results	40
Chapter 4	Adaptive Deadbeat Controller Design and Simulation	43
4.1.	Piecewise Adaptive Controller	43
4.2.	Piecewise Adaptive Controller Design.....	45
4.2.1.	Switches used for the adaptive controller	45
4.2.2.	Selection and optimisation for each controller counterpart	46
4.3.	Simulation Results and Performance Evaluation	46
4.3.1.	Step response for small excursions	46
4.3.2.	Step response for large excursions.....	49
Chapter 5	Deadbeat Controller Implementation	51
5.1.	Programming the Maglev System	51
5.1.1.	C programming language.....	51
5.1.2.	Programming the robust deadbeat controller	52

5.2.	Maglev Plant Initial Implementation Problems.....	55
5.2.1.	Sensor noise	55
5.2.2.	Vibration Issue	57
5.2.3.	Thermal compensation	58
5.3.	Results and Performance Optimisation	58
Chapter 6	Adaptive Controller Implementation.....	61
6.1.	Plant Programming for the Adaptive Controller	61
6.2.	Numerical Performance Optimisation for all Sections.....	62
6.3.	Boundary Performance Optimisation for Large Excursions	64
Chapter 7	Experiment Results and Evaluation.....	66
7.1.	Simulation and Plant Comparison.....	66
7.2.	PID performance analysis	69
7.2.1.	Performance results for small excursions	70
7.2.2.	Performance results for large excursions	71
7.3.	Performance Analysis of the Final Controller.....	71
7.3.1.	Small Magnet Displacements.....	72
7.3.2.	Large magnet displacements	73
7.3.3.	Disturbance effect on the robust system	75
7.3.4.	Frequency domain response	76
7.4.	Analysis of Results	79

CONTENTS	ix
7.5. Discussion of Results	81
7.6. Design Experience.....	82
Chapter 8 Conclusions and Further Work.....	84
8.1. Conclusions	84
8.2. Further Work	88
References	90
Appendix A Project Specification.....	92
Appendix B Matlab Code for Simulation	94
Appendix C C Code for the Maglev Plant	100
Appendix D Simulink Models	108
Appendix E Risk Assessment	112

List of Figures

Figure 1.1: Artist's impression of the landing of NASA's rover 'Curiosity' (NASA, 2012)	5
Figure 1.2: Cross-section of a Maglev train assembly (source: Lee & Kim 2006)	6
Figure 1.3: Representation of propulsion force generation (source: Lee & Kim 2006)	7
Figure 1.4: Maglev train (Brian Betts 2006)	7
Figure 1.5: Maglev plant	8
Figure 1.6: Maglev development system	8
Figure 1.7: Linearization about an operating point, A (Nise, N 2011)	11
Figure 1.8: Robust deadbeat controller and plant	12
Figure 1.9: Maglev response using feedback linearization technique (Liceaga, Hernandez and Amezquita (2009))	14
Figure 1.10: ECP Model 730 Magnetic Levitation Machine (Source: Educational Control Products 2002)	19
Figure 1.11: ECP Control System Software (Source: Educational Control Products 2002)	20

Figure 2.1: ECP Model 730 Representation (Source: Educational Control Products 2002)	24
Figure 2.2: Free-body diagram with force interactions (Parks, T 1999).....	24
Figure 2.3: Physical measurements obtained from the Maglev plant	26
Figure 2.4: Representation of the estimated curve against the experimental data.....	27
Figure 2.5: Taylor's linearization theorem representation, about point A (2cm).....	30
Figure 3.1: Deadbeat Control Technique (adapted from Dawes, Ng, Dorf & Tam 1994).	32
Figure 3.2: Robust deadbeat controller simulation model	37
Figure 3.3: Initial Simulation Results (settling point inset – 515ms)	39
Figure 3.4: Best Simulink response for a 2cm movement	41
Figure 3.5: Effect of moving to 3cm, for a linearization point of 2cm	42
Figure 4.1: Representation of a piecewise adaptive control technique	44
Figure 4.2: Switching technique for the deadbeat controller (representation only)...	45
Figure 4.3: Switched adaptive deadbeat controller, small movement response	48
Figure 4.4: magnification of the step response, 1cm movement.....	48
Figure 4.5: Simulation results for a large excursion	49
Figure 5.1: Representation of derivative and integral approximation errors	53
Figure 5.2: Key points used for C-code generation	54
Figure 5.3: Sensor noise, and FIR filtering response	57
Figure 5.4: Effect of moving away from linearization point	59

Figure 5.5: Linearization error from small movements	60
Figure 6.1: Comparison of single and switched deadbeat control results on the plant	64
Figure 6.2: Step response for large magnet movements	65
Figure 7.1: Comparison between the simulation and plant responses	67
Figure 7.2: Comparison of key control points	68
Figure 7.3: Comparison of key PID points (point-by-point related to the previous figure).....	69
Figure 7.4: PID Maglev plant controller, small step response.....	70
Figure 7.5: PID best response	71
Figure 7.6: Step response comparison for small magnet movements.....	72
Figure 7.7: Large step response comparison.....	74
Figure 7.8: Effect of a disturbance on the controllers.....	76
Figure 7.9: Frequency response	77
Figure 7.10: Gain plot vs. Frequency.....	78
Figure 7.11: Phase vs. Frequency for both controllers	79

List of Tables

Table 1.1: Milestones	22
Table 2.1: Definition of Maglev plant-specific variables	24
Table 3.1: Comparison of constants	40
Table 4.1: Optimised deadbeat controllers for the prescribed boundaries	46
Table 5.1: Code used to build the controller in C	54
Table 6.1: Boundary optimisation parameters	63
Table E.1: Risk Assessment Matrix.....	113
Table E.2: Risk assessment for uncontrolled movement of the Maglev magnets ...	114
Table E.3: Risk assessment for falling objects	115
Table E.4: Risk assessment for potential exposed electrical wires.....	115
Table E.5: Risk assessment for laser radiation from the machine	116

Chapter 1

Introduction

1.1. Introduction

Fast and efficient transportation of humans and goods has been the target of many research and development projects. Locomotives have been used in the transportation of goods and people for the last two centuries. In the time between the first concept design of a locomotive in 1804 (The Victorian Web, 2003) developments which aimed at increasing the effectiveness and efficiency include steam power, diesel motors and electrical supply systems. The most recent developments in the rail industry have been focused on ‘floating’ trains above the railway tracks. This is achieved by creating magnetic fields to provide both lift and directional forces to the train (Lee, H & Kim, K & Lee, J 2006). Propulsion is also achieved by creating magnetic fields in the same methods which are used in electric motors.

This magnetic levitation of the train reduces rolling friction on wheeled locomotives and also allows for better aerodynamic designs. By removing the friction between

the locomotive and the railway, much higher speeds and less energy lost to friction can be achieved. However as power is required to sustain the levitating field for the train, some argue that power is wasted levitating the train, where this motion is not actually performing any work, i.e. moving the train toward its destination.

A major obstacle which must be overcome is the drag force on the object (locomotive) which is proportional to the square of the velocity of the object (Benson, T 2011). This can be overcome by operating the locomotive in a vacuum, where wind resistance is null. Therefore, if magnetic levitation design techniques were used along with a vacuum to operate a locomotive, much higher speeds and better efficiencies could be obtained.

The design of magnetically levitated trains involves the alignment of different magnetic fields in a manner to achieve levitation, guidance and propulsion, within a small distance tolerance. For the Transrapid Maglev train in Germany, the gap between the train and the guidance coil is around 10cm (Lee & Kim 2006). Therefore the free play of the train is estimated to be around 1-2cm, which is considerably small. A control system must be designed which can keep the train within this tolerance distance between the rail guides, while operating the train in a safe manner. External factors which will change this height include variances in ground height, centripetal force from cornering and the effect of gravity on the train. Magnetic fields will also be cross-coupled between different magnetic systems.

It can be seen that for a train travelling at 500km/h (achievable by Maglev), keeping the train within a 2cm tolerance would require a precise control system. This control system must be able to read all the parameters and vary the coil currents in a short amount of time. The control system which reads all these factors must be able to calculate the correct current to supply into the guide coils. The goal of this is to ensure the train stays within the prescribed distances inside the guide rail while maintaining the safety of the passengers and cargo. Therefore there is a need to understand and apply control system law to the magnetic system in order to operate the train as required.

To understand the control system complexity, a simple magnetic levitation system has been designed by Educational Control Products (ECP). This is the ECP model

730 Maglev instrument, which is a development system based around the control of magnetic systems. This magnetic system has one degree of movement (vertical axis) and has the option of multiple inputs (coils) and multiple outputs (magnets). The control system has direct control of the current passed through the magnetic coils, to a very high precision. The control system also accesses readings from the laser positioning sensors, which provide a real-time reading of the position of the magnets in the plant.

The Maglev plant was designed for the development of students in control theory applications. It demonstrates the complexity of magnetic control systems, and outlines the importance of control theory to the precision control of magnetic levitation systems. Real world Maglev systems use similar control theory; therefore development on this control system will lead students into the world of complex control designs. The Maglev system has a real-time controller which executes a user defined control algorithm. Therefore any arbitrary control system can be designed and applied to the system. This is an ideal platform for the design of the control system in the scope of this project.

There is a nonlinear relationship governing the force and the distance between a magnet and a coil. A linear time-invariant (LTI) control system is one which has a linear relationship between the input of the system and the output of the system. Linear systems control theory can be applied to a system with purely linear devices. A system which does not have a direct input/output relationship such as this magnetic relationship is considered nonlinear, where linear systems control law cannot be applied (Nise, N 2011). This poses a major issue when a system must exhibit optimal control responses. A method which will control a system of this nature must be proposed.

In control theory, the most optimal control system design is a deadbeat controller. A deadbeat controller is defined as a control system which can reach steady state in the shortest time for a given input, i.e. the system is designed for optimal response. There are many methods which can be applied to effectively control a nonlinear system, including an adaptive controller, a fuzzy logic controller or a linear compensation controller. The system could also be assumed linear for small degrees

of motion. In any case, to apply a deadbeat controller, the system parameters must be realised. This can be done using the user manual supplied by ECP. Once the system parameters have been realised, the system control algorithm must be designed for deadbeat control. The system must also be simulated using MATLAB or Simulink. These are the broad objectives which are defined in Appendix A.

The resultant control system for this plant must be tested, to provide an indication of the practicality of this controller. Methods will be posed which help classifying the response of this system. Performance comparisons will be made between the proposed controller and a PID (Proportional, integral and derivative) controller. This will provide an indication to the success of the selected controller. Therefore the objectives of this project are to create a control system which could be applied to any nonlinear system of this nature. This control system must be able to reject the nonlinearities, give the most optimal response while operating in a way that does not damage infrastructure or create a risk to human life.

1.2. System and Background

Ensuring designs are safe and reliable is a major design goal for an Engineer. Selected tenets of the engineering code of conduct state that an engineer must demonstrate integrity in their work. Engineers must also practise their field of work competently to maintain the trust of the community (Engineers Australia 2010). Therefore it is imperative that an electromechanical control system for use by the public is safe and reliable. A Maglev train poses many potential risks to humans or property should failure occur. If the control system fails to correctly control the Maglev train and a crash occurs, the injury or death of many people could occur. An incident has already occurred relating to a Maglev train, which caused 23 people to be killed and many more to be injured (Brian Betts 2006). Therefore, to ensure the integrity of the engineer and to reduce the risks associated with control system failure, it is imperative that an understanding of control systems is created. To illustrate the importance of control systems in the real world, an example of a precision system is given in the following figure.



Figure 1.1: Artist's impression of the landing of NASA's rover 'Curiosity' (NASA, 2012)

1.2.1. Control system perfection in the modern world

The control systems which are required to control the thrust modules in Figure 1.1 must be precise. It can be seen that the failure to control this system effectively will cause the lander to crash, costing billions of dollars. The purpose of this project is to design a control system for a levitating magnet, which if successful, could also be used for a system such as the one shown above. The control theory seen here could be applied to any other control system.

1.2.2. Magnetically Levitated Trains

When applied to locomotives, magnetic levitation has been proven to be a viable alternative over a wheeled locomotive. Lee & Kim (2006) have researched and identified various control systems used by Maglev trains. The advantages outlined when comparing a Maglev train to a standard wheeled train include low noise, low maintenance, small turning circle and high speed capabilities. High efficiency is also identified as an advantage of Maglev trains. Figure 1.2 shows a Maglev train cross section.

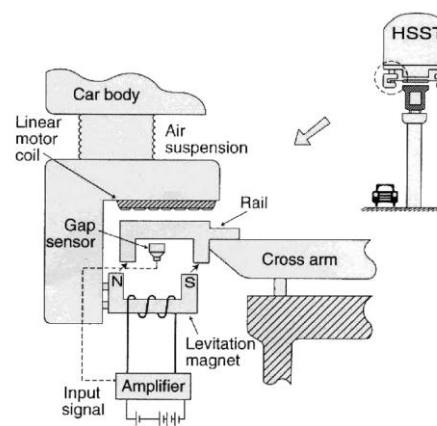


Figure 1.2: Cross-section of a Maglev train assembly (source: Lee & Kim 2006)

This image shows a resemblance to the Model 730 Maglev apparatus. The gap sensor is used as an input into the controller, and the levitating magnet coil is driven with current to ensure that the train stays within a certain tolerance. The propulsion of the train is achieved by creating a linear 'motor' which uses the stator with coils on the track, and the rotor (train) on the track. Figure 1.3 shows how this propulsive force is generated.

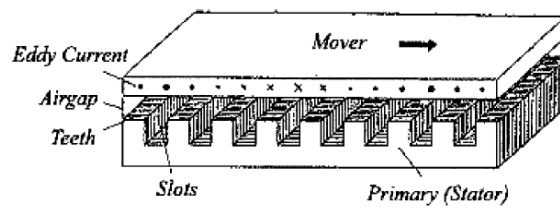


Figure 1.3: Representation of propulsion force generation (source: Lee & Kim 2006)

This shows how extensive Maglev control systems must be to correctly guide a train along the tracks, and also highlights the importance of the control system which governs a system like this. The following image shows this system in the real world, where it is applied to the German Transrapid train.



Figure 1.4: Maglev train (Brian Betts 2006)

1.2.3. ECP Model 730 Maglev Machine

In order to perfect a control system which could be used to control a maglev train, a magnetic development system must be used. This will give the ability to develop a magnetic control system, with little development and testing costs. Educational Control Products have designed a machine which provides a development system as described, and will be the centre focus of this project. The following image shows the Maglev plant to be used.

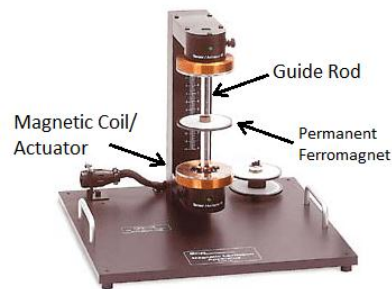


Figure 1.5: Maglev plant

The Maglev plant uses a coil with an aligned magnetic field parallel to a magnet on a guide rod. There is only one degree of freedom (pitch) for the magnet to travel in. The current in the coil determines the force applied to the magnet. A position sensor is present in the plant, which reads the instantaneous position of the magnet and reports it back to the controller. The control side of the plant is demonstrated in the following figure.

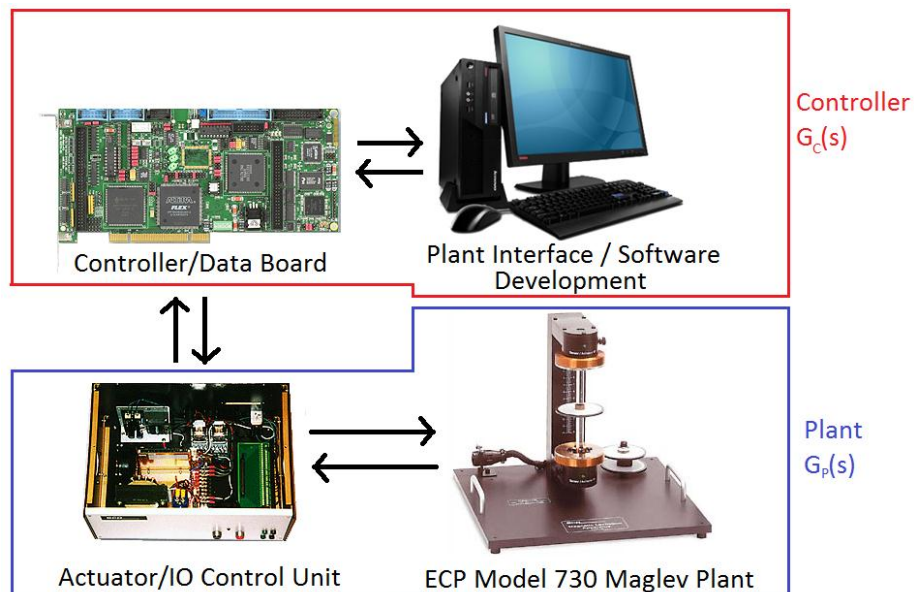


Figure 1.6: Maglev development system

The IO (Input/Output) control unit is a part of the plant and converts a control effort (proportional to current) provided by the real-time algorithm and amplifies this as a current into the plant coil. This unit also conditions the sending and receiving signals from the plant, to be read by the digital signal processor (DSP). There are also many

protection devices present in the IO control unit. The controller board has a real-time DSP on-board, which sends and receives signal data from the IO unit. This board uses a Motorola DSP, which applies a user-written control system to the sensor output. This generates a control signal, which is passed back to the control unit, forming a closed loop control system. The program executed on the real-time controller is user-written, entailing that any control logic can be applied to this plant. The controller written in C-code is denoted as $G_C(s)$. The last component in this system is the host PC, which is used as a user interface for the plant. It allows the compilation of C-code, and also gathers and plots information from the plant, including the instantaneous magnet position.

1.2.4. Nonlinear control background

A linear system definition is that the output of the system is directly proportional to a constant multiplied by any order of derivative of the input. Control theory is applicable to linear systems as the response of the system is easily determined. A non-linear system has a non-linear relationship between the output and the input. This problem is present in many control systems, and is a major design hurdle. A robotic arm is considered a nonlinear system, as the force applied by gravity to the arm is not directly proportional to the position of the arm.

Another system which is nonlinear is the Maglev machine described above. In a system with a magnet and a coil, the force applied to the magnet is not proportional to the distance the magnet is away from the coil. This causes a major problem when designing a controller for this system, as the transfer function which governs this system changes as the magnet moves. Therefore a control system must be posed which eliminates this effect, and controls the system to a 'deadbeat' response. The following section observes literature which is associated with the control of a non-linear system.

1.3. Nonlinear Control Systems

The topic of nonlinear control systems is well discussed in literary articles. Nise, N (2011) outlines the control theory phenomenon associated with nonlinear systems, and how they can be manipulated for easier analysis. The textbook written by Nise also contains an in-depth analysis for linear control systems and control system theories. It is stated that there are two distinguished categories in control systems, linear and nonlinear. The nonlinear system is defined as being non-homogenous and to lack superposition. The property of homogeneity is defined as when the input is multiplied by a scalar, the output is equal to the same scalar multiplied by the system output. This property will not be seen in nonlinear systems. The superposition property is defined as the sum of outputs of a system will be equal to the sum of the respective inputs multiplied by the system transfer function (Nise, N 2011). Figure 1.7 shows a nonlinear function, which by analysis could be proved to be non-homogenous and lack the superposition definition. By analysis, the multiplication of the input $R(s)$ by a scalar will not yield the output being multiplied by the same scalar, i.e. $KC(s) \neq KR(s)$.

Nise, N (2011) demonstrates an effective method in which to deal with the non-linear phenomenon involving Taylor's theorem of expansion. It is stated that when applied to a nonlinear system, Taylor's theorem can be expressed as the approximation noted in Equation 1.1. This equation shows that the tangential line at a given point is approximately equal to the straight line approximation between two close points on that line.

Equation 1.1

$$f(x) - f(x_0) \approx \left. \frac{df}{dx} \right|_{x=x_0} (x - x_0)$$

The above equation can be applied to any nonlinear system, provided that the plant model dynamics are known. When applied to a nonlinear system, the resultant tangential linear equation is seen in Figure 1.7. By inspection, it can be seen that for small deviations about point A, the linear function is approximately representative of the nonlinear system. It is also noted that larger deviations from point A yield larger

approximation errors, which is a major issue when using Taylor's theorem to linearize a system about an operating point. Nise, N (2011) then gives a worked example on how this linearization method is executed when linearizing a robotic arm based on the non-linear effect of gravity.

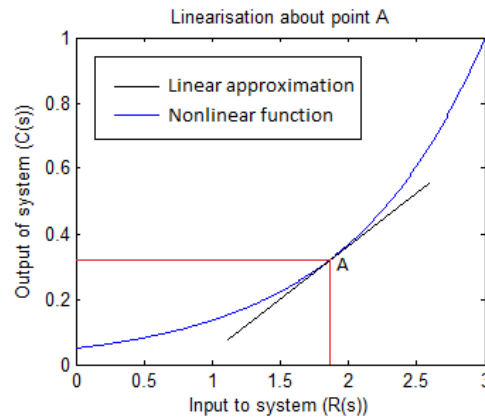


Figure 1.7: Linearization about an operating point, A (Nise, N 2011).

1.3.1. Design of Robust Deadbeat Controllers

The design procedure for a robust deadbeat controller is documented in a paper submitted to an IEEE conference by Dawes, Ng, Dorf and Tam (1994). This document demonstrates the methods required to design a robust deadbeat controller, as well as the advantages of using this type of controller over a Proportional, Integral and Derivative (PID) controller. This document also outlines the ability of the controller to handle any order of system. The ability of this type of controller to reject nonlinearities and external disturbances of up to 50% from the operating point is demonstrated. The ability to handle a 50% variation in the control system parameters is advantageous to a nonlinear system, as it will be able to reject a portion of the variations in the control system as described in the previous section.

The design concept uses a PID controller to achieve robustness (the ability to handle large system variations consistently) as it can reduce the steady-state error over time with an integral action. Dawes, Ng, Dorf and Tam (1994) show that coupling a PID controller with feedback, feed-forward and cascade gains will provide the ability to achieve deadbeat control of a non-linear, high order system. Figure 1.8 shows a graphical representation of this controller, with the constants which will be calculated in this project. A zero is defined as where the complex Laplace operator,

s , causes the transfer function to be zero as stated by Nise, N (2011). As shown in Figure 1.8, a zero is applied to the feedback loop, defined as $(1+K_b s)$. According to the text, the state variable feedback constant K_a forces the poles of the system to equal the zero in the feedback loop, effectively reducing the order of the system. All coefficients in Figure 1.8 are then designed to be on the real axis, so that the overall system will exhibit a deadbeat response.

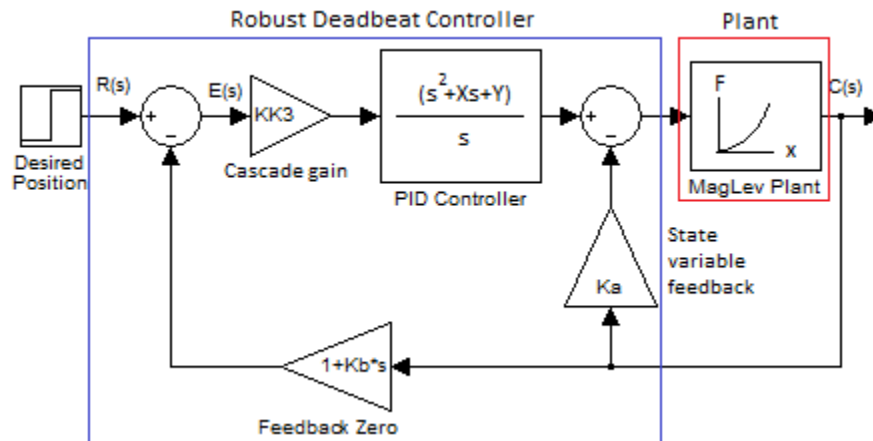


Figure 1.8: Robust deadbeat controller and plant

The figure above also shows how the Maglev plant will be interfaced with the proposed deadbeat controller. The variables must then be selected based on the order of the system which is to be used. Providing that Taylor's theorem is used to correctly represent the Maglev system, the robust deadbeat controller defined here can be directly applied. This gives a solid basis of the underlying concepts behind the controller to be applied to the maglev machine.

1.3.2. Adaptive solution to non-linear systems

Adaptive switching in control systems is a concept where a controller 'switches' the control law which it is currently applying to a system based on the present input and present output of the system. Angeli and Mosca (2002) show how this can be achieved using a digital controller. They state that inside their feedback loop of their controlled system, the control law applied is selected from a list of predefined controllers based on the current output of the plant. They also apply Lyapunov-based control law to their system to achieve robustness. It is shown that when applied in a digital sense, switching of control law parameters is quite simple, and can be

implemented using the 'IF' statement. This statement acts as an 'observer' as it tests the output, and applies the most optimal control law based on this output.

The aim of using the switching adaptive control is to set boundaries which constitute a switching of controller coefficients. In this paper, the boundaries are defined as points where the control law changes. This allows for each boundary to have a separate control law applied to it, where the control law will be optimised for that boundary. The results of the paper by Angeli and Mosca (2002) are indicated in a figure, which shows the improvement to the system response as a result of the switching control system. Another figure is also provided, which demonstrates the decision boundaries which are associated with this switching. This shows that adaptive switching is imperative to the Maglev system, as multiple linearized robust deadbeat controllers can be employed to remove the linearization error.

Callender, Cowan and Theodoridis (1991) have also used the adaptive switching control method when designing nonlinear filters. It is stated that using piecewise linear approximations for nonlinear control systems is advantageous due to the fact that the computation time required is much smaller than that seen in other nonlinear control techniques, including signal smoothing algorithms. This paper then outlines the simplicity of designing each filter based on the boundaries in which they are switched, for the linear approximations of that area. It is explained that say between the boundaries x and $2x$, controller A is used, and then between the boundaries $2x$ and $3x$, controller B is used and so on. Therefore from the information gathered, the adaptive switching technique is deemed suitable for rejecting the nonlinear phenomenon present in the plant.

1.3.3. ECP Model 730 Maglev control system

Literature related to control systems specific to the Model 730 Maglev machine is quite restricted. Most articles are related to the PID control and experimental development of linear control system theory, based on small movement displacements about the control point. Liceaga, Hernandez and Amezcua (2009) presented a paper to an electrical control conference based on the nonlinear control of the ECP model 730 magnetic levitation system. The issue addressed is the

nonlinear static-kinetic friction seen between the glass rod and the magnet in the Maglev plant. It is observable that a PID controller will accumulate an error when the magnet is slightly off the desired position, until the static friction is overcome and the magnet overshoots the desired position. This repeats indefinitely unless the magnet rests perfectly on the desired position.

Liceaga, Hernandez and Amezcua (2009) used a feedback linearization technique coupled with a second order linear, time invariant transfer function. The two poles of the second order function were chosen arbitrarily in order to get the best output of the system. Figure 1.9 shows the best response produced from the ECP Model 730 Maglev machine, where the change in position was 1 cm and the settling time was approximately 200ms.

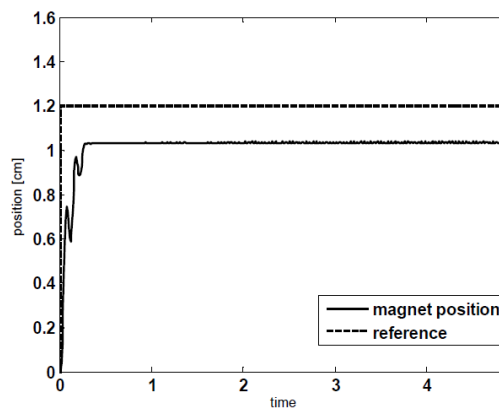


Figure 1.9: Maglev response using feedback linearization technique (Liceaga, Hernandez and Amezcua (2009))

The authors then went on to design an observer which switched the control algorithm based on the current output. This was designed to halt the integral accumulation inside the controller, in order to stop the machine from being affected by the static-kinetic issue. It can be seen that this observer effectively eliminated the stick-slip behaviour seen from the nonlinear friction on the maglev machine. For the purposes of this project, the ‘stick-slip’ behaviour of the friction seen between the glass rod and the magnet is assumed viscous and negligent.

Nataraj and Mukesh (2010) used an interesting method on controlling the ECP Model 730 Maglev system, involving the quantitative feedback theory (QFT). This paper claims to be the first to apply QFT to a magnetic levitation system. This theory

involves designing the controller to be able to accept any kind of disturbance and nonlinearity that could be seen on one of these plants. The trade-offs of this design are noted, involving stability, plant nonlinearities, performance and disturbance levels. Their results are shown to be slower, however much more consistent and robust.

The literature presented demonstrates many different approaches to both nonlinear control and deadbeat control methods, along with control system approaches to the Model 730 Maglev machine. The procedures found may be adapted and applied to this project. Although robust deadbeat control has not been attempted with an ECP model 730 Maglev machine, it can be seen that it is possible to apply the researched techniques to this machine.

1.4. Assessment of Consequential Effects and Ethical Responsibilities

1.4.1. Consequential Effects

Refer to the Appendix E for the risk assessment undertaken to ensure the responsible and safe use of the Maglev machine. Sustainability, the environment and engineering practice is an on-going initiative of the Institution of Engineers, Australia. The project outlined in this document must adhere to the ten aspects of sustainability set out by the institution. As outlined in the aim of this project, there is no manufacturing required other than that of the machine being used. The design of the Maglev Apparatus is compliant with appropriate Australian standards. An environmental impact which will be seen from the use of this machine is the power consumption. The use of coal to generate energy will have an impact on the environment, which should be kept in mind when the machine is in use. Although the machine is small, usage should be kept to a minimum.

Electronic waste also has a major impact in landfills when disposed of. Recycling and other disposal methods should be considered when the Maglev and associated hardware reaches its EOL (End of Life). Without the appropriate techniques for dealing with EOL products, severe damage can be caused to the surrounding areas of landfills where e-waste is disposed. Therefore consideration must be given while disposing of the Maglev apparatus and associated hardware.

The development of real-world applications of Maglev has already caused dangers to human life. As Maglev trains are capable of speed much greater than a wheeled locomotive, risks for animals, objects and humans on the track is greatly increased, including those on board the train. Therefore when developed, Maglev systems must have failsafe safety devices and procedures in place to reduce the risk of catastrophic failure.

1.4.2. Ethical Responsibilities

As a professional engineer, the ethical code set out by the Institution of Engineers, Australia must be adhered to. The first ethical responsibility of a professional engineer is to demonstrate integrity. This is adhered to by respect for oneself and others, being honest and acting on a well-informed conscience. This is a behavioural responsibility and will be adhered to throughout the project. Another ethical responsibility which must be adhered to is to practise engineering competently. To uphold the image of a professional engineer, one must constantly maintain and develop skills based on their discipline. This will ensure competency and keep public trust in the 'Engineer'. The competent engineer will also act on the basis of adequate knowledge; therefore will not attempt to perform work without having full competency in that area. Leadership is also a quality which all engineers must exhibit. Having good leadership qualities will uphold the trust for engineering practices. Sustainability must be promoted while executing the project. Responsibility extends from the community to the environment, and it is imperative that the engineer keeps the wellbeing of these aspects to heart.

1.5. Methodologies, Resources and Timelines

In order to execute a major technical task by the required completion date, deadlines, resources and methodologies used must be defined. The resources must be defined with lead times and contingency plans in place. This will ensure that before the task is started, delays are understood and measures can be taken to reduce downtimes. Milestones which define key positions within the project execution must be defined. A timeline will then be devised which will place these milestones with deadlines, and execution requirements. Contingencies will be put in place for unforeseen events which will hinder the progress of the task. Refer to sections 1.5.2 and 1.5.3 for an analysis of the resources required, and the timelines which must be adhered to.

1.5.1. Task Execution Methodology

Strategies must be set up to streamline the execution of the task. The design of these strategies must give light to research, method of approaching, execution and review. This technical task has two distinct parts, which are the research and theoretical analysis, and the physical implementation. It is imperative that the research and

theoretical counterparts are satisfied first, in order to ensure that the system is understood before physical implementations occur. This will reduce the risk of error due to a lack of understanding of the system. The following subsections outline the method in which each task required for this project will be executed. The entire analysis will be applied to each task.

Research

Methods in which the task may have been executed previously must be researched. This will ensure that any methods and issues arising from other attempts at the subject matter are understood. This can help reduce time which is wasted on certain areas which have been previously fixed. Research will also provide a better understanding of the task itself, which will in turn provide a better approach and reduce time wasted from not properly understanding the task at hand.

Approach Analysis

Once all research has been completed, steps can be devised which will guide the execution of the task. This will ensure that the task is executed with the main goal in mind, and reduce the risk of going off topic or heading in the wrong direction. A better understanding of the goal will also be made.

Execution

The execution of the task will rely on the information gathered in the research and approach analysis. If major issues are presented by the task, the above steps should be backtracked until an alternative is found.

Review

After the task is executed, the entire task should be reviewed. Questions that will be asked include: what was the outcome of the task? Was the method efficient? Could the previous steps be changed to streamline the process? This will allow for better methodologies for future tasks.

1.5.2. Resource Analysis

To ensure that all resources are acquired on time and to avoid delays, all resources which will be required for the completion of the project must be identified. The

methods on which these resources are acquired and the timeframes involved will also be stated, where contingency plans are also devised. This will ensure that no undesirable delays are experienced while executing the project. The following resources are or will be required throughout the duration of the project.

Maglev Apparatus

This project is based on the ECP Model 730 Magnetic Levitation machine shown in Figure 1.10. This figure shows the ‘plant’ section of the electromechanical system. This product also comes with a rectangular control system, and a PCI interface board which is required to be installed into a personal computer.



Figure 1.10: ECP Model 730 Magnetic Levitation Machine (Source: Educational Control Products 2002)

There are two machines which have been purchased by the University of Southern Queensland, both of which are available in the control systems laboratory. Therefore there are no delays for work which must be performed on this machine. As there are two devices ready for use, in the event of a system failure on one machine, the other machine is able to be online instantly.

ECP Control System Software

The control system software which relays information and control algorithms between the user and the Maglev apparatus requires alterations when new control system algorithms are written. This program is imperative for the use of the machine, and if corruption occurs, will cause delays when the program installation files are required to be tracked down. Therefore a backup has been made of the control

system product to remove the time loss threat which is posed if the program were to malfunction. Figure 1.11 shows a representation of the program described above.



Figure 1.11: ECP Control System Software (Source: Educational Control Products 2002)

Development Interface (Personal Computer)

The development interface used to edit the control algorithms which are used by the Maglev controller is written for a personal computer. This interface program (by ECP) also provides data logging, real-time analysis and the ability to input any kind of disturbance to the Maglev system. The personal computer required for this task also requires the connection of a PCI card, which provides the ability to transfer data to the Maglev control system. If this personal computer failed, a replacement personal computer will be sourced from the faculty and the PIC interface card will be installed. Downtime will be considered insignificant compared to the duration of the project.

MATLAB Program and computer system

The MATLAB program (student version) with appropriate toolboxes, along with Simulink has been purchased prior to the commencement of this program. The use of this program for the final year project does not conflict with the terms and conditions set out in the student version. A desktop and Laptop computer will be used in the development and simulation of MATLAB scripts. These two machines will provide redundancy for the duration of the project. Catastrophic failure may cause a downtime of a week while a new machine is sourced.

Data Storage and Protection

This project requires the use of many different data and file types, including MATLAB and Word files. As data loss will have catastrophic consequences toward the project, contingencies and backup plans must be put in place. All files will be stored on a USB drive attached to the desktop computer. This USB drive will be backed up weekly to another USB drive, which will not be used to edit any files. The USB on the desktop computer has write permissions enabled for the user on the laptop computer; therefore the files can be accessed and edited via a wireless uplink. Temporary files will be used when the two computers are out of wireless range.

1.5.3. Time Usage Guidelines

In order to ensure that the project is executed seamlessly across the allocated time frame, it is imperative to identify certain milestones which must be reached. These milestones will be assigned a completion date which will include contingencies in the event of delays. The following table outlines the identified milestones, Tasks which must be completed prior to starting, the anticipated finish time and the current progress.

Table 1.1: Milestones

Milestone/Task	Prerequisite
1) Investigate the ECP Model 730 hardware and software	None
2) Test the system and study the existing demo program and experiments	1)
3) Write a script to linearize the plant using Taylor's theorem	1)
4) Design and simulate a dead-beat controller for the Maglev system	1), 3)
5) Implement, test and evaluate the design using the Model 730 Maglev plant	2), 4)
6) Design and simulate a switched adaptive deadbeat controller for the plant	4)
7) Implement, test and evaluate the switched adaptive deadbeat controller using the Model 730 Maglev plant	5), 6)
8) Analyse the results	7)
9) Compile Dissertation	Ongoing

Chapter 2

ECP Model 730 Maglev System

2.1. ECP model 730 Maglev Machine

This section draws information from the ECP Model 730 Maglev user manual (Parks, T 1999). Figure 2.1 provides a visualisation of the Maglev development system. It can be seen that there is one degree of freedom (vertical) for the magnet. There are two coils in the machine, and the potential for one or two magnets to be used at any single time. The bottom coil is effective for the bottom half of the glass rod, and the top coil is effective for the top half (with minor cross-coupling). SISO (Single Input, Single Output) operation uses the bottom coil and one magnet, and the top coil is not energised.

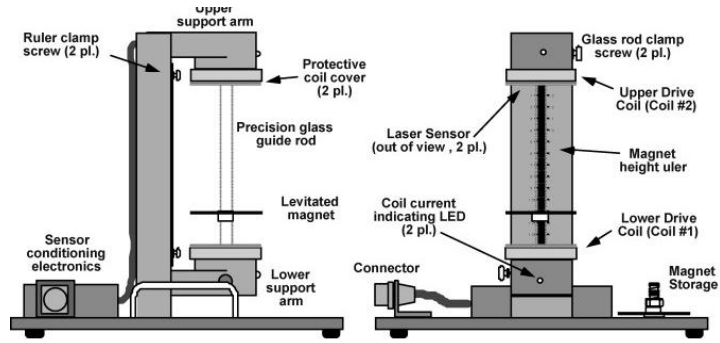


Figure 2.1: ECP Model 730 Representation (Source: Educational Control Products 2002)

Figure 2.2 shows the interaction of forces which are exerted on the magnets (MIMO configuration), and Table defines all the variables and constants used when calculating the interaction forces.

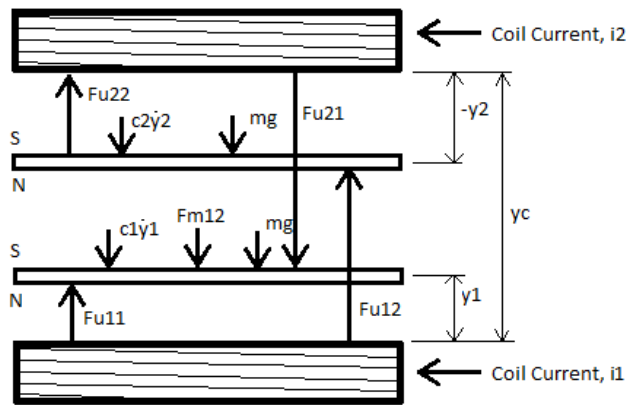


Figure 2.2: Free-body diagram with force interactions (Parks, T 1999).

Table 2.1: Definition of Maglev plant-specific variables

Constant or Variable	Definition
c_1, c_2	Friction/wind resistance. Modelled as Viscous
F_{m12}	Force between each magnet
F_{u22}	Force applied to magnet 2 from coil 2
F_{u11}	Force applied to magnet 1 from coil 2
F_{u21}	Force applied to magnet 2 from coil 1
F_{u12}	Force applied to magnet 1 from coil 2

m	Mass of the magnet, 120g
g	Force of gravity, assumed 9.81m/s^2
y_1	Distance between magnet 1 and coil 1. The linear sensor gives 10000 counts/cm
y_2	Distance between magnet 2 and coil 2. The linear sensor gives 10000 counts/cm
y_c	Distance between the coils
i_1, i_2	Current through the respective coils (input)
U_1, U_2	Control effort, proportional to i_1, i_2 . Linearization gives 10000 counts/N
a, b, c, d	Constants which are found from linearizing the system

This project specifically uses a single magnet with the bottom coil, in a SISO configuration. Therefore all forces related to coil 2 and magnet 2 is null. The following equation shows the summation of forces in the system.

Equation 2.1

$$m\ddot{y}_1 = F_{U_{11}} - mg$$

Where:

Equation 2.2

$$F_{U_{11}} = \frac{I_1}{a(y_1 + b)^4}$$

Note from the equation above, the coil current I is directly proportional to the control effort U . The control effort is the output of the real time algorithm, and the IO controller converts this to an actual current. From Equation 2.1, it can be seen that as the distance increases between the coil and magnet, a much greater control effort (current) is required to produce the same force (Parks, T 1999). This creates nonlinearities in the control system, and must be compensated for if the plant

parameters are to be realised. The next section shows the force-distance relationship, which is nonlinear.

2.2. Nonlinear Actuator (coil) Characteristics

The relationship between the distance of the magnet from the coil and the force imparted on the magnet is not proportional, for a unit of current through the coil. Therefore in order to apply Taylor's linearization theorem to this situation, a linear approximation of this nonlinear characteristic must be made. ECP states that there may be up to a 10% variance in the force/distance/current equation for any Maglev machine. Therefore for the accurate realisation of the Maglev system parameters, this relationship must be numerically calculated. To obtain the force/distance/current relationship, the machine must be energised with different coil efforts (proportional to current) and the height at which the magnet settles is recorded. The following figure shows the measured values from the Maglev plant.

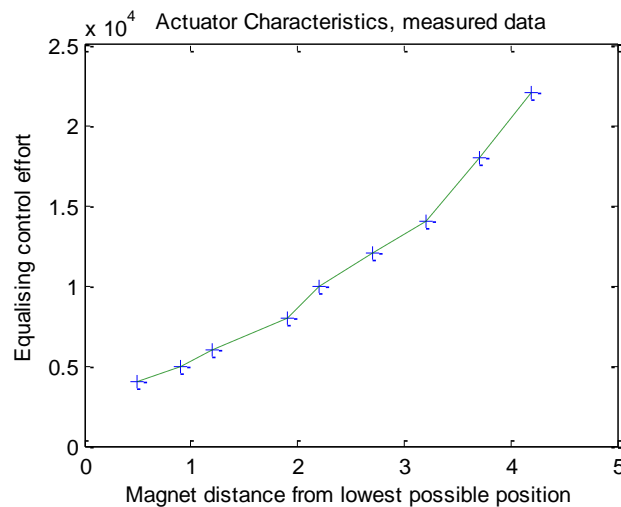


Figure 2.3: Physical measurements obtained from the Maglev plant

The method used to obtain these measurements was to energise the coil with different intervals of 'control effort' and measure the position at which the magnet rests. Note that this data may have discrepancies from the actual characteristics of the plant due to static friction, which may cause the magnet to rest at a position close to the real position.

According to the ECP manual for the model 730 Maglev machine, the force applied to the bottom magnet based on the current (control effort) through the coil can be represented as

Equation 2.3

$$F_{U_{11}} = \frac{U_1}{a(y_1 + b)^N}$$

Typical values for the order of the approximation (N) typically range from three to five. It is assumed that an accurate approximation can be obtained using a fourth order approximation (N=4). Using the MatLab script shown in Appendix B, the following values have been obtained which accurately represent the magnet to magnet force characteristic. Via numerical analysis, the following constants have been calculated to satisfy Equation 2.3.

Equation 2.4

$$a = 1.64, b = 6.2$$

Using Equation 2.3 and Equation 2.4, the following shows how the estimated curve follows the data points.

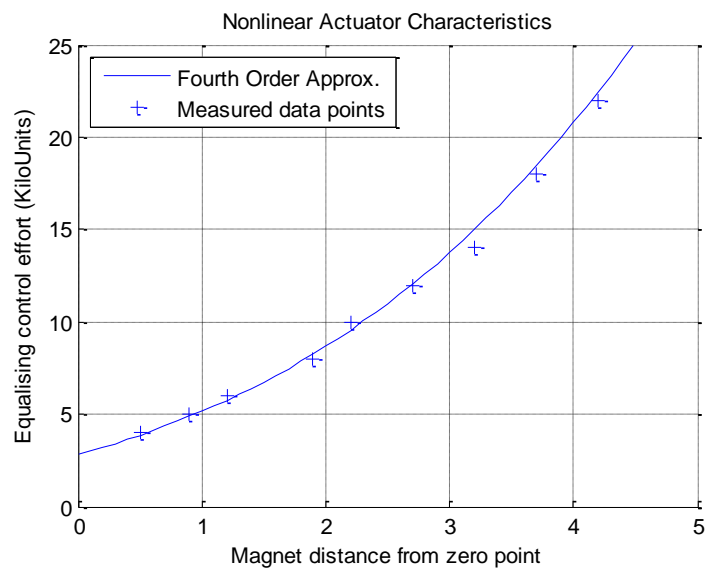


Figure 2.4: Representation of the estimated curve against the experimental data

2.3. Linearization using Taylor' Series expansion

Figure 2.4 shows the nonlinear characteristics of the system. As this estimation is nonlinear, control law is unable to be used on this system. Control law can only be used on a linear system. An effective method to control a system of this nature is to nominate an operating point, and assume that the system is linear for deviations from this operating point. This will give the ability to apply linear control law to the system which will be accurate for small deviations about this point. Nise, N (2011, pp. 88-91) shows how the Taylor's series expansion method is applied to a nonlinear pendulum system. The result is a linearized system for small excursions about the operating point. The methods used by Nise have been adapted to apply to the Maglev plant. The Taylor's series expansion method is now shown.

Equation 2.5

$$f(x) - f(x_0) \approx \left. \frac{df}{dx} \right|_{x=x_0} (x - x_0)$$

or, $\delta f(x) \approx m|_{x=x_0} \delta x$

According to the technical documentation, the equation which represents the control system in terms of differentials is:

Equation 2.6

$$\frac{d^2 y_1}{dt^2} + \frac{c_1}{m} \frac{dy_1}{dt} + \frac{c}{m(y_{12} + d)^4} + \frac{U_2}{ma(y_c - y_1 + b)^4} + g + \frac{U_1}{ma(y_1 + b)^4} = 0$$

Note that the above equation has all terms on the left side. This shows that the system is at equilibrium at the operating point. The control effort at the operating point of 2 cm (y_{1_0}) has been found experimentally to be 8000 counts (u_{1_0}). The small deviations about the operating point are represented as δy_1 . The friction will be assumed null as it is negligible in this system and nonlinear in nature. Therefore at equilibrium, using the Taylor's series expansion equation (Equation 2.5), Equation 2.6 is manipulated to get the linearized equation:

Equation 2.7

$$f(y_{10}, y_{20}, u_{10}, t) = -\delta u_1 \left. \frac{\delta f}{\delta u_1} \right|_{y_{10}, y_{20}, u_{10}} - \delta y_1 \left. \frac{\delta f}{\delta y_1} \right|_{y_{10}, y_{20}, u_{10}} - \delta y_2 \left. \frac{\delta f}{\delta y_2} \right|_{y_{10}, y_{20}, u_{10}}$$

Resulting in

Equation 2.8

$$\frac{\delta u_1}{ma(y_{10} + b)^4} = \frac{d^2 y_1}{dt^2} + \frac{4c\delta y_1}{m(y_{120} + d)^5} + \frac{4U_1\delta y_1}{ma(y_{10} + b)^5} - \frac{4c\delta y_2}{m(y_{120} + d)^5}$$

As the SISO system is of interest, the top coil-magnet and magnet-magnet forces can be nulled.

Equation 2.9

$$\delta u_1 \frac{1}{ma(y_{10} + b)^4} = \frac{d^2 y_1}{dt^2} + \delta y_1 \frac{4U_1}{ma(y_{10} + b)^5}$$

This system can now be represented in the state-space form,

Equation 2.10

$$x = \begin{bmatrix} y_1 \\ \dot{y}_1 \end{bmatrix}, A = \begin{bmatrix} 0 & 1 \\ \frac{4U_1}{ma(y_{10} + b)^5} & 0 \end{bmatrix}, B = \begin{bmatrix} 0 \\ 1 \\ ma(y_{10} + b)^4 \end{bmatrix}, C = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

The above equation gives a linear approximation about the desired operating point (y_{10}) and the corresponding equalising control effort (U_1). Figure 2.5 shows a representation of the above theorem and how it applies to the Maglev machine.

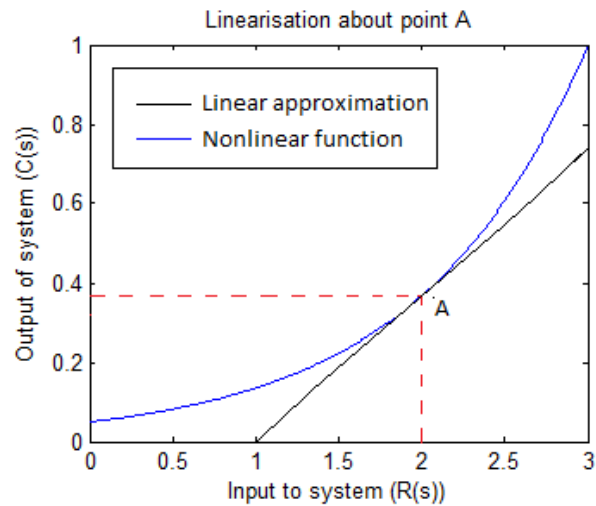


Figure 2.5: Taylor's linearization theorem representation, about point A (2cm)

From the figure above, it can be seen that the linearization theorem exhibits the highest accuracy when the magnet is close to the linearization point, A. However, the accuracy degrades exponentially as the magnet moves away from point A. Therefore it is deemed that this model of the system is accurate for small magnet movements about the linearization point A. Now that an estimation of the system has been made, the robust deadbeat control theory can now be analysed and applied.

Chapter 3

Robust Deadbeat Control Technique

3.1. Robust Deadbeat Control Theory

3.1.1. Introduction to Robust Deadbeat Control

A PID controller achieves a response by adjusting process variables until the feedback error is zero (Nise, N 2011). This response is considered robust as it is able to resist change in the system variables without violating the integrity of the controller response. While the response of the PID controller is robust, the timeliness of the response is gradual as the process variables must have time to adjust to the system. Therefore the response characteristics of a PID controller are not considered to be deadbeat. A 'deadbeat' response is defined as 'controlling a system to the desired position in the shortest time' (Nise, 2011). When applying this definition to the Maglev system, robustness is required to reject the nonlinear characteristics of the magnet as well as external disturbances. Deadbeat control is required to achieve an optimal response.

A solution for the robust deadbeat control of a known system has been proposed by Dawes, Ng, Dorf and Tam (1994). This solution includes the robust control abilities of a PID controller, and applies feedback poles and constants which allows the deadbeat response property. Figure 3.1 shows a diagram of the proposed robust deadbeat controller.

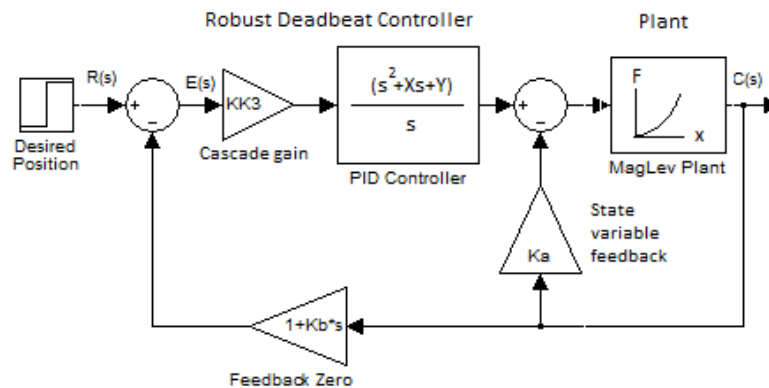


Figure 3.1: Deadbeat Control Technique (adapted from Dawes, Ng, Dorf & Tam 1994).

From Figure 3.1, a feedback ‘zero’ is applied in a feedback loop. Complex numbers in the ‘s’ domain which cause the system to be ‘zero’ are considered zeros. Therefore the feedback loop in this system $(1 + K_b s)$ will be zero when $s = -1/K_b$. After the feedback zero has been applied to the system output, the error between the desired position and the current position is calculated. A PID controller is then applied to this error signal, making the system robust. A state variable feedback constant is applied to the output of the controller, based on the output of the plant.

The net result of the control theory described is the reduction of the order of the maglev system, which contributes to the robust deadbeat control criteria. The next goal is to place the zeros of the system on the real axis, resulting in a robust system with deadbeat response (Dawes, Ng, Dorf & Tam (1994)).

3.1.2. Design Method

The robust deadbeat controller must be designed to exhibit the zeros described above. A linear approximation of the plant has been calculated using Taylor’s theorem. Dawes, Ng, Dorf and Tam (1994) claim that the robust deadbeat controller can reject up to a 50% change in the system dynamics before the response of the controller is affected. Therefore it is assumed that for small movements of the

magnet, the controller is able to reject the nonlinear error as described in Section 2.3. Note that the procedures for applying this controller do different order plants differ slightly. The following procedure outlines the method which will be used to apply a robust deadbeat controller to a second order plant, as the approximation for the Maglev characteristics is second order.

The constants in the deadbeat controller must be calculated based on the transfer function which describes the plant. Due to the nature of the controller, a majority of the constants are calculated using the prescribed control theory, and the other constants are arbitrarily selected. The arbitrary constants are selected and varied until the best system response is found. From Figure 3.1, the following constants associated with the deadbeat controller are:

- X =Proportional gain in the PID controller (dependant)
- Y =Integral gain in the PID controller (dependant)
- K, K_3 =Feed forward gains (both arbitrary)
- K_b =Feedback derivative (dependant)
- K_a =Cascade gain (dependant)
- T_s =Desired settling time (arbitrary)

Note that the ‘settling time’ is the time in which the system reaches 2% of the desired position. This is a performance criteria, and will be used to test the system once the testing stage has been reached. The following steps have been prescribed to calculate the abovementioned constants.

1. Approximate the plant transfer function $G_P(s)$
2. Let $K=1$ (or any other arbitrary number)
3. Have the characteristic equation of the system equal

Equation 3.1

$$C.E = s^3 + \alpha w_n s^2 + \beta w_n^2 s + w_n^3$$

4. Find the alpha, beta gains based on the order of the plant
5. Compare the characteristic equation above to the closed loop transfer function of the controller and plant, finding X, Y etc.

6. Simulate the system with different K values until the best response is found

3.2. Design procedure for the Maglev system

3.2.1. Maglev design method using the linearized plant

The next step in the control system design process is to calculate the controller parameters specific to the Maglev plant. The design of the controller must use an approximation of the plant in order to achieve a deadbeat response. The approximation for the preliminary design was made to be about a magnet position of 2cm. As stated, linearizing the plant about 2cm will cause the plant to be accurate at this position; however the accuracy will degrade as the magnet moves away from this position. Equation 3.2 shows the result of applying Taylor's theorem to the plant for a magnet position of 2cm.

Equation 3.2

$$G_p(s) = \frac{1111}{s^2 + 4s + 478.5}$$

This equation is a linear approximation of the plant parameters. The characteristic equation of the plant is to be compared with Equation 3.1. Dawes, Ng, Dorf and Tam (1994) state that for a second order system, the following constants are to be used:

Equation 3.3

$$\alpha = 1.90, \beta = 2.20, Tr_{90} = 3.48, T_s = 4.04$$

Combining Equation 3.1 and Equation 3.3, the characteristic equation is described as:

Equation 3.4

$$C.E = s^3 + 73.104s^2 + 3256.9s + 56961$$

The closed loop transfer function of the controller and plant (Figure 3.1) is described as:

Equation 3.5

$$G(s) = \frac{G_c(s)G_P(s)}{1 + G_c(s)G_P(s)KK_3 + G_c(s)G_P(s)(1 + K_b)}$$

Substituting the plant and controller parameters, the entire system transfer function with the controller variables is:

Equation 3.6

$$s^3 + \frac{s^2(1110KK_3 + 1110KK_3K_bX + 4)}{1110KK_3K_b + 1} + \frac{s(1110K_a + 1110KK_3X + 1110KK_3K_bY + 479)}{1110KK_3K_b + 1} + \frac{1110KK_3Y}{1110KK_3K_b + 1}$$

The characteristic equation is the entire denominator of the above equation. Note that the s-variable orders (s^2 , s etc.) have been separated, which assists in the following comparison between the characteristic equations. The following equation shows the comparison between Equation 3.3 and Equation 3.6.

Equation 3.7

$$\begin{aligned} \frac{(1110KK_3 + 1110KK_3K_bX + 4)}{1110KK_3K_b + 1} &= 73.104 \\ \frac{(1110K_a + 1110KK_3X + 1110KK_3K_bY + 479)}{1110KK_3K_b + 1} &= 3256.9 \\ \frac{1110KK_3Y}{1110KK_3K_b + 1} &= 56961 \end{aligned}$$

By choosing arbitrary K , K_3 and T_s values, the variables X , Y and K_a can be found using simultaneous solutions. Note that K_b is directly related to the T_{desired} value. It can be seen that varying both K and K_3 will result in dramatic changes in the control constants, and that these values must be optimised in order to get the best response.

3.2.2. Matlab script design for fast deadbeat control calculations

The design specifications of the robust deadbeat controller require that the constants must be varied to find the best response of the plant. As the computation time for all of the plant parameters is intensive, a Matlab script was written to perform the above calculations based on the user selected constants. This script was improved to simulate the controller in a loop, and repeat while numerically converging toward the

best system response. Appendix B shows the Matlab script written to do this. This script also designs the controller based on any linearization point, allowing the best controller to be designed for any magnet position.

3.3. Simulation Analysis

3.3.1. Plant Simulation

Before the controller was implemented on the plant, it was simulated using a professional simulation package. The reason for simulating the controller first was to ensure that the chosen variables did not cause the plant to operate in an unsafe manner. Inputting the wrong variables could cause the plant to overdrive the coil or operate the magnet in a dangerous manner. The simulation package used is Simulink, which can be called using a Matlab script. Due to the large amount of large amount of computations required for the numerical method, an integrated script will reduce the time required in finding the best response of the system.

An accurate version of the plant (the Maglev machine) must first be designed in Simulink before the robust deadbeat controller is implemented. The nonlinearity present in the maglev machine is a complication which cannot be perfectly simulated. Therefore, an assumption was made with the simulation that the plant parameters will be linear about the desired magnet position. The nonlinear errors present with this approximation are the same as that described in Section 2.3. It is deemed that for large excursions, the simulation response time will be slightly inaccurate. This error will be reduced as the magnet gets closer to the desired position. When applied to the plant, the results of the simulation may not perfectly reflect the results of the plant, however will provide a good indication of the controller's performance.

The Matlab script shown in Appendix B uses Taylor's theorem to obtain a linear approximation of the system. When the Simulink model is called, the linear approximation parameters are passed through to a transfer function block, which simulates the system about the desired control point. Figure 3.2 below shows the plant parameters for a linearization point of 2cm.

Due to the limitations of the Maglev plant, a block was added to the input of the plant in the simulator, which clamps the controller output to the maximum current allowable. This clamping will occur if the controller output exceeds 4 amperes of current (positive or negative), protecting the plant from over-current conditions. This was done to ensure that the controller designed for simulation mimics the Maglev plant.

3.3.2. Controller simulation

Now that an approximate plant has been created which represents the system at the desired settling point, the robust deadbeat controller can be designed. The Simulink model is designed to accept constants from the Matlab script in order to automate the entire simulation process. The simulation model must also be able to pass data from the input, output and other key points of interest back to the Matlab variable workspace. This will help in plotting the data, finding the settling time and system diagnostics. Figure 3.2 shows the basic Simulink model which was used to simulate the controller and plant. Due to the vast differences between the simulation and the Maglev machine, there may be large discrepancies between the simulated and actual responses.

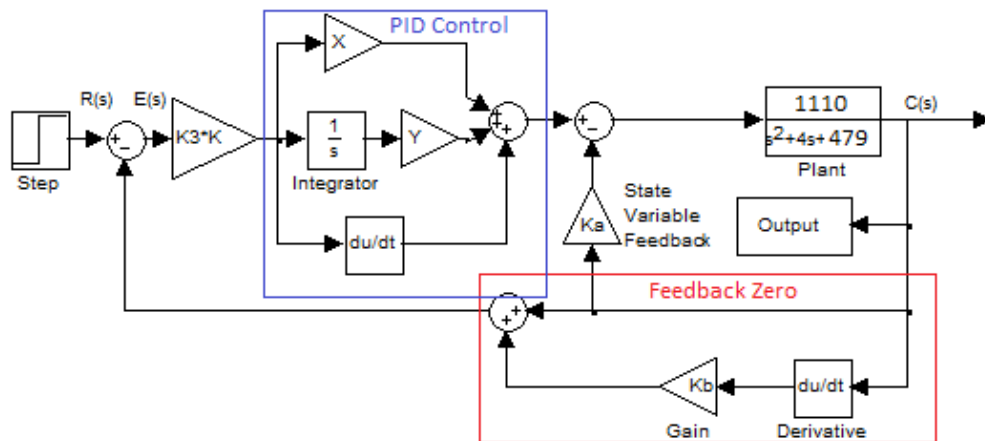


Figure 3.2: Robust deadbeat controller simulation model

It can be seen that the basic controller representation in Figure 3.1 has been broken down into sections, each being created using certain blocks in Simulink. The result is shown in Figure 3.2. The following s-domain definitions were used to convert the controller to the complex domain: (Nise, N (2011))

- The proportional component, $\mathcal{L}[Kf(\tau)] = KF(s)$
- The integral component, $\mathcal{L}\left[\int_0^t f(\tau)d\tau\right] = \frac{F(s)}{s}$
- The derivative component, $\mathcal{L}\left[\frac{df}{dt}\right] = sF(s)$
- The PID control, $PID = K_P f + K_I \int_0^t f(\tau)d\tau + K_D \frac{df}{dt}$

The relationships above were used when converting the s-domain model to a time domain model, suitable for Simulink. Figure 3.2 shows the result of converting the robust deadbeat controller to the time domain. The feedback zero, $(1 + sK_b)$, was converted to the summation of the output and the K_b constant multiplied by the derivative of the output. This signal was then negated from the input, creating an error signal. The PID controller inside the robust deadbeat controller is a simple summation of the proportional, integral and derivative of the error signal, multiplied by the respective gains.

The cascade gain, K_a , is then negated from the output of the PID controller, giving the final control signal which is fed into the Maglev plant simulation block. This completes the controller design in Simulink. The constants are pre-computed, and Simulink handles the actual step-by-step simulation.

3.3.3. Simulation examples

The simulation was then tested with arbitrary constants. Figure 3.3 shows the results from the initial simulation.

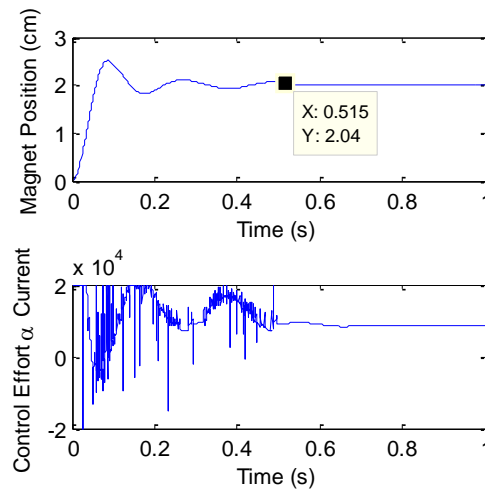


Figure 3.3: Initial Simulation Results (settling point inset – 515ms)

It can be seen from Figure 3.3 that the simulation worked to a satisfactory standard, however is very unresponsive and contains unwanted oscillations. The coil current (input into the plant) contains large spikes which could cause unwanted amplifier and coil stress. Therefore the results of the initial simulation of the controller were deemed acceptable; however a more satisfactory response will be obtained by optimising the system constants. Due to the nature of the arbitrary constants, the following section shows how the constants were chosen for the best response.

3.4. Performance Optimisation using Numerical Methods

3.4.1. Demonstration of the user-defined parameters

It was found that smaller T_{desired} values gave much higher coil currents, which were clipped by the simulation logic, and caused the system to oscillate. This was a desired outcome, as it is seen that the clamping of the coil current protected the machine from overcurrent conditions. The table below shows the effects of varying the arbitrary constants.

Table 3.1: Comparison of constants

Constant Type:	Test 1	Test 2	Test 3	Test 4
Arbitrary (Independent)	$T_{\text{desired}}=0.2$ $K=0.1$ $K_3=0.01$	$T_{\text{desired}}=0.2$ $K=0.1$ $K_3=10$	$T_{\text{desired}}=0.2$ $K=0.01$ $K_3=0.01$	$T_{\text{desired}}=0.1$ $K=0.01$ $K_3=0.01$
Dependant	$X=949.6$ $Y=22820$ $K_a=-0.7813$	$X=43.98$ $Y=1418$ $K_a=-1.108$	$X=9182$ $Y=217430$ $K_a=-0.7783$	$X=38098$ $Y=1.73 \times 10^6$ $K_a=-1.9648$
Settling Time (s)	0.273	0.372	0.273	0.191

It is also noted that for certain combinations of K and K_3 , the system is completely unstable. From the above table, it is seen that the arbitrary values have a large effect on the system parameters and response times. Therefore it is imperative that the correct values are chosen in order to obtain the smallest settling or response time.

3.4.2. Optimisation of the plant model using numerical methods

As stated in Section 3.2, the dependant constants must be calculated from the arbitrary constants. As there are more than one arbitrary constant, numerical procedures were used to converge the simulation response toward the best result. The Matlab script was written in a way that calculates the steady state response time from each simulation (The script is shown in Appendix B). This then allowed the script to be written in three nested loops, where each loop varies one parameter slightly. Once all possible simulation parameters have been simulated, the script then returns the constants which gave the quickest settling time into the command screen. The resolution of the steps for each loop cycle determined the accuracy of the constants resulting in the best controller. Using a fine resolution, the constants which resulted in the smallest settling time of the system were returned.

3.4.3. Simulation Performance Results

The previous sections detailed the methods which are used to find the constants responsible for the best response of the controller and plant. This data was then used

to simulate the system, and the following figure was generated which represents the best possible response. Higher T_{desired} parameters caused current clipping, and resulted in unwanted oscillations.

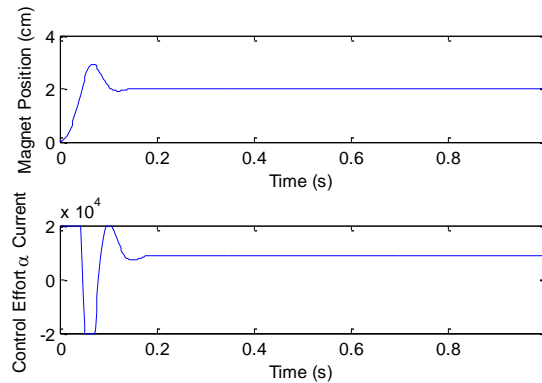


Figure 3.4: Best Simulink response for a 2cm movement

The controller shown in Figure 3.4 is linearized for a movement about 2cm, therefore the settling time may differ to that seen on the plant. The movement is from magnet rest to 2cm, and the response is seen by the top figure. The settling time is 181ms, which is quite satisfactory when compared to the PID response time of around 600ms (found in Chapter 7 of this report). When observing the initial current surge at <50ms, the current limiter clamps the machine to maximum current, which is quite acceptable, as the inertia of the magnet must be overcome.

As the controller is only designed for movements about 2cm, it is known that the linearity error will increase as the desired position moves away from 2cm. Figure 3.5 shows the error generated by deviating 1cm away from the linearized point, to 3cm. The oscillations can be seen, and this characteristic is also present on the Maglev machine.

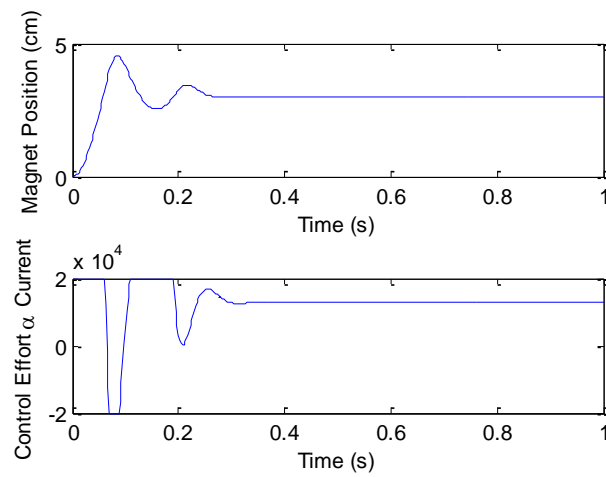


Figure 3.5: Effect of moving to 3cm, for a linearization point of 2cm

When the desired position is moved more than 2cm from the linearization point, the plant becomes too unstable and exhibits violent oscillations. The next chapter proposes a solution to this issue.

Chapter 4

Adaptive Deadbeat Controller Design and Simulation

4.1. Piecewise Adaptive Controller

As Figure 3.5 in Section 3.4.3 describes, the effect of the nonlinearity error as the magnet moves away from the linearization point is seen. As the desired position moves away from the linearization point, oscillations are seen in the magnet. The system that has been designed here is only effective for $2\pm 0.5\text{cm}$, instability and dangerous motion is seen outside these bounds. The design process for the robust deadbeat controller can be simply altered for different linearization points. The Matlab script used in designing the first controller is designed so that it can handle any linearization point, and can also simulate the controller using this linearization point.

As the deadbeat controller can only be designed for one linearization point, a system must be devised that is able to select the most optimal controller based on the desired

position of the magnet. This type of system is termed a ‘piecewise adaptive’ or ‘switched adaptive’ controller, as it segments the operation range into discrete intervals. Each interval has its own control parameters, optimised for the linear approximation for the relative operating point. Figure 4.1 illustrates this operation.

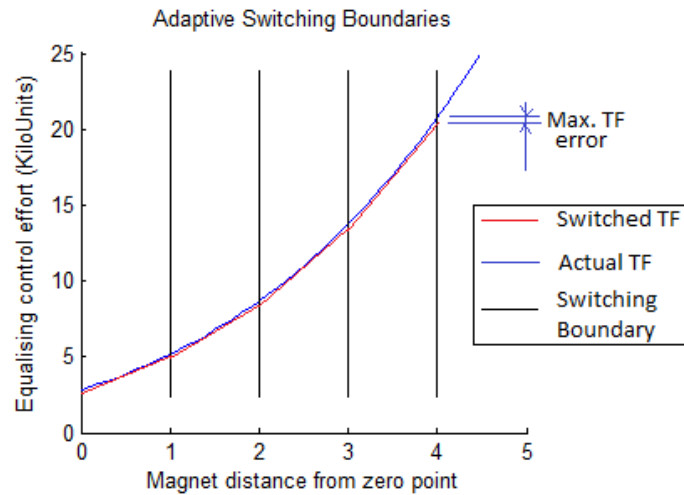


Figure 4.1: Representation of a piecewise adaptive control technique

From Figure 4.1, the operation of a switched (or piecewise) adaptive controller can be observed. The linearization has been performed for each segment, and the boundaries of these segments have been set to $\pm 0.5\text{cm}$. If the magnet is to be operated at 3.5cm , the controller which is most optimal at 3.5cm will be selected. The response time will be as accurate as the single linearized counterpart for that point. The advantage of this control technique is that the deadbeat theory can be applied to the $0\text{-}4\text{cm}$ range with a minimal linearization error. Figure 4.1 also illustrates the maximum linearization error present when this control technique is used. Note that a greater number of segments will result in the better reproduction of the nonlinearity.

It can be seen that the transfer function linearization error seen in the deadbeat controller is a major issue, and must be overcome if the system is to be operated across the entire $0\text{-}4\text{cm}$ range. Therefore the controller which performs the switching between the boundaries described must be designed in Simulink. The simulation of this type of controller will be advantageous when attempting to implement the controller on the plant.

4.2. Piecewise Adaptive Controller Design

4.2.1. Switches used for the adaptive controller

A method must be devised to implement a system which uses the theory presented in the previous section. A simple device was designed in Simulink, to perform the switching between each controller. This device must read the desired magnet position, and select the controller which has been optimised for that position. Therefore it is guaranteed that the most optimal deadbeat controller is used for the desired position. Figure 4.2 illustrates the design of the switching controller.

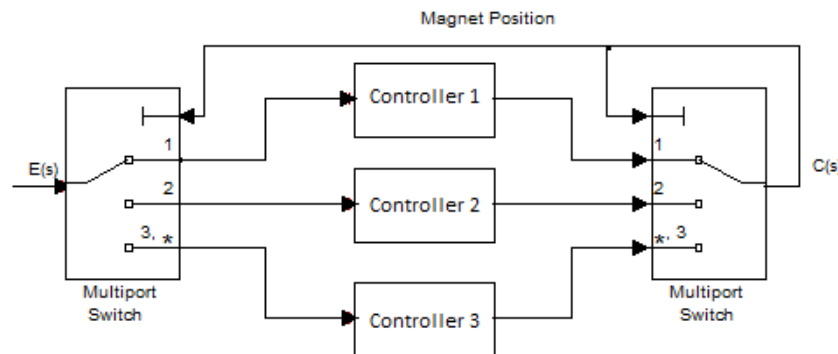


Figure 4.2: Switching technique for the deadbeat controller (representation only)

It can be seen from the above figure that the switching in and out of each controller is done at the input and output of each controller. Initial simulations showed that if only the output was switched, the integral component of each controller would still accumulate, which caused erroneous behaviour. Therefore the switch at the front of the logic was used to only allow the integral of the currently selected controller to accumulate. If a piecewise adaptive controller is used, the optimal controller for each segment could be designed and placed in the respective switching bounds. Now that a method has been devised to switch in the different controllers, the controllers themselves must be optimised. Refer to Appendix D for a capture of the resultant Simulink model. Note that the switching was performed using a Matlab script, which reduced the complexity of the Simulink file. Appendix D also shows the internal Matlab script used in the switching blocks.

4.2.2. Selection and optimisation for each controller counterpart

As per Section 3.4, numerical procedures must be used to optimise the deadbeat controller. This technique is quite time intensive, and was required to be repeated for the 0-4cm bounds at 1cm intervals. This was performed, and the following table represents the coefficients which were found to give the best result, based on the associated boundaries.

Table 4.1: Optimised deadbeat controllers for the prescribed boundaries

Boundaries (cm)	T_{desired}	K_a	X	Y
0-1*	0.054	-3.5786	79590	6.6195×10^6
1-2*	0.087	-2.4675	50218	2.6062×10^6
2-3*	0.126	-2.1019	37637	1.3732×10^6
3-4*	0.25	-2.8629	4051.6	2.1893×10^5

*=All boundaries use $K=0.01$ and $K_3=0.01$ for the optimised performance

From 4.1, it can be seen that an increase in distance directly correlates to larger T_{desired} and Y values, and smaller X parameters. This is associated with the exponential decay relationship between force and distance with the magnet. It was also seen that at larger distances, the clipping effect of the current limiter was more prominent, as the current required to hold the magnet at 4cm was nearing the limit. Therefore it was deemed impractical to operate the magnet to a point higher than 4cm due to the maximum current limitation on the plant.

4.3. Simulation Results and Performance Evaluation

4.3.1. Step response for small excursions

Once the controllers for each boundary were optimised, the entire switching controller was tested. The expected results from this testing was to have a deadbeat response for the entire control range of 0-4cm. Testing of the simulation showed excellent results, as the responses were found to be equivalent to those that were seen when operating the controllers on their own (within their respective operating ranges). In order to fairly evaluate this controller, simulation tests will be performed

for small excursions such as 1cm, and large excursions such as 2.5cm. Here the step response of a small excursion will be tested. Note that the unit step used in this testing is defined by the following equation:

Equation 4.1

$$f(x) = \begin{cases} x = 2 & t < 0 \\ x = 3 & 0 \leq t < 1 \\ x = 2 & 1 \leq t < 2 \\ x = 1 & 2 \leq t < 3 \\ x = 2 & t \geq 3 \end{cases}$$

This equation will provide a good performance indicator for movements about the point of 2cm. This will be used as a standard for all future testing, as this control signal will give good stability and steady state response times. The simulation results for the entire switched adaptive deadbeat controller are shown in Figure 4.3. Each step movement has a consistent settling time of 133ms, which is a considerable achievement. The overshoot for each position is consistent, which is excellent when compared to the response of a single deadbeat controller.

Figure 3.5 shows the same simulation using a single deadbeat controller, linearized at 2cm. It can be seen that the response becomes unstable for a 1cm movement. Figure 4.4 shows that the switched controller eliminates this effect, and provides a smooth response with a fast settling time. One issue seen with these graphs is the fact that the system still overshoots the desired position. This overshoot is the same as the PID control overshoot, as detailed later on. It was found that by choosing different constants, the overshoot could not be removed. It was assumed that if the integral was removed, for large displacements, the overshoot could be removed. This would be grounds for further work.

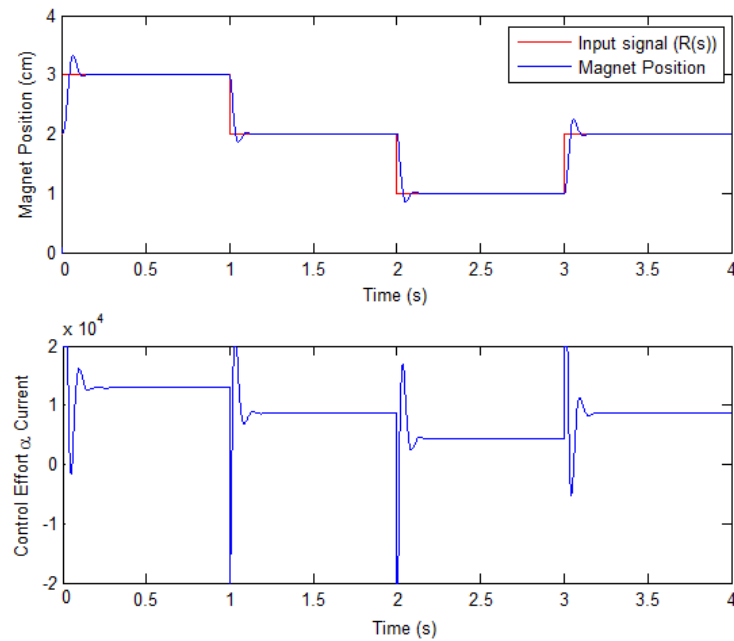


Figure 4.3: Switched adaptive deadbeat controller, small movement response

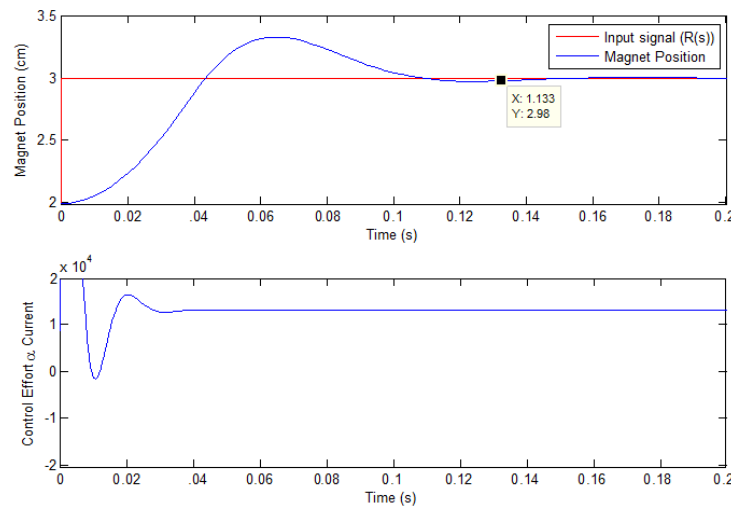


Figure 4.4: magnification of the step response, 1cm movement

The figures above also illustrate the effect of the coil current limiter. If this limiter was not in place, analysis shows that the coil would be overloaded and the machine protection would shut the plant down. Also note that when the steady state is reached at 3cm, the coil current is close to the limit, reinforcing the decision to not design a controller that would work past 4cm.

4.3.2. Step response for large excursions

Now that the step response has been seen for small excursions about a linearization point, the deadbeat controller must be tested for a large excursion. Due to limitations of the plant software, the unit step is limited to an excursion of 2.5cm. Therefore all testing will be limited to a maximum excursion of 2.5cm. Figure 4.5 shows the output of the simulation for an excursion of this size. It can be seen that the response is almost identical to that seen on the smaller excursions. The large excursion yielded a smaller settling time, as 2% of 1cm is much smaller than 2% of 2.5cm. Overall, the settling times for any magnet movements were consistent, as detailed in Chapter 7. The results show that the switching controller has greatly improved the response of the controller over the entire operation range. These responses will be compared with the PID responses in Chapter 7.

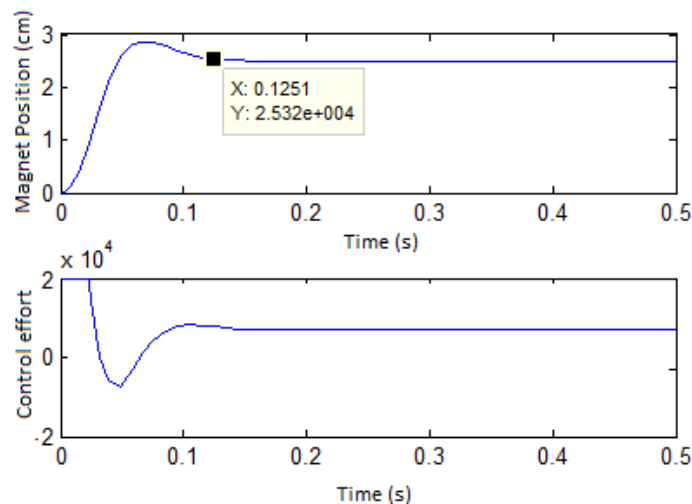


Figure 4.5: Simulation results for a large excursion

This concludes the simulation of the switched adaptive deadbeat controller. The results shown in this section will be different to the results on the plant, as there are many factors which will hinder the operation of the magnet. The simulation is done assuming that every component is perfect in the plant and the controller is continuous, therefore does not exhibit discrete time steps. There may also be discrepancies between the nonlinearities in the plant and the simulated plant. Although there may be differences between the simulator and the plant, the simulation does give a solid foundation for variable selection. This simulation will

reduce the risk of unstable magnet movements, as the large instabilities have been removed.

Chapter 5

Deadbeat Controller Implementation

5.1. Programming the Maglev System

5.1.1. C programming language

Now that the control theory has been tested and proven viable in simulation, the plant must now be programmed in an attempt to reproduce these results. The basics of the Maglev plant have been outlined in Chapter 2, where the free body diagram was observed. As described, the Maglev machine has real-time software which is run on a digital signal processing chip. For interest this chip is a Motorola M56000 chip running at 40MHz, and uses 24-bit instructions. The host computer which harbours the DSP board is loaded with ECP-written programming software, which provides the ability to program raw instructions into the DSP chip. The programming language used is a limited variation of C code, where higher-level functions are not

supported. This programming interface is excellent for the digital implementation of the controller, as it allows the easy implementation of DSP instructions.

The basic structure of the C-code programming must be understood in order to implement the robust deadbeat controller to a real-world plant. The Maglev machine has 100 32-bit floating point variables, which must be assigned in an initialisation process. This memory mapping is handled by the compiler, and does not have a part in the real-time algorithm. After the variables have been initialised by using the `#define` instruction, the constants must be allocated to their predefined values. In this case, the X, Y and other variables must be allocated. Once again this is handled by the compiler, which reduces the computation overheads in the real time logic execution. The real-time program is then written between the 'begin' and 'end' instructions. Each sampling period, the instructions inside this loop are executed once. Therefore the real-time script is executed at 1.1 kHz.

5.1.2. Programming the robust deadbeat controller

In order to have the real-world Maglev plant operate the magnet correctly based on the designed controller, the plant must be programmed using discrete control theory. Each Simulink block was converted into code, and sequentially added to the Maglev programming script. The optimal constants which were found using the simulation were then passed into the plant control algorithm, as it was assumed that these parameters would give a desirable output from the plant.

A major assumption here is that the control system for the plant is continuous. When comparing the sampling rate to the speed of the magnet, aliasing and other digital signal processing issues will not be present. That is, the sampling rate is much greater than the Nyquist rate. Therefore the effect of implementing the continuous logic onto the discrete controller will not affect the operation of the controller. This is reflected by the results shown in Chapter 7.

Another assumption made is that the backward derivative method will be adequate for the calculation of the derivative in real-time. This is the only possible method, as the system is real time and future signals are unknown. This is similar to the integration method, where the current position and previous position are used. The trapezoidal method is used for the integral, where the average is calculated from the

current signal and the last signal. This value is then multiplied by the time difference, in order to find the area. This is added to the previous accumulation of the integral. A representation of these errors is shown in Figure 5.1.

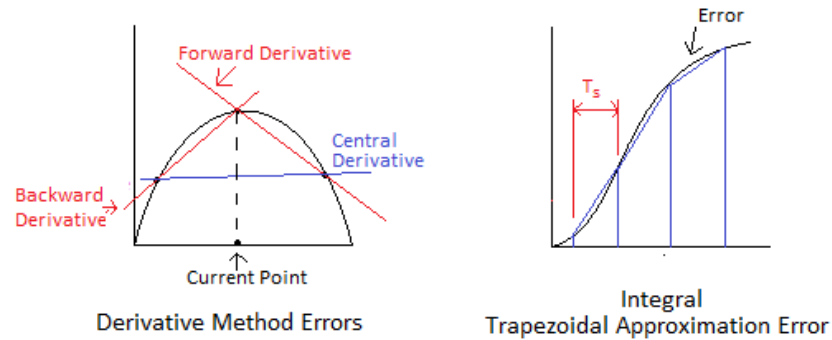


Figure 5.1: Representation of derivative and integral approximation errors

One of the challenges posed here is the requirement to convert simple function blocks (integrals, derivatives, state-variable feedback etc.) into raw code which can be executed on the Maglev machine. The Maglev machine executes the code at a frequency of 1.1 kHz, which means that in the discrete sense, the sampling period T_s is 0.9ms. Therefore, the code written must not have a larger execution time than the sampling time, or the code execution will fail. The following information explains the implementation of the controller in C code, and the complete code is shown in Appendix C. Figure 5.2 shows a representation of the controller to be coded, with key points to be used in the C coding.

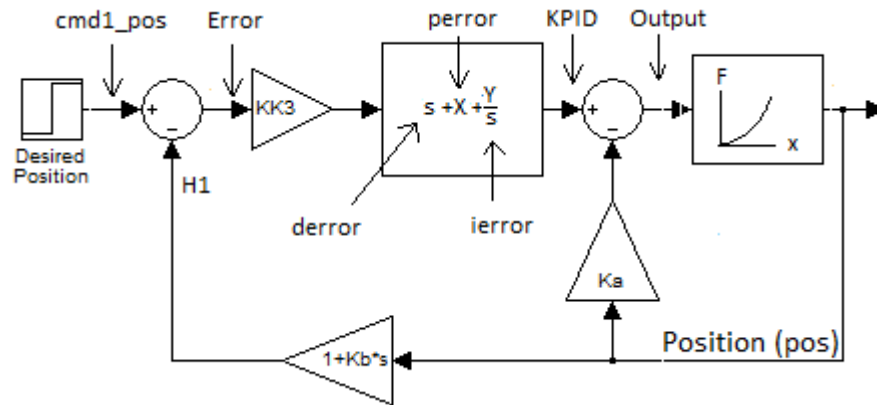


Figure 5.2: Key points used for C-code generation

Now that these points have been indicated, they must be sequentially calculated using the C-code for each sampling cycle. The following table shows how this will be done.

Table 5.1: Code used to build the controller in C

Point:	Formula:	Explanation:
H1	$H1 = Kb \frac{\text{currentpos} - \text{lastpos}}{Ts} + \text{currentpos}$	S is a derivative in the time domain. Therefore, rise over run plus the original position is H1
Error	$\text{Error} = \text{cmd1_pos} - H1$	Simply an error calculation
Perror	$\text{perror} = X \times \text{error}$	Value is proportional to the error
Ierror	$\text{ierror} = \frac{\text{error} + \text{lasterror}}{2} \times Ts + Y + \text{lastint}$	Trapezoidal integration method. Adds to the previous integral value
Derror	$\text{Derror} = \frac{\text{error} - \text{lasterror}}{Ts}$	Change in error per time unit. Backward differential method
KPID	$\text{KPID} = (\text{perror} + \text{ierror} + \text{derror}) \times K \times K_3$	Adds P, I, D control mechanisms.
Output	$\text{KPID} - K_a \times \text{currentpos}$	State variable feedback

Note that at the end of the code, the current error is transferred to the last error, in order to provide a current and previous error. Refer to Appendix C for the code listing. All of the functions in the above table have been written in C-code, where the

operations were performed sequentially on a line-by-line basis. This concludes the design of the Maglev control algorithm. Testing was then performed on this algorithm, with the results shown in the next section.

5.2. Maglev Plant Initial Implementation Problems

The initial implantation of the controller on the plant using the optimised simulation values failed. The magnet was found to vibrate about the control point, with movements of up to $\pm 1.5\text{cm}$. It was found that the plant controller was critically stable, as the oscillation amplitudes were consistent over time. This response was anticipated, as the plant is subjected to many different environmental and machine issues which could not be simulated. Diagnostics were undertaken to identify the issue, and solutions to obtain a better response were sought.

For a graphical representation of the response, a method was required to export the data from the Maglev machine to a program which can generate an image of this data. The ECP software has a facility to generate figures, however was not sufficient for the comparison of the simulated results and the plant results. The ECP program provided a facility which outputted a raw data text file, which contained all of the data which was read in real-time from the Maglev plant.

A Matlab script was written (Shown in Appendix B) which could read this data and plot the response of the Maglev plant. As the Maglev plant has four user-assigned data acquisition arrays, as well as four controller specific arrays, plots can be generated from any point in the controller file. This allows for the simultaneous plots of the sensor information and the integral information, which gives the ability to troubleshoot key points on the plant. Therefore all of the Maglev machine responses have been exported from the ECP program and plotted using Matlab. The following problems were diagnosed, and solutions to these problems are stated.

5.2.1. Sensor noise

Diagnostics initially showed a large amount of noise present on the sensor reading, which appeared to be quite intrusive when the differential in the feedback zero was calculated. To test the noise on the sensor, code was written to disable the coil current and simply read the sensor data. This script then stored the data, in raw data

bites, and Matlab was used to plot the sensor noise. Note that the magnet is resting on the bottom stopper, just above the coil. The blue line in Figure 5.3 shows the noise which is seen on the sensor. When the magnet is at 2cm, the actual position reading is 2 ± 0.04 cm.

From this figure, the signal noise appears to be a repeating sinusoidal-like function at 195Hz. This frequency appears to be close to a fourth harmonic of the 50Hz mains power supply (200Hz). Therefore it is assumed that this interference is electrically coupled from supply harmonics through the Maglev power supply and into the Maglev circuitry. This interference could also be coupled magnetically.

This noise is part of the Maglev machine hardware. Therefore the only solution available is to implement a filter in the real-time algorithm. This filter must rely on previous sensor data, which is logged and updated each real-time servo cycle (at 1.1kHz). The quickest solution to this issue was to use a finite impulse response (FIR) filter. A finite impulse filter satisfies this criterion, and does not use the control algorithm output data in the way that an infinite impulse response (IIR) filter does. The FIR filter was implemented using a Hamming window (not explained here) for its square-like filter response. As the interference is at 195Hz, the filter was designed in a low-pass configuration, with a cut-off frequency at 100Hz. This cut-off frequency was deemed to be suitable as it will not filter the actual movement of the magnet. The following FIR filter terms were used based on the Hamming window:

$$C(z) = 0.0264z + 0.1405z^{-1} + 0.3331z^{-2} + \dots + 0.3331z^{-3} + 0.1405z^{-4} \\ + 0.0264z^{-5}$$

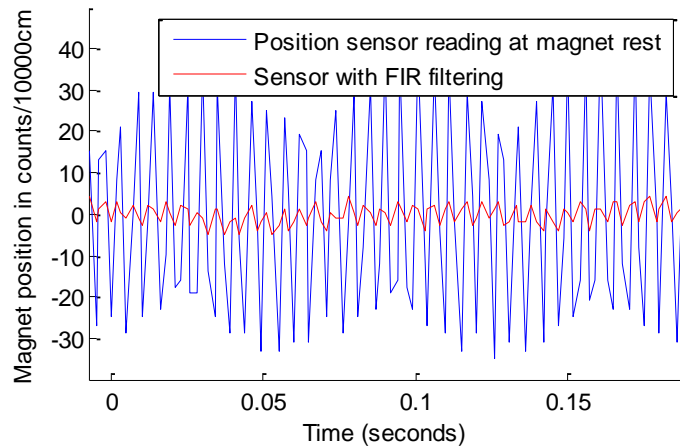


Figure 5.3: Sensor noise, and FIR filtering response

Figure 5.3 illustrates the effect of the FIR filter, where the output was reduced 10-fold. This will largely reduce the effect of the sensor noise on the derivative component in the controller. The results of the implementation of this controller showed that the magnet oscillations were only slightly reduced, by a magnitude of about 10%. Therefore it was concluded that the robust deadbeat controller was already able to reject the sensor noise, and that the sensor noise was not a major contributing factor to the magnet's oscillations.

5.2.2. Vibration Issue

The high-frequency oscillations seen on the plant were still present and were causing the plant to operate uncontrollably. A diagnosis method applied was to plot the Maglev's calculated values at the key points described in Figure 5.2, and compare them to the same points on the simulator. The Maglev coding appeared to be operating correctly, however more analysis showed that the integral accumulation was at a very large rate. This rate caused the plant to put a large amount of current into the coil, and when the magnet overshoot the control point, the same current was spiked in the negative direction, causing oscillations.

The solution posed to this issue was to reduce the T_{desired} term in a manner that reduced the integral term Y . It was seen in the Matlab script that the T_{desired} term was related to the integral term, Y . It was found that if the T_{desired} term was increased (i.e. the expected settling time was relaxed), the system stability was increased. Further

analysis showed that T_{desired} values which caused the integral component Y to be less than 3×10^5 caused the system to become more stable. Therefore all of the simulated deadbeat controllers were re-calculated to have Y values of around this point. The response of the system was moderately better than that of the system with a large integral value, however still showed oscillations which caused the magnet to deviate past the 2% settling time bound. Analysis showed that none of the possible T_{desired} values could stop this oscillation, therefore the possibility of another problem was considered.

5.2.3. Thermal compensation

The oscillation of the magnet about the desired position was reduced, however was still a major issue, as the steady-state point was never found. Diagnostics performed on the code operation did not show any errors; therefore it was assumed that a software problem was causing the oscillations. When the Maglev manual was consulted, it was found that the machine had a function that compensated for the changes in the sensor temperature. This function alters the current in the coil based on the temperature of the laser position sensor. According to the Maglev documentation, the thermal compensation creates a considerable delay between the time the current was applied and the time it was amplified by the IO unit (Educational Control Products 2002).

By removing this thermal compensation, the plant appeared to operate in a stable manner, where the magnet would rest exactly on the desired output position. Therefore, it is assumed that the delay caused by the compensation was too large for a controller of this precision. These oscillations exhibited the same characteristics as that which would be seen on a linear system with a large pure time delay. In any case, the removal of the thermal compensation caused the plant to operate as expected by the performance criteria. Now that the problems with the plant have been ironed out, the performance evaluation and optimisation was conducted.

5.3. Results and Performance Optimisation

When the single deadbeat controller was applied to the plant, it was quickly seen that the switching controller was needed. Analysis showed a large effect from the magnet

moving away from the linearization point, using the single deadbeat controller. As stated before, the effect of moving away from the linearization point with the single deadbeat controller was more significant when compared to the simulated controller. The following figure demonstrates this effect.

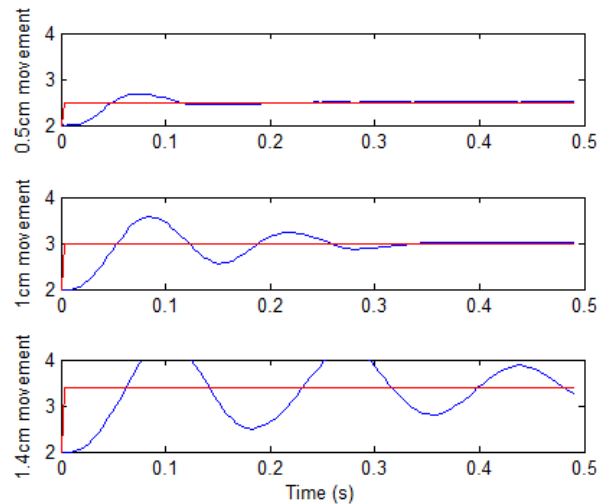


Figure 5.4: Effect of moving away from linearization point

It can be seen from Figure 5.4 that the effect of moving away from the linearization point of 2cm was quite significant. From this figure, it can be seen that deviating for more than 0.5cm from the linearization point caused oscillations, which were more significant as the distance increased. It was deemed that the single deadbeat controller on the plant was only sufficient for a movement of 0.5cm, which was smaller than the simulated result of 1cm. A movement of 1.5cm away from the linearization point caused a critically stable oscillation, where the oscillation magnitude did not change over time. Therefore the single deadbeat controller exhibited excellent response times for a small range, and must be improved if it is to operate across the entire magnet range. The following figure shows how the magnet responded to a ± 1 cm movement.

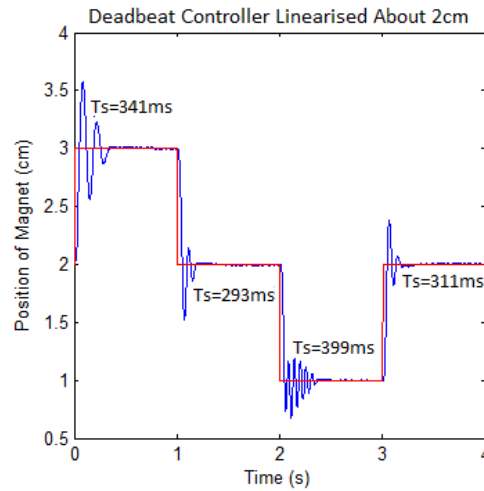


Figure 5.5: Linearization error from small movements

It is seen that the frequency of oscillations is increased for magnet positions closer to the coil (the bottom). This is attributed to the higher force relationship as the magnet moves closer to the coil. This results in a larger force being applied to the magnet, and causes overshoot and instability in the controller. The need for a switched adaptive controller has now been established, and the following chapter shows how a better response is achieved.

Chapter 6

Adaptive Controller Implementation

6.1. Plant Programming for the Adaptive Controller

As shown before, the effect of moving away from the linearization point in the plant is significant. Therefore the switching method defined before must now be applied to the Maglev machine, in order to make the response constant for any position. It has been proved that the performance of the single deadbeat controller is excellent for a small range of positions. Therefore as the switching controller was simulated with success, it was deemed possible for the switching controller to be applied to the plant. As the programming of the plant is in C code, the switching controller in Simulink must be converted to this syntax. If successful, the plant will exhibit a deadbeat response for all magnet positions. This will prove that the deadbeat controller can be manipulated to control a nonlinear system.

The switching controller solution shown in Chapter 4 was manipulated for implementation on the plant. The switching technique used to change between controllers in the Simulink model was handled by two simple logic blocks

(Appendix D). A posed solution to the switched plant control was to use ‘IF’ statements, which can be manipulated to act as a switching block. A disadvantage to this instruction is that although the computational time is almost the same (an IF statement acts as a simple branch command), the amount of instructions is a multiple of the amount of controllers, which could become a problem if memory space is an issue. Advanced process management and programming flow control could be used to reduce the memory space usage, however was not employed as the Maglev plant was able to handle the large programming file.

The application of the control system to the plant was made possible with multiple IF statements. It is able to test for a prescribed bound, and execute the relative code for that bound. The following segment of code will show how the IF statement will be applied to the plant. (This code is written in Matlab, which has similar IF syntax to the programming language on the plant)

```
if position=>0cm && position<1cm
%put controller code for this bound here
else if position=>1cm && position<2cm
%put controller code for this bound here
else if ...etc
```

From the above code, the program tests for each desired position, and when the position is inside the bounds, the code relative to that bound is executed. The first step was to design the code for only two bounds, and work more bounds in once the switching method was proven to work. Initial implementation showed that the issue found in the simulation carried over to the plant. This issue was the integral accumulation, where the plant behaved inappropriately when changing from one controller to another. A solution to this problem was simply to add another IF statement which tests for a controller change, and resets the integral at that point. Therefore, the system will not behave inappropriately when the plant changes controllers.

6.2. Numerical Performance Optimisation for all Sections

Once the system was capable of changing between controllers, the optimised parameters of each switched boundary were to be found. These parameters will be optimised in the same way as the procedure described in Section 5.3. In order to reduce the complexity of the plant optimisation, the single deadbeat controller

algorithm was used. This code was linearized about the midpoint of each boundary. As described in the previous Chapter, the plant behaved in a different manner to the simulation. The T_{desired} parameter was far too small for the plant to operate correctly. Therefore, this constant was reduced, and altered until the most optimal response was found. This was done for each controller, until all boundaries exhibited the same deadbeat response.

Results showed a stable response could be found for any linearization point, however if the plant deviated by 0.5cm, it became unstable. This is not reflected by the simulation, however is explainable by the differences between the Simulation and the real-world plant. Therefore a decision was made to half the distance between the switching boundaries, effectively doubling the amount of controllers. Therefore the simulation was able to handle 1cm boundaries, and it was found that the plant must use 0.5cm boundaries for an effective and repeatable output. When using the smaller boundaries, the response of the plant was repeatable for any magnet position, and no oscillations were present. The following table shows the optimised parameters for each boundary.

Table 6.1: Boundary optimisation parameters

Boundary	T_{desired}	K_a	X	Y
$0 \leq x \leq 0.5^*$	0.13	-0.5477	8569.8	304250
$0.5 \leq x \leq 1^*$	0.16	-0.5133	6909	201600
$1 \leq x \leq 1.5^*$	0.14	-0.6682	10197	337230
$2 \leq x \leq 2.5^*$	0.15	-0.6780	9902.5	306790
$2.5 \leq x \leq 3^*$	0.14	-0.7396	11408	377160
$3 \leq x \leq 3.5^*$	0.16	-0.8725	12550	365710
$3.5 \leq x \leq 4^*$	0.16	-0.9213	11364	381370

*=All boundaries use the $K=0.01$, and $K_3=0.01$ values, for the optimal response

From the optimised parameters, there is no logical progression of the T_{desired} parameter. The optimisation showed that those parameters were the most ideal. An explanation for this is that there is a nonlinear static friction-slip characteristic plant, which caused the responses to vary. One notable progression is that generally, the X parameter is increasing slowly and the K_a parameter progresses almost linearly. This

shows that for greater distances away from the coil, the state variable feedback is increased for better stability. Note that the same issue found in the single deadbeat controller was present here, as the Y parameter must be kept low to ensure that no oscillations were present. This range was found to be around 3×10^5 , and increased slightly for larger displacements. The following figure illustrates the performance gain over the standard deadbeat controller.

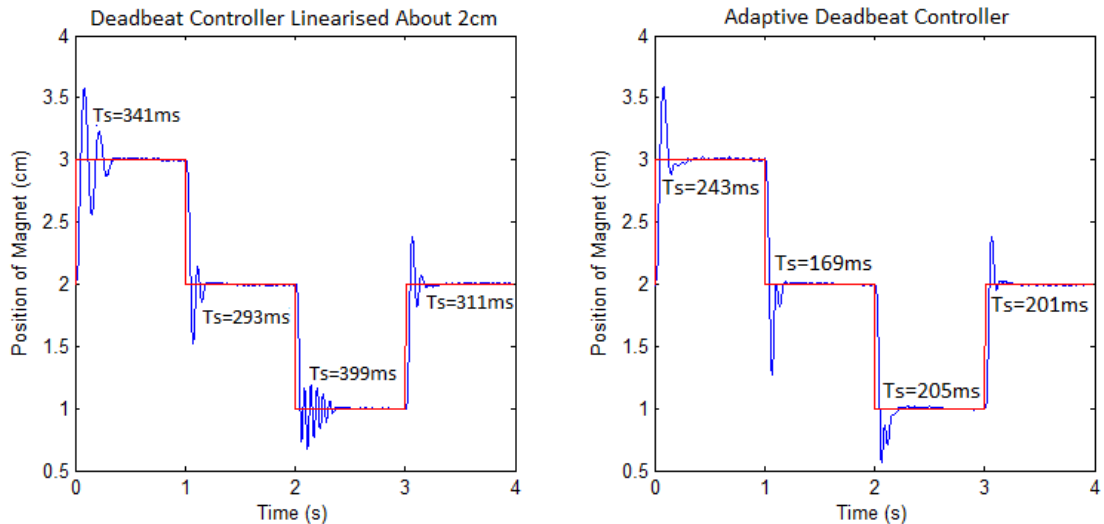


Figure 6.1: Comparison of single and switched deadbeat control results on the plant

Figure 6.1 illustrates the effect of using a switched controller rather than a single controller, for larger movements. It can be seen that the oscillations present from deviating from the desired position have been nulled. The response shown on the right hand graph is quite fast, and will be compared to the PID control response in Chapter 7. This figure still shows a large amount of control point overshoot, which is quickly rectified. The oscillations seen on the single deadbeat controller are more prominent as the distance from the linearization point is increased. It can be seen that for any magnet movements, the adaptive deadbeat controller response is superior to the single controller. This response is consistent for any magnet position, and proves that the switching controller was an effective solution to the nonlinear error.

6.3. Boundary Performance Optimisation for Large Excursions

It has been seen that for small excursions, the plant with the switched adaptive controller is responsive. The response shown in Figure 6.1 is consistent for movements greater than 1cm. However, for a large movement of 2.5cm, it can be

seen that there is a rather large overshoot and correctional undershoot in the response. Therefore the need to reduce this violent response was required. It was theorised that the violent response was caused by the high-speed integral accumulation (a large Y value), which caused large overcorrecting forces. A method was trialled in an attempt to optimise the large response settling time of the plant.

Through trialling, it was found that for movements of greater than 1cm, the system response was more violent. The solution to this issue was a simple 'IF' statement which halted the integral accumulation for movements of greater than 1cm. The result for doing this was that the system response was damped, and settled in the same time as the response with the integral component. As the control effort for the large displacement was quite small, another solution was trialled where the coil current was set to maximum for a displacement of above 1cm. The response of this trial was excellent, and is shown in the following figure.

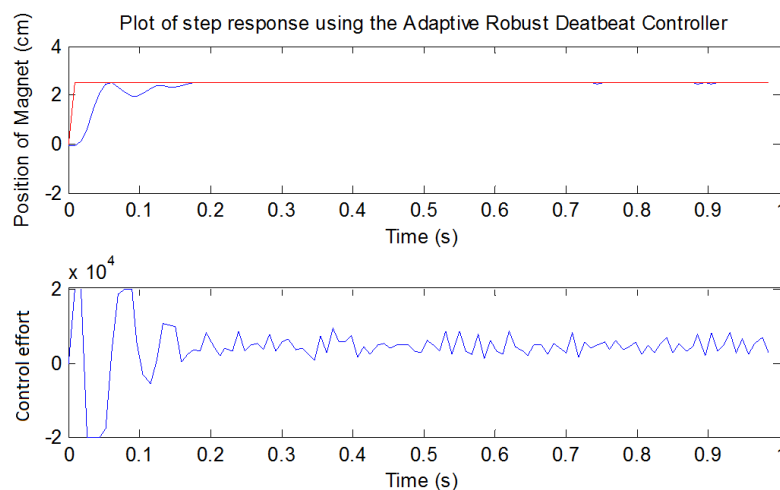


Figure 6.2: Step response for large magnet movements

The above figure is indicative of the response for a large magnet movement. Note that this response is consistent for any large magnet movement. Now that the plant has been optimised to the best possible response, a comparison can be made between the switched adaptive deadbeat controller and the PID controller.

Chapter 7

Experiment Results and Evaluation

7.1. Simulation and Plant Comparison

One of the notable problems between the simulation and the plant was the fact that the plant required a larger T_{desired} parameter to operate correctly. It was assumed that the performance of the plant would be degraded by using this larger parameter. However, visual analysis shows that the plant results are similar to the simulator. Figure 7.1 shows the difference between the final simulated switched deadbeat controller and the response from the plant controller.

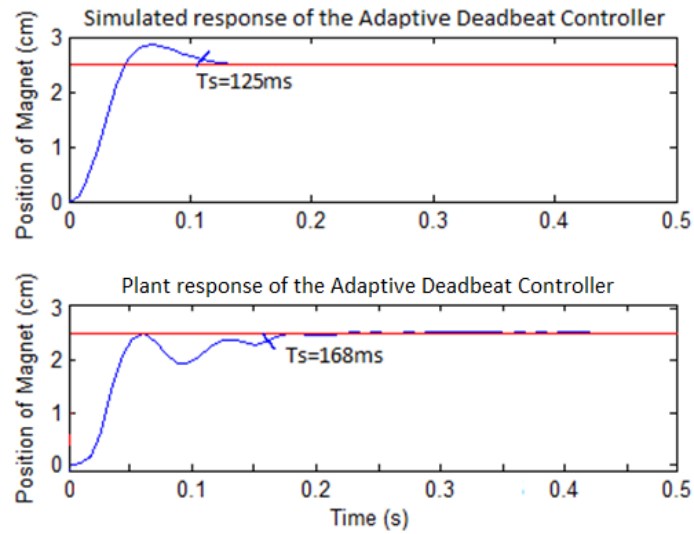


Figure 7.1: Comparison between the simulation and plant responses

Note that from the above figure, the plant behaves differently in regards to the rise pattern. This is due to the fact that the integral accumulation is halted for displacements of larger than 1cm on the plant, therefore there was no overshoot. The simulator does not have this feature as it did not exhibit violent oscillations in the large movement, as seen in Figure 7.1. Even with the halted integral accumulation, oscillations are still seen on the rise of the deadbeat controller. This could be removed by reducing the T_{desired} parameter further, however the settling time will be increased. Also note that the rate of change of the two graphs are similar (even though the displacements are different) which indicates the similarities between the plant and the simulator.

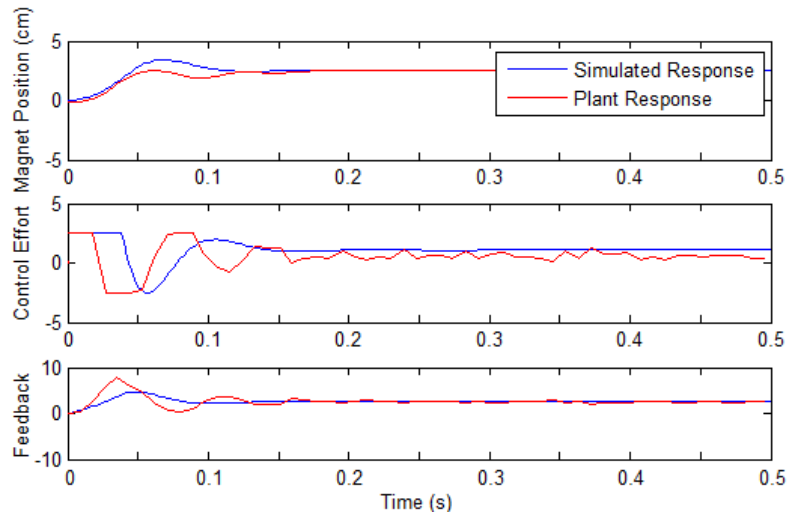


Figure 7.2: Comparison of key control points

Figure 7.2 shows a comparison between key points in the controller, for the simulation and the plant. These key points have been captured and plotted using functions inside the Maglev plant. The key points have been represented by the block diagram of the deadbeat controller, shown in Figure 5.2. Analysis of the above figure shows that the plant is similar to the simulated system. The magnet position is moved to the same point (with a slight difference in rise pattern), and the feedback components are similar in nature when compared to the magnet position. The simulation and plant's control effort outputs appear to be marginally different. The transient response of the figure (where the unit step has occurred) is different due to the fact that the integral component is removed from the plant for large movements.

Also note that there is a discrepancy in the steady state control efforts for each controller. This is explained by small differences between the simulated plant and the actual plant, where the Taylor's linearization theorem is not a perfect representation of the plant. There are slight oscillations present in the response of the plant with the feedback point, which is attributed to the high derivative gain on the sensor noise, explained more in Section 5.2.1. Now that these key points have been explored, the following figure uses applies a PID controller to the feedback signal.

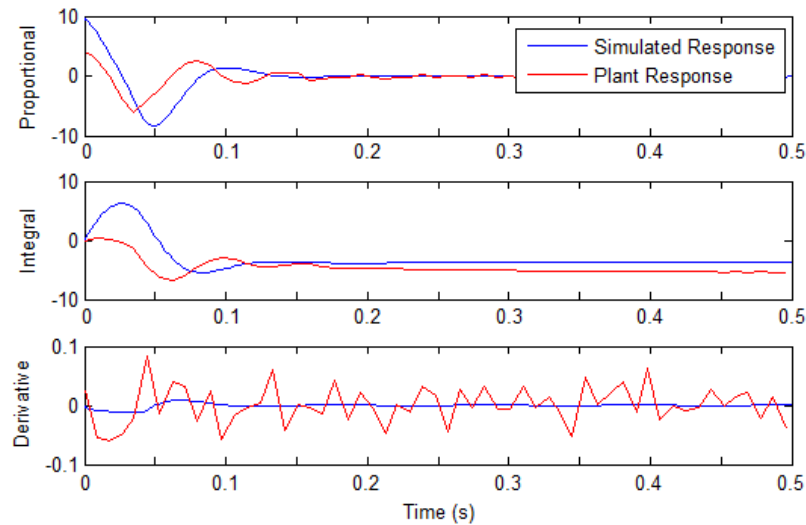


Figure 7.3: Comparison of key PID points (point-by-point related to the previous figure)

Once again, the key points shown in the above figure show that the plant behaves much like the controller does. It can be seen that as the plant reaches steady state, the proportional gain reaches zero, as is expected. It can be seen from the integral trend that the integral component moves toward the value which causes the magnet to rest at the 2.5cm mark. Note that the integral start is different, as the integral for the plant does not accumulate until the magnet is within 1cm of the desired position. There is also a difference between the steady state of the integral components, which is a small error between the Taylor's theorem and actual plant transfer function.

The last graph in Figure 7.3 shows the derivative component in the feed-forward loop. Note that this derivative again amplifies the noise seen in the sensor on the plant, resulting in major oscillations when compared to the simulated response. This however, does not affect the operation of the plant, as the proportional and integral gains far outweigh the derivative gains. This is found by looking at the magnitude of the PID components in Figure 7.3. Now that a comparison between the simulated deadbeat controller and the real deadbeat controller has been made, the plant can be compared with the industry standard PID controller.

7.2. PID performance analysis

Educational Control Products have provided a PID controller with their Maglev plant. The experiments which ECP prescribed for the Maglev plant show how a PID

controller is designed to exhibit the best response possible. The optimised parameters for the best response from the PID controller have also been supplied. As a PID controller is the most common controller used in the industry (Nise, N 2011), a decision was made to compare the designed controller with a PID controller. This will provide an indicative result of the viability of this controller in the real world. The PID controller written by ECP uses the same techniques used in the deadbeat controller for the Integral and Derivative action. This action was done in the discrete time domain; however the integral was a current estimation integral, rather than the trapezoidal method used in the deadbeat controller. It was found that the trapezoidal method was more precise. A PID controller is simple to implement on a plant, where the proportional, integral and derivative gains are tuned to get the best response.

7.2.1. Performance results for small excursions

ECP have already demonstrated the PID parameters which give the best response from the Maglev machine. In order to compare the plant's PID and deadbeat controllers, the same small movement and large movement steps used in the previous tests must be used. These steps are 2.5cm for a large magnet movement, and 1cm for a small movement. The following figure shows the response of the system.

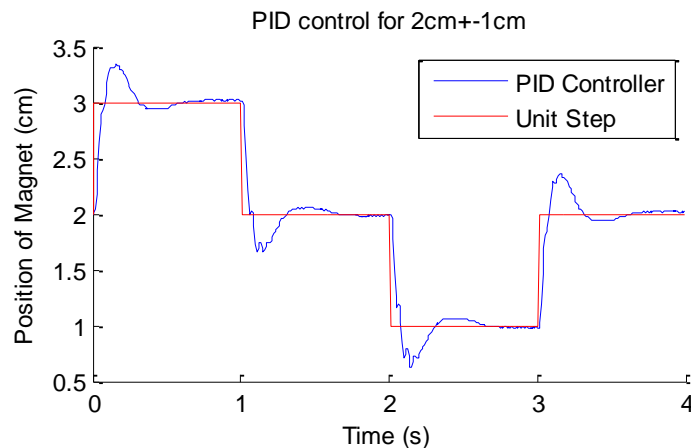


Figure 7.4: PID Maglev plant controller, small step response

The critically damped response above is claimed to give (by ECP) the quickest settling time. PID parameter optimisation showed that this claim was true. Therefore a performance benchmark has been established for small magnet movements which the robust deadbeat controller will be tested against.

7.2.2. Performance results for large excursions

Now the large excursion results for the PID controller must be examined. The following figure shows a critically damped case for the large unit step.

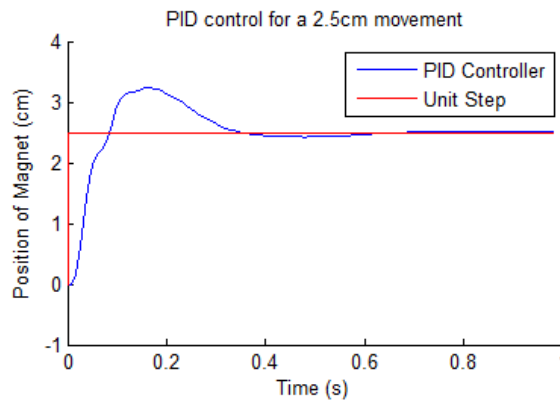


Figure 7.5: PID best response

This figure shows the best response for a large unit step, as prescribed by ECP. This will give a benchmark which the deadbeat controller will be compared against. Now that a comparable controller has been tested, the results from the final switched deadbeat controller can be presented and analysed.

7.3. Performance Analysis of the Final Controller

The viability of the robust deadbeat controller can be judged by comparing its performance with a PID controller. The following criteria/control systems testing will be used in determining the performance of these controllers:

- Settling Time – The time in which the system takes to reach 2% of the desired position
- Stability – Checking for unwanted oscillations in the output
- Overshoot – How far the magnet goes past the desired position
- Controllability – The ability to operate the magnet to any desired position
- Consistency – The settling time and stability must be consistent for any magnet position

Other criteria which will be tested and compared between the PID and the robust deadbeat controller will be:

- Frequency domain analysis – The system’s ability to respond to different frequencies
- Disturbance rejection – The ability to control the system with a disturbance

This will provide a solid basis in comparing this controller to a mainstream controller used in the industry.

7.3.1. Small Magnet Displacements

The controller must be able to accept small position movements with speed and accuracy. Due to the nature of the controller switching, there may be transients present when a different controller is selected. The switched boundaries for the deadbeat controller are at 0.5cm intervals; therefore a ± 1 cm unit step will be applied to the magnet. This movement will be about the 2cm point, where the performance of 3 of the switched controllers will be tested. The following figure shows the response of the controller, as well as the best response obtainable from the PID controller.

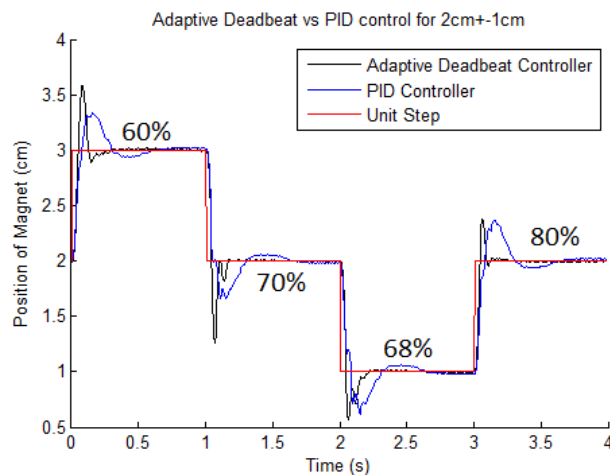


Figure 7.6: Step response comparison for small magnet movements

Figure 7.6 shows two different responses. The red line is the input to the system (which is applied to both the switched adaptive deadbeat controller and the PID controller). This line is about 2cm, and deviates by ± 1 cm, which shows a good range of responses. This provides a good test for the controllers, as it causes the switched deadbeat controller to switch between controllers. The blue line indicates the response of the best PID controller, as shown by ECP. This response is critically

damped. It can be seen that there is a large overshoot, which is present for a significant amount of time. The response here is quite slow, as the controller does take time for correction. The settling time is 550ms on average, and the stability of the system is quite good as there are no significant oscillations present.

The above figure also shows the response of the plant when the adaptive deadbeat controller is applied (black line). It can be seen that the response of this controller is superior to the response of the PID controller, for small magnet displacements. The percentage reduction in settling time is shown in the figure, where the worst case is 60%. Further testing found that the worst case settling time of this controller is 60% better than the PID controller. By analysing the figure, it can be seen that the response of the adaptive controller is consistent for different operation boundaries. There is an overshoot of the magnet, then a quick correction back to the desired position.

The overshoot of the switched deadbeat controller in the Figure 7.6 is still significant, and in some cases worse than the PID controller. This could be an issue when applying this control theory to a plant in the industry. Attempts were made to reduce this overshoot; however a reduction in overshoot would cause performance degradation. It is also noted that the rise time of the magnet is quite violent, which could be an issue in a system where the force applied to the magnet must be minimised. A feed-forward derivative component could help slow the response of the magnet; however would add more complexity to the system. This solution was not designed due to the complexity of the system. The last performance criterion, stability, is also satisfied. It can be seen that once the magnet settles from the overshoot, no further oscillations are present. From these tests, the viability of the switched deadbeat controller is seen, as the performance in the tests was excellent.

7.3.2. Large magnet displacements

As stated before, it was found that the plant would operate inappropriately when faced with a large magnet displacement. This was due to the accumulated integral found from the large instantaneous error signal, and caused the magnet to overshoot and oscillate quite violently. A separate controller was added, to force the plant to halt the integral accumulation for large movements. This proved to give the best

response. Therefore it is imperative that this control technique is tested in the same way as the small movements section above. The following figure shows the magnet performance for a step displacement of 2.5cm.

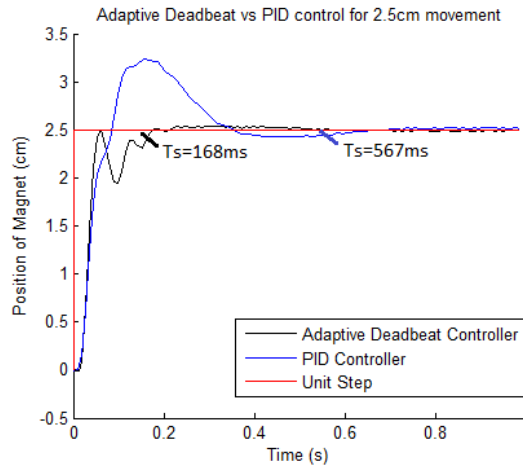


Figure 7.7: Large step response comparison

The figure above again shows the response of both the PID and deadbeat controllers. The PID controller once again exhibits the same second-order PID response, which is critically damped. The overshoot is present for a long time, and the overcorrection is significant. Note that there is noise on the position signal, and this noise is consistent for both controllers. The stability of this controller is quite good, as a steady state is reached in 567ms.

Figure 7.7 also shows the response of the deadbeat controller. The effect of the lack of integral accumulation is quite significant here, when comparing this response to that in Figure 7.6. There is no overshoot, and the system acts more like a damped system. Oscillations are still seen on the rise of the magnet; however the magnet does not overshoot the desired position. This is a desirable response, as the magnet does not exceed the required position. The result from this image is also repeatable for any large displacement; however the software for the plant is limited by a movement of 2.5cm. Note that the response time is 70% better than the PID controller.

7.3.3. Disturbance effect on the robust system

Relating back to the core applications of Maglev, it can be seen that disturbances in a Maglev train will be present. Changes in the mass of the train, cornering and higher-speed down forces will cause the forces to be applied to the train to change in time. This is known as a Time-Variant System (TVS), as the transfer function which governs how the machine will act changes over time (Nise, 2011). Here these disturbances will be modelled by changing the weight of the magnet and observing the difference in the control system response. According to the authors of the document which poses the ‘Robust Deadbeat Controller’, this controller is able to reject up to a 50% variation in the transfer function parameters (Dawes, Ng, Dorf & Tam 1994). As the force-distance relationship transfer function has a high dependency on the magnet weight, it was deemed that by increasing the weight of the magnet, the transfer function of the system will change.

A test was made where an even weight was added to the magnet, in order to cause a ‘disturbance’. This could be directly associated with a Maglev train being filled with passengers. This was simulated by adding an 80g uniform weight to the 120g magnet. The following equation shows the change in the transfer function, for a linearization point about 2cm.

Equation 7.1

$$G_P(s)_{120g} = \frac{1111}{s^2 + 4s + 478.5}$$

$$G_P(s)_{200g} = \frac{669}{s^2 + 4s + 478.5}$$

Therefore the addition of an 80g mass causes a large change in the transfer function. The following figure shows the effect of adding more weight on the plant controllers. The addition of the weight was performed in a safe manner, and did not breach any safety guidelines outlined in the risk assessment.

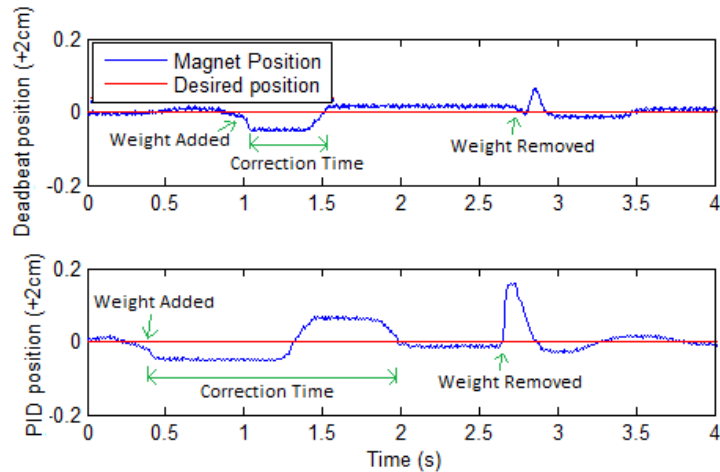


Figure 7.8: Effect of a disturbance on the controllers

Figure 7.8 shows the responses from the PID and deadbeat controllers. Note that direct comparison by time is not possible, as the addition and removal of the weights has human error on the time domain. However a comparison between the magnet responses can still be made. The PID controller response time is three times that of the deadbeat controller. The PID controller overshoots the desired position in an attempt to minimise the error, and dwells on this point for further time. The large point of interest here is the anomaly that occurs when the weight was removed, as the transfer function is instantly altered, testing out the robust properties of each controller. The PID controller had an anomaly of a 2.5 times larger magnitude over the deadbeat controller when the magnet was removed. It is seen that the switched deadbeat controller exhibited a better response, therefore has a better ability to reject a system disturbance. Therefore it was deemed that the deadbeat controller operated the plant in a superior fashion when compared to the PID controller. This reinforces the title ‘Robust Deadbeat Controller’.

7.3.4. Frequency domain response

Another performance testing method which gives a good idea of the system is a frequency domain test. This will test the ability of the system to respond to a range of sinusoidal inputs. A frequency domain analysis will show the gain of the controller over a range of frequency inputs. Another result of this analysis will be a phase diagram, which will show how much the phase varies when the frequency of the input is altered. The results of this analysis will show any resonant characteristics

of the system. The ‘roll off’ frequency is also a point of interest, as it determines the operating bandwidth of the controller. The following figure shows a frequency sweep through the controller on the plant. This sweep was consistent with both the Deadbeat and the PID controllers, in order to allow for comparisons.

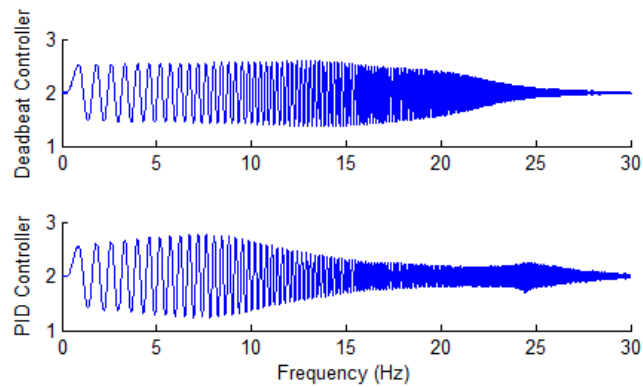


Figure 7.9: Frequency response

Each controller in Figure 7.9 was fed with a sine wave which had a frequency matching the x-axis of the figure. This will allow for a frequency response understanding of the system. It can be seen that the response of the deadbeat controller is significantly different when compared to the PID controller. Note that the magnet is being oscillated at a 2cm point, with a $\pm 0.5\text{cm}$ oscillation. In order to gain a better understanding of the frequency response of the system, a gain and phase plot was generated. A gain vs. frequency plot shows the magnitude of the output against the magnitude of the input, across a frequency range. This will give the ability to determine a range of control system characteristics, including the bandwidth and the roll-off frequency. The bandwidth is defined here as the frequencies which are -3dB or above from the low frequency response of the system. The -3dB point is a half in the power spectrum, therefore frequencies which give a response of above half of the low frequency response are considered inside the bandwidth.

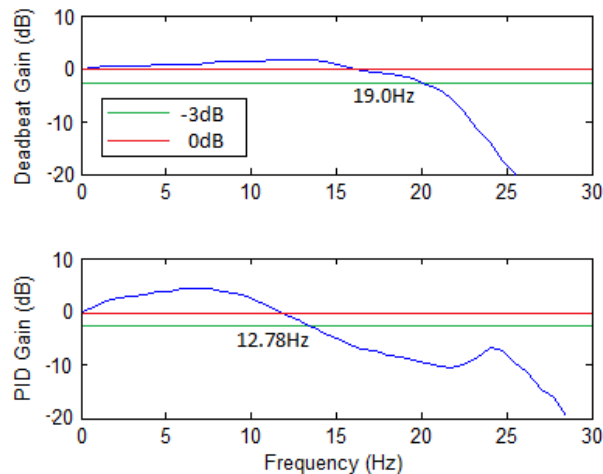


Figure 7.10: Gain plot vs. Frequency

The gain plot shows a large difference between the PID controller and the deadbeat controller. The importance of an even frequency response is that the output will be predictable based on the input signal. Resonances will cause the output to oscillate with a higher magnitude over the input, and could cause equipment damage as limits may be exceeded. The deadbeat controller shows a slight resonance (a point of frequency which gives a peak in the gain) at 13Hz. Resonance can become a problem in the real world, as unwanted higher magnitude vibrations may cause a system to fail. It can be seen that there are two resonant points in the PID controller, the most significant at 7Hz. There is another peak which is present at 24Hz. The bandwidth of the deadbeat controller is 19Hz, which is significantly greater than the PID controller's bandwidth of 12.8Hz. Therefore it can be seen that the gain response of the deadbeat controller is significantly better than the PID controller.

Another point of interest is a Phase plot of the system. This plot measures the phase difference between the output and the input, for the different frequencies. Figure 7.11 shows this plot, and it can be seen that there are no significant improvements of the phase lag between the PID and deadbeat controllers. It would be advantageous to have a lower phase lag as the frequency increases.

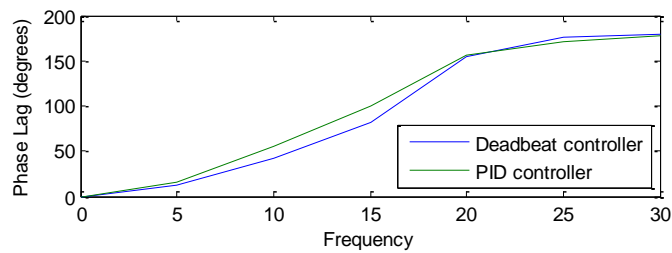


Figure 7.11: Phase vs. Frequency for both controllers

The results from the frequency response comparison show that the deadbeat controller has a more even and predictable gain over a larger frequency range. This is significantly better than the PID controller. Therefore it is deemed that the switched adaptive deadbeat controller is superior to the PID controller when the frequency domain responses are compared.

7.4. Analysis of Results

In control systems, deadbeat response and robustness are two significant design criteria. By using the literature presented in Chapter 1, it is seen that the results shown above exhibit these design criteria. An analysis is undertaken of the results and the significance of these results is presented. The effect of this control system to the real world is also analysed, as it is imperative that practicality is proven.

Now that the best responses have been found using the Maglev machine, an analysis must be undertaken to get a better understanding of the responses. The discrepancies between the plant and the simulator may be the cause of the differences between the plant and the simulation. These differences are now outlined, and the assumptions which were made to ensure the practicality of the controller:

- Continuous plant and discrete controller – it was assumed that the sampling time of the controller was sufficient enough to ensure the quantization error was insignificant
- Backward derivative – As the signal is in real-time, the controller only has the current and previous position readings. It is assumed that the backward derivative was sufficient. Slope prediction and higher order analysis could reduce the small error from this assumption.

- Trapezoidal integral method – It was assumed this method was sufficient for calculating the integral. For better accuracy, a polynomial estimation of the last sampling period could be used to obtain a more accurate integral figure.
- Linearization error – Section 2.3 outlines the approximation method used to model the plant nonlinearity. This data was measured and may not perfectly represent the Maglev machine
- Wind resistance – The effect of wind resistance was neglected, however would have a small effect on the magnet

A major assumption which was made at the beginning of this project was to assume that the friction seen on this machine was viscous and perfectly linear. A viscous assumption denotes that the friction is the same as a damped system, where the change in movement was damped. Therefore the deadbeat controller design was designed with this assumption, however is not entirely correct. There appeared to be a ‘stick-slip’ effect on the magnet, which appeared to affect the output of the plant slightly. As stated in the Section 2.1, a paper has already been written where methods were implemented in an attempt to remove this nonlinearity, with moderate success. From the results of the deadbeat controller on the plant, it is seen that this nonlinearity has little to no effect on the magnet response.

All of the assumptions stated above are all contributory to the discrepancies between the simulation and the plant. However, all of the assumptions made were small in nature, and it can be seen that this has only slightly affected the implementation of the controller on the plant. As proved before, the final deadbeat controller was able to reject large changes in the plant model. Therefore linearity errors are rejected by the controller in most cases.

Another analysis can be undertaken of the maximum coil current which can be used in the operation of this plant. If the controller were to output a current above the amplifier’s limits, the system protection causes the plant to fail. This poses a large safety risk in the real-world, as a Maglev train would derail if the control system failed. Therefore current limits must be put in place to ensure that the coil is not overloaded. Contingencies would also have to be put in place, where emergency backup/limp home modes are engaged if the control system were to fail.

There are many other controllers which are an alternative to the PID controller. Even though the PID controller is the most implemented in the industry, other controllers are available when performance is a necessity. The literature presented in this report showed that there are other control techniques which have been applied to the Maglev plant, in an attempt to get a better response. The best response found was 200ms, for a magnet movement of 2cm, shown in Figure 1.9. This was using a feedback linearization technique, and is a little slower than the deadbeat controller's response. However, the response of the deadbeat controller exhibited an overshoot, which means that feedback linearization is an alternative to the deadbeat controller.

An excellent selling point for this controller was the ability to reject large changes in the control system dynamics. It was proven that the deadbeat controller response to this system change was superior when compared to a PID controller. Therefore if a performance criteria was robustness (the ability to reject changes in the plant), which is a requirement of a Maglev train. This shows the deadbeat controller designed here would be an effective controller. Changes in a Maglev train plant could be the addition of passengers, or the down force generated with larger train speeds. Being able to reject these external disturbances is imperative in control systems, particularly systems which have the ability to cause serious damage. Changes in temperature of the system also results in a change of the system transfer function.

7.5. Discussion of Results

The results shown in this chapter are indicative of the viability of the robust switched adaptive deadbeat controller. In order to gain an understanding of the practicality of this controller, it requires comparison with a PID controller against many criteria. These criteria include performance, design time and reliability. Although the results of this controller are superior to a PID controller, the design time and resources required for this controller is quite intensive. Therefore a discussion of this controller is required.

The controller designed here has been proven to exhibit an excellent response time when compared to a PID controller. Therefore when performance is a necessity, this controller would be selected over a PID controller. A performance criterion could be that the settling time is to be 200ms, which is highly likely to be the case in a system

like this. As the PID controller could not meet this performance criterion, it would not be selected in this case. Therefore the deadbeat controller could be used in a situation like this. As described in Chapter 1, NASA requires robust, reliable and high performance control systems in order to correctly control their rovers. The deadbeat controller has exhibited these characteristics, and therefore could be used in a situation with these criteria.

Although the deadbeat controller exhibits excellent characteristics in the control world, it does have some disadvantages. A large disadvantage of this controller is the requirement to have an estimation of the control system at hand. A PID controller only requires the tuning of some parameters, which can be done by observing the response of the plant. As shown in this report, the deadbeat controller requires a deep understanding of the underlying theory, as well as an understanding of the controller. Development of this system was time and resource intensive, when compared to that of the PID controller. Therefore if performance was not a necessity, the PID controller would be chosen as it is a cheaper (in terms of time upkeep) solution to a majority of control system requirements.

Although the design resources and time was extensive for the deadbeat controller, a set of guidelines and design software could be made to streamline this process. Matlab was imperative to the success of this project. A toolbox could be designed which performs the entire design process, based on the user inputted dynamics of the system. This could allow the deadbeat controller to become a viable option over a PID controller, as there would be much less upkeep in the parameter design. Another alternative would be to have this controller design underwritten into industry available PLC units. This would allow for this deadbeat controller to be an effective alternative to the PID controller. If this occurred, the robust deadbeat controller could become a viable industry-available tool.

7.6. Design Experience

Another result of designing this machine is the experience gained. The entire design cycle has been observed here, where a concept was applied to a real-world machine. Real-world problems were experienced when applying a real-time controller onto a plant, which will be seen in the industry. Methods were applied in order to fix the

design issues, and were applied to the plant. The results were quite indicative of the successful design and implementation of a controller of this precision. Now that the results have been analysed, conclusions can be made on the Switched Adaptive Deadbeat Controller.

Chapter 8

Conclusions and Further Work

8.1. Conclusions

The entire design process has been executed in this project. The design process involves many steps for the successful design and implementation of a control system similar to this. This process is applicable to any design situation, and is an excellent core skill for any Engineer.

The first step in this process was to find supportive literature on the topic at hand. This literature was found using accessible professional databases, where the most relevant literature was selected. This allowed for an analysis to be made of the existing control systems in the real world, as well as work which other entities performed on the Maglev machine. The results of the investigation showed that the robust deadbeat controller was well suited to this application; however literature on this topic was scarce and complex in nature.

The Maglev plant was then analysed, with all of the plant dynamics being drawn from the manual written by ECP. The manual provided a basic understanding of the programming language used to control the Maglev plant, along with a programming example. This was built upon to design the entire control system for the plant. The Maglev manual also provided PID experiments which were designed for classroom use. These experiments were also analysed, and the advanced programming functions were understood. This gave a full understanding of the Maglev plant, which was imperative when applying this project to the plant. Assumptions were made, in order to keep the complexity of the controller within the scope of the project. One of these assumptions was to model the friction as viscous, however in reality the friction had a 'stick-slip' characteristic.

Taylor's linearization theorem was used to remove the nonlinear characteristics from the plant, which gave the ability to simulate the Maglev machine. Assumptions were made based on the linearization error, and it was predicted that more than one controller was required to control the plant properly. Once the plant model dynamics were understood and was able to be simulated using Simulink, the robust deadbeat controller was to be designed. The literature provided a basic procedure in designing this controller, which was adapted for use on the Maglev plant. The controller design used the plant model dynamics found, and a Matlab script was developed to streamline the design process.

Before the controller was implemented on the plant, it was decided that a simulation would be a safer and more practical approach to the issue. Problems in the controller could cause the Maglev plant to operate undesirably, causing coil and plant damage. Therefore a Simulink model was built that had an approximation of the plant connected to the robust deadbeat controller. Simulink blocks were used to build the controller, where real-time integral and derivative blocks were used to represent the complex s-domain parameters.

Testing of the simulation immediately showed that the arbitrary parameters had a sizeable effect on the operation of the plant. Therefore the Matlab script was designed to numerically step through a large number of possible values, and select the parameters which resulted in the optimal response. The simulation showed

promising results, as the controller was able to outperform the built-in PID controller on the plant. However, the success of the controller was not gauged on the response of the simulation, as there are many discrepancies between the simulation and the plant. The simulation was performed in an 'ideal' situation, where the dynamics prescribed by ECP were assumed perfect and all of the external factors (wind, sensor noise etc.) were not modelled. Therefore in order to prove that the controller was actually viable, it was implemented on the plant.

The plant was programmed, where the continuous controller was written in a discrete sense which was not similar to the simulation programming style. This was required, as the real-time controller for the plant was discrete. Each Simulink block was converted into code, and sequentially added to the Maglev programming script. The optimal constants which were found using the simulation were then passed into the plant control algorithm, as it was assumed that these parameters would give a desirable output from the plant.

Initial algorithm testing on the plant found that the controller was unstable and large oscillations were present in the magnet position. Further investigation found that there were many contributing factors to the failure of the control system, which were to be removed if the plant were to function correctly. The most prominent factor diagnosed was the noise on the sensor, which was removed in order to reduce the oscillations. A FIR filter was used to smooth this noise out, and the result showed that it only slightly removed the oscillations. The next issue found was the fact that the plant was unable to operate with integral values above a certain point. This caused the system to drive large overcorrection currents, and gave oscillations. Therefore the variables inputted to the Matlab script were altered to reduce the integral value to a more acceptable level, and a great improvement was obtained in the plant.

Once the integral accumulation was reduced, small oscillations were still present with the controller. Further diagnosis testing showed no discrepancies between the simulation and the plant. As diagnosis testing was limited to the current signal values presented to the real-time controller, the cause of these oscillations was not found. Consultation of the Maglev documentation showed that a thermal compensation

variable was causing a software delay, which was attributed to the small oscillations. Removing this compensation enabled the controller to operate the plant as per specifications.

The results of the responses of the plant about the linearization point showed that the plant and the simulation exhibited similar results. This was promising, however as predicted the plant became more unstable as the magnet deviated further from the linearization point. Therefore a method was proposed to switch between controllers optimised by their prescribed boundaries. The design of this controller was once again performed using the simulation. Obstacles were present with the switching of the controllers, which were addressed inside the simulation. Once the switched controller was proven to work in the simulation, the same optimisation method was used for each boundary controller as that used for the single deadbeat controller.

Results from the simulation showed a deadbeat response superior to the PID controller for any boundary. Once again this controller must be implemented to the plant to be proven viable. Implementation to the plant with the solutions to the plant problems showed that the controller design was a success. Once again the results which were found on the plant were consistent with the results from the simulation.

Once the controller was finalised, comparisons were made between the switched adaptive deadbeat controller and the best optimised PID controller. Control system performance criteria were established for the testing of this machine, which allowed for thorough performance benchmarking. This benchmarking proved that the controller designed was superior to the prescribed PID controller, and proved that this control method was a viable alternative. A notable achievement was a consistent 60% better settling time exhibited by the controller designed here. Frequency response and disturbance rejection characteristics were tested and the results showed that this controller was a viable and effective means for controlling the Maglev plant.

It was reinforced that this controller could be applied to any electro-mechanical plant with the same successful results found in this report. Downsides were also posed to a controller of this precision, where the large development time and resources were presented. It was deemed that with further work, this controller design could be

streamlined to become a quick and effective way of controlling any plant in the real world. This is reinforced with the real-world responses seen on the plant.

Here the entire design process for a controller of this nature has been observed. The real-world applications of this project have also been outlined, where precision control such as this is imperative for real-world electrical and mechanical problems. Proven results also show that the underlying theory of this controller could be applied to a magnetically levitated train, such as the Transrapid in Germany and the Maglev train in China.

8.2. Further Work

Further work could be done to improve on this design for other control systems. A drawback which was outlined in the design of this controller was the time and resources required for development. This design time could be eliminated if time were spent to automate the design and optimisation of this controller, which would make it a viable control system in the real world. This automation would include performing all of the design and testing shown in this report, on a user inputted dynamic control system. This could be written using Matlab, which can perform these calculations in a small amount of time.

As mentioned in the design process, a large amount of assumptions were made in order to keep the scope of this project within the prescribed boundaries. The following methods could be undertaken to reduce their effect on the resultant system.

- Continuous plant and discrete controller – From a plant design prospective, the sampling rate of the controller could be increased (currently 1.1 kHz) in order to allow for a higher integral constant.
- Backward derivative – Predictive and polynomial analysis could be implemented to increase the simultaneous accuracy of the derivative components in the controller
- Trapezoidal integral method –Polynomial and predictive analysis could be used to decrease the error when computing this mathematical function.
- Linearization error – More switched boundaries could be added to the system to reduce the linearization error

- Wind resistance – This could be added to the plant model dynamics and will have a small effect on the controller parameters
- Static/Kinetic friction – The static-kinetic friction could be modelled in the same way which was used by Liceaga, Hernandez and Amezcua (2009), where the integral accumulation was halted when the magnet was near the desired position

All of the actions stated above are methods which could be applied to improve the response of the Maglev plant. This will give a better response, and increase the viability of a system like this. This controller is also not limited to improvements; it can also be used in other configurations on the Maglev plant. This entire project is based on the single magnet/single coil configuration, where the Maglev plant has the facilities for the use of both coils and magnets. The deadbeat controller could be applied to the top magnet, and the cross-forces between the magnets could be modelled as disturbances to each other. This would further test the ability of the robust deadbeat controller.

Further work could also include applying this theory to a completely different plant, one which would be used in industry to perform a prescribed task (i.e. a car manufacturing robot). This would provide excellent benchmarking for the deadbeat control theory and implementation.

References

Angeli, D & Mosca, E 2002, *Adaptive switching supervisory control of nonlinear systems with no prior knowledge of noise bounds*, conference paper, Nevada, USA, viewed 4 July 2012,

<http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1184675>

Benson, T 2011, National Aeronautics and Space Administration, viewed 4 July 2012, <<http://www.grc.nasa.gov/WWW/k-12/airplane/falling.html>>

Brian Betts 2006, The Register UK, viewed 18 May 2012,

<http://www.theregister.co.uk/2006/09/22/german_maglev_crash/>.

Callender, C & Cowan, C & Theodoridis, S 1991, *Non-linear adaptive equalisation using a switched coefficient adaptive filter*, conference paper, Loughborough

University of Technology, Loughborough, viewed 3 July 2012,

<<http://ieeexplore.ieee.org/xpl/articleDetails.jsp?reload=true&arnumber=263742&contentType=Conference+Publications> >

Dawes, J & Ng, L & Dorf, R & Tam, C 1994, *Design of Deadbeat Robust Systems*, pp1597-1598, conference paper, University of California, Davis, viewed 3 July

2012, <http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=381478&tag=1>

Educational Control Products 2002, Educational Control Products California,

America, viewed 14 March 2012, <http://www.ecpsystems.com/controls_ifsw.htm>

Engineers Australia 2010, Institution of Engineers, Australia, viewed 18 May 2012, <<http://www.engineersaustralia.org.au/sites/default/files/shado/About%20Us/Overview/Governance/CodeOfEthics2000.pdf>>.

Fry, D 2003, *The Victorian Web*, viewed 4 July 2012, <<http://www.victorianweb.org/technology/railway1.html>>

Hajjaji, A & Ouladsine, M 2001, '*Modeling and Nonlinear Control of Magnetic Levitation Systems*', viewed 10 May 2012, <<http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=937416>>.

Lee, H & Kim, K & Lee, J 2006, *Review of Maglev Train Technologies*, Hanyang University, Korea, viewed 10 May 2012, <<http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=1644911>>.

Liceaga, J & Hernandez, D & Amezquita, L 2009, *Nonlinear control of a Magnetic Levitation System*, conference paper, Edo, Mexico, viewed 4 July 2012, <<http://dl.acm.org/citation.cfm?id=1673894>>

Nataraj, P. S. V. & Mukesh, D 2004, *Robust Control Design for Nonlinear Magnetic Levitation System using Quantitative Feedback Theory (QFT)*, Indian Institute of Technology Bombay, Mumbai India, viewed 10 May 2012, <<http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4768751>>.

Nise, N 2011, *Control Systems Engineering*, 6th edn, California State Polytechnic University, Pomona

Parks, T 1999, *Manual for Model 730 Magnetic Levitation System*, Educational Control Products, California, America.

Wen, P & Lu, T 2008, *Decoupling control of a twin rotor MIMO system using robust deadbeat control technique*, University of Southern Queensland, Toowoomba, viewed 3 June 2012, <http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4668654&tag=1>

Appendix A

Project Specification

University of Southern Queensland
FACULTY OF ENGINEERING AND SURVEYING
ENG4111 Research Project
PROJECT SPECIFICATION

FOR: **ADAM CLARKE**

TOPIC: **ROBUST DEADBEAT CONTROL OF THE NON-LINEAR
MAGNETIC LEVITATION SYSTEM USING ADAPTIVE
CONTROL TECHNIQUES**

SUPERVISOR: DR. Paul Wen

PROJECT AIM: To investigate, develop and implement precise control systems using the ECP 730 Magnetic Levitation control/development instrument.

PROGRAMME: Issue B, 5th September 2012

1. Investigate the ECP Model 730 hardware and software
2. Test the system and study the existing demo program and experiments
3. Design and simulate a dead-beat controller for the Maglev system
4. Design and simulate an adaptive dead-beat controller for the Maglev system
5. Implement, test and evaluate the deadbeat controller on the Maglev system
6. Implement, test and evaluate the adaptive deadbeat controller on the Maglev system
7. Compare the design responses with proportional, integral and derivative (PID) control

APPROVED Adam Clarke (Student) 19/10/2012

Paul Wen (Supervisor)

Examiner/Co-Examiner: _____

Appendix B

Matlab Code for Simulation

Refer to the CD for the full code listing.

1) Numerical analysis of the force-distance relationship

```

%% Adam Clarke - Final Year Project
%% Script to create a fourth-order nonlinear approximation of the MagLev
actuator characteristics
clc
clear
b=6.2; %offset parameter, taken from ECP instructors supplement
magMass=0.119; %Specify magnet weight, measured
gravity=9.81; %Specify gravity applied, known
magWeight=magMass*gravity; %calculate magnet weight

%% Input the measured data from the Maglev plant
%Control effort applied to the machine coil
controlEffort=[4000; 5000; 6000; 8000; 10000; 12000; 14000; 18000; 22000];
%Physically measured output
measuredDisplacement=[0.5; 0.9; 1.2; 1.9; 2.2; 2.7; 3.2; 3.7; 4.2];
%Apply mean regression (gradient) to function to obtain a value
a=mean(controlEffort./((measuredDisplacement+b).^4.*magWeight));

%% plot experimental data
figure (1)
plot(measuredDisplacement,controlEffort,'+',measuredDisplacement,controlEffort,'--')
axis([0 5 0 25000]) %set the current axis
title('Actuator Characteristics, measured data');
xlabel('Magnet distance from lowest possible position');
ylabel('Equalising control effort')

%% plot numerically estimated curve data
estimatedDisplacement=0:0.1:5; %matrix of test points
%evaluate test points to show numerical solution
estimatedControlEffort=magWeight.*a.*((estimatedDisplacement+b).^4);
figure (2)
hold on
plot(estimatedDisplacement,estimatedControlEffort/1000)

%% plot(measuredDisplacement,controlEffort/1000,'+')
axis([0 5 0 25])
title('Nonlinear Actuator Characteristics');
xlabel('Magnet distance from zero point');
ylabel('Equalising control effort (KiloUnits)')
display(['For a displacement of b=', num2str(b), ' units, the other
constant a=', num2str(a), ' units'])
display('This satisfies the numerical estimation of Full=U1/(a(y1+b)^4)')

```

2) Single deadbeat controller – simulation and numerical optimisation (Calls the Simulink files in Appendix D)

```

%% Adam Clarke - Final Year Project
%% Single Deadbeat controller simulation and numerical optimisation
%clc
%clear

%% Define plant constants

```

```

a1=1.6448; %Found using numerical methods
b1=6.2; %Given, a specific magnet offset
m=0.121; %Magnet mass, kilos
W=m*9.81; %Magnet weight
desiredposition=20000; %linearisation point
maxlimit=20000; %Define controller current limits
%Define symbolic toolbox variables
syms X Y s K3 K aa bb cc dd Ka Kb %allows for easier calculations
%Get Deadbeat Characterisitc equation
alpha=1.90; beta=2.20; Tsetd=4.04; Tr90=3.48; %Optimised characteristic
equation
numplot=0; %Used to plot different optimised figures
indkb=0; % index for Kb
indk=0; % index for K
indk3=0; % index for K3

%% Loop for first dimension of arbitrary variables
for Tdesired=[0.1] %Define the desired settling point (an array for
optimisation)
    omega=Tsetd/(0.7*Tdesired); %Calculate numerator for characteristic
equation
    F3=vpa(omega^3/(s^3+omega*alpha*s^2+omega^2*beta*s+omega^3)); %Define
characteristic equation
    G1=(K*(K3*(s^2+X*s+Y)))/s; %Define the controller, Gc(s)
    G2=aa/(bb*s^2+cc*s+dd); %Define the plant , Gp(s)
    H1=1+Kb*s; %Feedback zero
    H2=Ka; %State-variable feedback
    ylo=desiredposition/10000;
    Ulconst=W*a1*(ylo+b1)^4; %Find the acuator/current gain (N/count)
    kldid=4/(ylo+b1)*W*100; %Find the linearized x' constant(N/m)
    Klconst=1/(a1*(b1+ylo)^4); %Find the acuator/current gain (N/count)
    ksens1=10^6; %Sensor gain in counts per meter
    ksys1=Klconst*ksens1; %Get the system gain, sensor gain x acuator gain
(N/m)
    Num1=ksys1/m; %Get the numerator of the linearised transfer function
    Den1=[1 4 kldid/m]; %Get the denominator of the linearised transfer
function

    %% Calculate the closed-loop transfer function
    syms X Y s K3 K Ka Kb %Use symbolic toolbox to assist in variable
handling
    Ts=G1*G2/(1+G2*H2+G1*G2*H1); %Calculate the closed loop transfer
function
    aa=Num1; %get the numerator of the plant transfer function
    bb=1; %get the denominator coeficcients for the plant transfer function
    cc=4; %get the denominator coeficcients for the plant transfer function
    dd=Den1(1,3); %Convert the linearised equation into a system transfer
function
    Ts=subs(Ts); %Calculate the equivilent closed loop and crelevant
characteristic points
    %Get the characteristic equation, compare it with the cosed loop TF
characteristic=s^3+(1110*K*K3*Kb*X+1110*K*K3+4)/(1110*K*K3*Kb+1)*s^2+(1110*
Ka+1110*K*K3*X+1110*K*K3*Kb*Y+479)/(1110*K*K3*Kb+1)*s+1110*K3*K*Y/(1110*K*K
3*Kb+1);
    %% Loop to vary Kb (0.25 found to be the optimal point)
    for Kb=[0.25]*Tdesired
        indkb=indkb+1;
        %% Loop to vary K, 0.01 found to be the optimal point
        for K=[0.01] %0.01

```

```

indk=indk+1;
%% Loop to vary K3, 0.01 found to be the optimal point
for K3=[0.01]%0.01
    indk3=indk3+1;
    %Show the s^2 equivilency
    s2=vpa(subs((aa*K*K3 + (aa*K*K3*X*Kb) + cc)/((aa*K3*K*Kb) +
1)))));
    %disp([s2 ' lllislll ' vpa(omega*alpha) 's2'])
    %Show the s equivilency
    s1=vpa(subs((aa*Ka + aa*K3*K*X + (aa*K*K3*Y*Kb) +
dd)/((aa*K*Kb*K3) + 1)))));
    %disp([s1 ' lllislll ' vpa(omega^2*beta,4) 's1'])
    %Show the 0 equivilency
    s0=vpa(subs((aa*K*K3*Y)/((aa*K*Kb*K3) + 1)));
    %disp([s0 ' lllislll ' vpa(omega^3) 's0'])
    X=((omega*alpha*(aa*K3*K*Kb + 1))-cc-aa*K*K3)/(aa*K3*Kb)
%Calculate the X parameter
    Y=omega^3*(aa*K*Kb*K3+1)/(aa*K*K3) %Calculate the Y
parameter
    Ka=(omega^2*beta*(aa*K*K3*Kb+1)-dd-aa*K*K3*Kb*Y-
aa*K*K3*X)/(aa) %Calculate the Ka parameter
%% Simulate with the current constants, using the test file
sim('Simulationplantadaptivetest',[0 0.5]);
magnetposition=out.signals.values(:); %Get the results for
plotting
    %Get the step parameters for this output (settling time)
    S =
stepinfo(magnetposition,out.time(:),desiredposition,'SettlingTimeThreshold'
,0.0045);
    S=S.SettlingTime(:) %Get the settling time
    settmin(indk3)=S;
    if (S<0.5804) %If the response is good enough, plot the
results
        numplot=numplot+1; %increment register
        figure(numplot) %Current figure number
        subplot(2,1,1) %Set up the plots for comparison
        %Plot the simulated response

plot(out.time(:),input.signals.values(:)/10000,'r',out.time(:),magnetpositi
on/10000)

        legend('Input signal (R(s))','Magnet Position')
        ylabel('Magnet Position (cm)')
        xlabel('Time (s)')
        subplot(2,1,2)
        controloutput=controleffort.signals.values(:); %Get the
control effort (simulated)
        plot(controleffort.time(:),controloutput) %Plot the
control effort
            %xlabel([num2str(K3) 'K3 ' num2str(K) 'K '
num2str(Kb/Tdesired) 'Kb ' num2str(sum(abs(desiredposition-
magnetposition))) 'error' num2str(Tdesired) 'Tdesired'])
            ylabel('Control Effort \alpha Current')
            xlabel('Time (s)')
        end
    end
end
end
end
end

```

3) Switched adaptive deadbeat controller simulation (Calls the Simulink files in Appendix D)

```

%% Switched Adaptive Robust Deadbeat Controller Simulation
%Written by Adam Clarke - 0050101263, Research Project
clc
clear

%% Linearise the plant about the desired position (for simulation purposes)
a1=1.6448; %Define the exponent constant for the curve of best fit
b1=6.2; %Define the magnet-coil offset at zero sensor position
m=0.121; %Magnet weight, measured physically (grams)
weight=m*9.81; %Find the weight of the magnet, based on the gravity and
mass
desiredposition=25000; %define the position which the magnet must be moved
to
maxcontroleffort=40000; %define the control effort limit of 4amps
linearpoint=desiredposition/10000; %find the linearisation point in cm
Klconst=4/(linearpoint+b1)*weight*100; %Find the linearized x'
constant (N/m)
Ulconst=1/(a1*(b1+linearpoint)^4); %Find the acuator/current gain (N/count)
ksens1=10^6; %Sensor gain in counts per meter
ksys1=Ulconst*ksens1; %Get the system gain, sensor gain x acuator gain
(N/m)
Num1=ksys1/m; %Get the numerator of the linearised transfer function
Den1=[1 4 Klconst/m]; %Get the denominator of the linearised transfer
function
aa=Num1; %Get the first coefficient of the numerator
bb=1; %Get the third coefficient of the denominator
cc=4; %Get the second coefficient of the denominator
dd=Den1(1,3); %Get the first coefficient of the denominator
K=0.01; %Define the optimised K parameter (found from numerical solving)
K3=0.01; %Define the optimised K3 parameter (found from numerical solving)

%% Define the parameters for the 4cm piecewise section
Tdesired4=0.25; %Desired settling time value
Kb4=0.7*Tdesired4; %Feedback derivative constant
X4=4.0516e+003; %Proportional term in feedforward loop
Y4=2.1893e+005; %intergal term in feedforward loop
Ka4=-2.8629; %Cascade gain constant

%% Define the parameters for the 3cm piecewise section
Tdesired3=0.126; %Desired settling time value
Kb3=0.25*Tdesired3; %Feedback derivative constant
X3=3.7637e+004; %Proportional term in feedforward loop
Y3=1.3732e+006; %intergal term in feedforward loop
Ka3=-2.1019; %Cascade gain constant

%% Define the parameters for the 2cm piecewise section
Tdesired2=0.0873; %Desired settling time value
Kb2=0.25*Tdesired2; %Feedback derivative constant
X2=5.0218e+004; %Proportional term in feedforward loop
Y2=2.6062e+006; %intergal term in feedforward loop
Ka2=-2.4675; %Cascade gain constant

%% Define the parameters for the 1cm piecewise section
Tdesired1=0.0538; %Desired settling time value
Kb1=0.25*Tdesired1; %Feedback derivative constant
X1=7.9590e+004; %Proportional term in feedforward loop

```



```

Y1=6.6195e+006; %intergal term in feedforward loop
Ka1=-3.5786; %Cascade gain constant

%% Simulation and results plotting
sim('Simulationplantadaptive', [0 0.5]); %pass the given data into the
piecewise controller
magnetposition=out.signals.values(:); %Get the simulation response based on
the desired position
figure(1)
subplot(2,1,1)
plot(out.time(:),magnetposition) %Plot the magnet position over time
subplot(2,1,2)
controloutput=controleffort.signals.values(:); %plot the control output
applied to the plant
plot(controleffort.time(:),controloutput) %plot the control output applied
to the plant
xlabel([num2str(K3) 'K3 ' num2str(K) 'K ' num2str(Kb) 'Kb      '
num2str(sum(abs(desiredposition-magnetposition))) 'error' num2str(Tdesired)
'Tdesired'])

```

4) Data manipulation from the Maglev plant (Full Matlab listing and data arrays are provided on the CD)

```

%% Adam Clarke - Final year project
%% Script used to read data from the Maglev plant
importData=load('MaglevwithhandDeadbeat.txt'); %Load the data file into
Matlab
sizeplot=round(size(importData(:,5))/2-1); %Segment the data (to remove
unwanted data)
sampleTime=importData(2:sizeplot,2)-0.009; %Split the data file up to get
the sample time (s)
figure (1) %Assign a figure number
axis([-0.08 1 2 4]) %Set the axis to represent required portion
hold on %Allow for multiple plots on the same figure
output=(importData(2:sizeplot,5)+20000)/10000; %Get the magnet position
data
plot(sampleTime,output) %Plot the magnet against time

```

Appendix C

C Code for the Maglev Plant

1) C-Code for the single deadbeat controller

Note: Comment blocks are normally delimited with a semicolon. For readability, the colon was converted to a percentage and presented in Matlab.

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%ECP Model 730 Maglev Machine - Deadbeat Controller

%*****Initialise Variables*****
#define Ts q4 %Sampling time
#define plot q10 %Plottable data
#define perrorplot q11 %Plottable data
#define ierrorplot q12 %Plottable data
#define derrorplot q13 %Plottable data
#define error q18 %Error signal e(t)
#define derror q19 %Derivative feedforward
#define perror q20 %Proportional feedforward
#define ierror q21 %Integral feedforward
#define lastpos q22 %Previous sensor position
#define dcalc q23 %Calculation step for derivative
#define icalc q24 %Calculation step for integral
#define intlact q25 %Integral accumulation figure
#define intnow q26 %Current integral area
#define pid q27 %PID response
#define kpqid q28 %PID*feedforward gain
#define output q29 %Output of the entire system
#define K q30 %K gain (controller)
#define Ka q31 %Ka gain (controller)
#define K3 q32 %K3 gain (controller)
#define X q33 %X gain (controller)
#define Y q34 %Y gain (controller)
#define Kb q35 %Kb gain (controller)
#define H1 q36 %Feedback pole
#define lasterror q37 %Last error position value
#define previous1 q38 %Previous position (for FIR filter)
#define previous2 q39 %Previous position (for FIR filter)
#define previous3 q40 %Previous position (for FIR filter)
#define previous4 q41 %Previous position (for FIR filter)
#define previous5 q42 %Previous position (for FIR filter)
#define currentpos q48 %current position (for FIR filter)
#define index q50 %Index for counter loop

%*****Assign Variables*****
Ts=0.000884 %Set Sampling
%Specify Parameters
control_effort2=0
control_effort1=0
intlact=0
currentpos=0
lastpos=0
lasterror=0

%*****Assign Controller Parameters*****
K=0.01 %Found by optimising response
K3=0.01 %Found by optimising response
Ka=-0.5133 %Found by optimising response
Kb=0.25*0.16 %Found by optimising response
X=6909 %Found by optimising response

```

```

Y=201600 %Found by optimising response

%**Assign Indexing and Previous Positions**
previous1=0 %Assign first signal to be zero
previous2=0 %Assign second signal to be zero
previous3=0 %Assign third signal to be zero
previous4=0 %Assign fourth signal to be zero
previous5=0 %Assign fifth signal to be zero
index=0 %Reset indexing figure

%*****Start of Real Time Algorithm*****
begin %Start of program loop
currentpos=sensor1_pos %Get the current sensor position
%% FIR filter implementation
%currentpos=0.0264*sensor1_pos+0.1405*previous1+0.3331*previous2+...
    ...+0.3331*previous3+0.1405*previous4+0.0264*previous5
H1=Kb*(currentpos-lastpos)/Ts+currentpos %Use backward derivative for zero
error=(cmd1_pos+20000-H1)*K*K3 %calculate error, (input-output*pole)
dcalc=error-lasterror %Use backward derivative for feedforward PID control
derror=(dcalc/Ts) %Calculate slope of derivative, from previous signal
intnow=(error+lasterror)*Ts*0.5*Y %Calculate integral using Trapezoidal
method
ierror=intnow+intl %Accumulate the integral from the last value
perror=X*error %Calculate the feedforward proportional error
pid=perror+ierror+derror %Calculate the PID output
kpid=pid %Multiply the PID output by the feedforward gain
output=kpid-Ka*currentpos %Minus the state-variable feedback value

%*****Over-current protection*****
if (output>20000) %If the current is too high
control_effort1=20000 %Clamp the current
endif
if (output<-20000) %If the current is too negative
control_effort1=-20000 %Clamp the current
endif
if (output>-20000 and output<20000) %If current range is safe
control_effort1=output %feed output with current control effort
endif

plot=control_effort1 %Plot the current data for analysis and Matlab
plotting
perrorplot=lasterror %Plot the current data for analysis and Matlab
plotting
ierrorplot=intnow %Plot the current data for analysis and Matlab plotting
derrorplot=ierror %Plot the current data for analysis and Matlab plotting

%*****Previous signal update*****
intl=intl %Update the last integral to represent the current integral
lasterror=error %Update the last error to represent the current error
lastpos=currentpos %Update the last position to represent the current
position
previous5=previous4 %Update the last position to represent the current
position
previous4=previous3 %Update the last position to represent the current
position
previous3=previous2 %Update the last position to represent the current
position
previous2=previous1 %Update the last position to represent the current
position

```

```
previous1=sensor1_pos %Update the last position to represent the current
position
end %End of control loop, returns to 'begin'
```

2) Adaptive deadbeat controller source code. Functions common to the deadbeat controller are not commented

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%ECP Model 730 Maglev Machine - Switched Adaptive Controller

%*****Initialise Variables*****
#define Ts q4
#define plot q10
#define perrorplot q11
#define ierrorplot q12
#define derrorplot q13
#define uterm1 q17
#define error q18
#define derror q19
#define perror q20
#define ierror q21
#define lastpos q22
#define dcalc q23
#define icalc q24
#define intlact q25
#define intnow q26
#define pid q27
#define kpid q28
#define output q29
#define K q30
#define K3 q31
#define Ka1 q32
#define X1 q33
#define Y1 q34
#define Kb1 q35
#define H1 q36
#define lasterror q37
#define currentpos q38
#define Ka2 q39
#define X2 q40
#define Y2 q41
#define Kb2 q42
#define Ka3 q43
#define X3 q44
#define Y3 q45
#define Kb3 q46
#define Ka4 q47
#define X4 q48
#define Y4 q49
#define Kb4 q50
#define Ka5 q51
#define X5 q52
#define Y5 q53
#define Kb5 q54
#define Ka6 q55
#define X6 q56
#define Y6 q57
#define Kb6 q58
```

```
#define test q59

%*****Assign Variables*****
Ts=0.000884
%Specify Parameters
control_effort2=0
control_effort1=0
intl=0
currentpos=0
lastpos=0
lasterror=0
K=0.01
K3=0.01

%*****Assign Boundary Controller Parameters*****
%Boundary optimised for 0-0.5cm
Ka1=-0.5477
Kb1=0.25*0.13
X1=8569.8
Y1=304250

%Boundary optimised for 0.5-1cm
Ka2=-0.5951
Kb2=0.25*0.13
X2=9390.0
Y2=333280

%Boundary optimised for 1-1.5cm
Ka3=-0.6682
Kb3=0.25*0.14
X3=10197
Y3=337230

%Boundary optimised for 1.5-2cm
Ka4=-0.6780
Kb4=0.25*0.15
X4=9902.5
Y4=306790

%Boundary optimised for 2-2.5cm
Ka5=-0.7396
Kb5=0.25*0.14
X5=11408
Y5=377160

%Boundary optimised for 2.5-3cm
Ka6=-0.8725
Kb6=0.25*0.16
X6=12550
Y6=365710
test=0

%*****Start of Real Time Algorithm*****
begin
%*****Controller optimised to 0-0.5cm*****
currentpos=sensor1_pos
if (cmd1_pos<10000) %Test for this controller's range, if true execute code
H1=Kb1*(currentpos-lastpos)/Ts+currentpos
```

```
error=(cmd1_pos-H1)*K*K3
dcalc=error-lasterror
derror=(dcalc/Ts)/(0.0015)
intnow=(error+lasterror)*Ts*0.5*Y1
ierror=intnow+intl原因
perror=X1*error
pid=perror+ierror+derror
output=pid-Ka1*currentpos
intl原因=ierror
lasterror=error
lastpos=currentpos
endif %End the control logic for this bound, no other controllers will be
used

%*****Controller optimised to 0.5-1cm*****
if (cmd1_pos!<10000 and cmd1_pos<15000) %Test for this controller's range
H1=Kb2*(currentpos-lastpos)/Ts+currentpos
error=(cmd1_pos-H1)*K*K3
dcalc=error-lasterror
derror=(dcalc/Ts)/(0.0015)
intnow=(error+lasterror)*Ts*0.5*Y2
ierror=intnow+intl原因
perror=X2*error
pid=perror+ierror+derror
output=pid-Ka2*currentpos
intl原因=ierror
lasterror=error
lastpos=currentpos
endif %End the control logic for this bound, no other controllers will be
used

%*****Controller optimised to 1-1.5cm*****
if (cmd1_pos!<15000 and cmd1_pos<20000) %Test for this controller's range
H1=Kb3*(currentpos-lastpos)/Ts+currentpos
error=(cmd1_pos-H1)*K*K3
dcalc=error-lasterror
derror=(dcalc/Ts)/(0.0015)
intnow=(error+lasterror)*Ts*0.5*Y3
ierror=intnow+intl原因
perror=X3*error
pid=perror+ierror+derror
output=pid-Ka3*currentpos
intl原因=ierror
lasterror=error
lastpos=currentpos
endif %End the control logic for this bound, no other controllers will be
used

%*****Controller optimised to 1.5-2cm*****
if (cmd1_pos!<20000 and cmd1_pos<25000) %Test for this controller's range
H1=Kb4*(currentpos-lastpos)/Ts+currentpos
error=(cmd1_pos-H1)*K*K3
dcalc=error-lasterror
derror=(dcalc/Ts)/(0.0015)
intnow=(error+lasterror)*Ts*0.5*Y4
ierror=intnow+intl原因
perror=X4*error
pid=perror+ierror+derror
output=pid-Ka4*currentpos
```

```
intlast=ierror
lasterror=error
lastpos=currentpos
endif %End the control logic for this bound, no other controllers will be
used

%*****Controller optimised to 2-2.5cm*****
if (cmd1_pos!<25000 and cmd1_pos<30000) %Test for this controller's range
H1=Kb5*(currentpos-lastpos)/Ts+currentpos
error=(cmd1_pos-H1)*K*K3
dcalc=error-lasterror
derror=(dcalc/Ts)/(0.0015)
intnow=(error+lasterror)*Ts*0.5*Y5
ierror=intnow+intlast
perror=X5*error
pid=perror+ierror+derror
output=pid-Ka5*currentpos
intlast=ierror
lasterror=error
lastpos=currentpos
endif %End the control logic for this bound, no other controllers will be
used

%*****Controller optimised to 2.5-3cm*****
if (cmd1_pos!<30000 and cmd1_pos<40001) %Test for this controller's range
H1=Kb6*(currentpos-lastpos)/Ts+currentpos
error=(cmd1_pos-H1)*K*K3
dcalc=error-lasterror
derror=(dcalc/Ts)%
intnow=(error+lasterror)*Ts*0.5*Y6
ierror=intnow+intlast
perror=X6*error
pid=perror+ierror+derror
output=pid-Ka6*currentpos
intlast=ierror
lasterror=error
lastpos=currentpos
endif %End the control logic for this bound, no other controllers will be
used

%*****Apply Large excursion control law*****
if ((cmd1_pos-sensor1_pos)>21000)
intlast=0
output=25000
endif
if ((cmd1_pos-sensor1_pos)>13000)
intlast=0
endif
if ((cmd1_pos-sensor1_pos)<-21000)
intlast=0
output=-25000
endif
if ((cmd1_pos-sensor1_pos)<-13000)
intlast=0
endif
if (cmd1_pos<2000)
intlast=0
output=0
endif
```

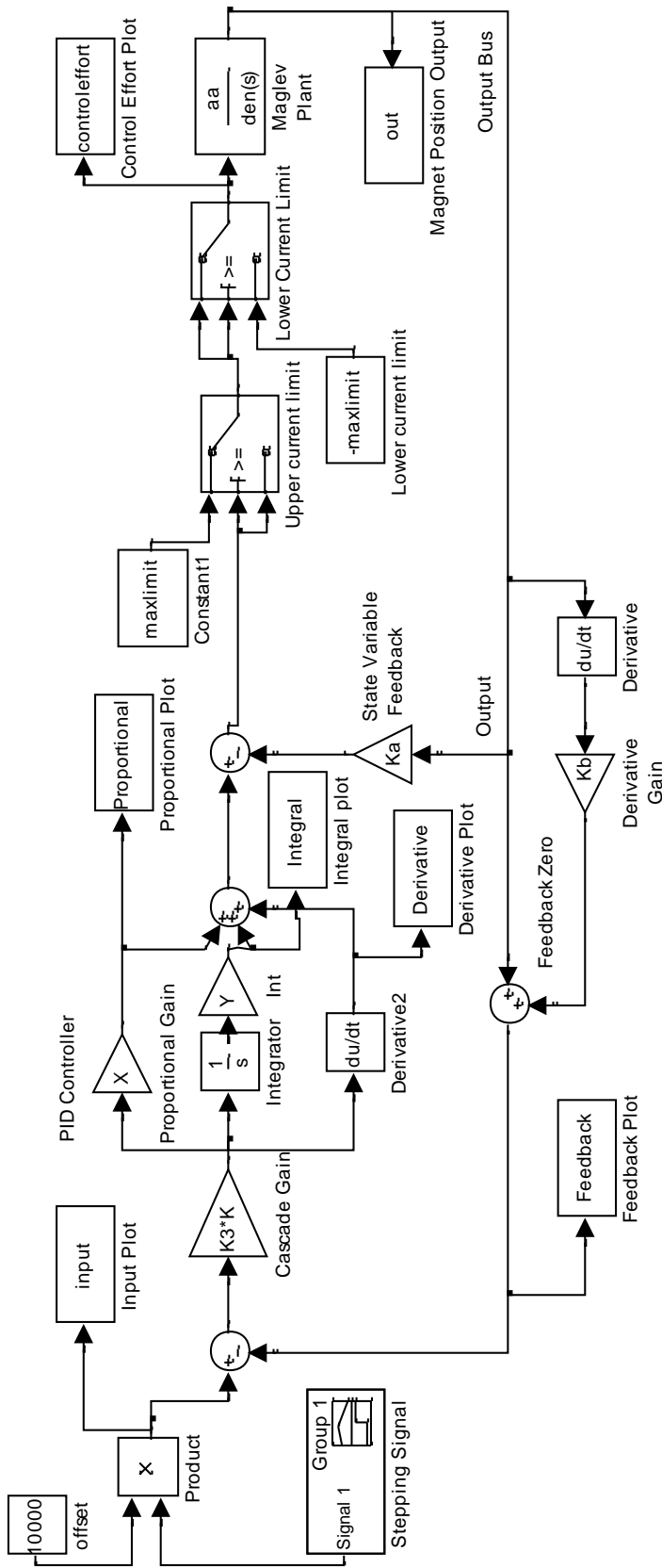


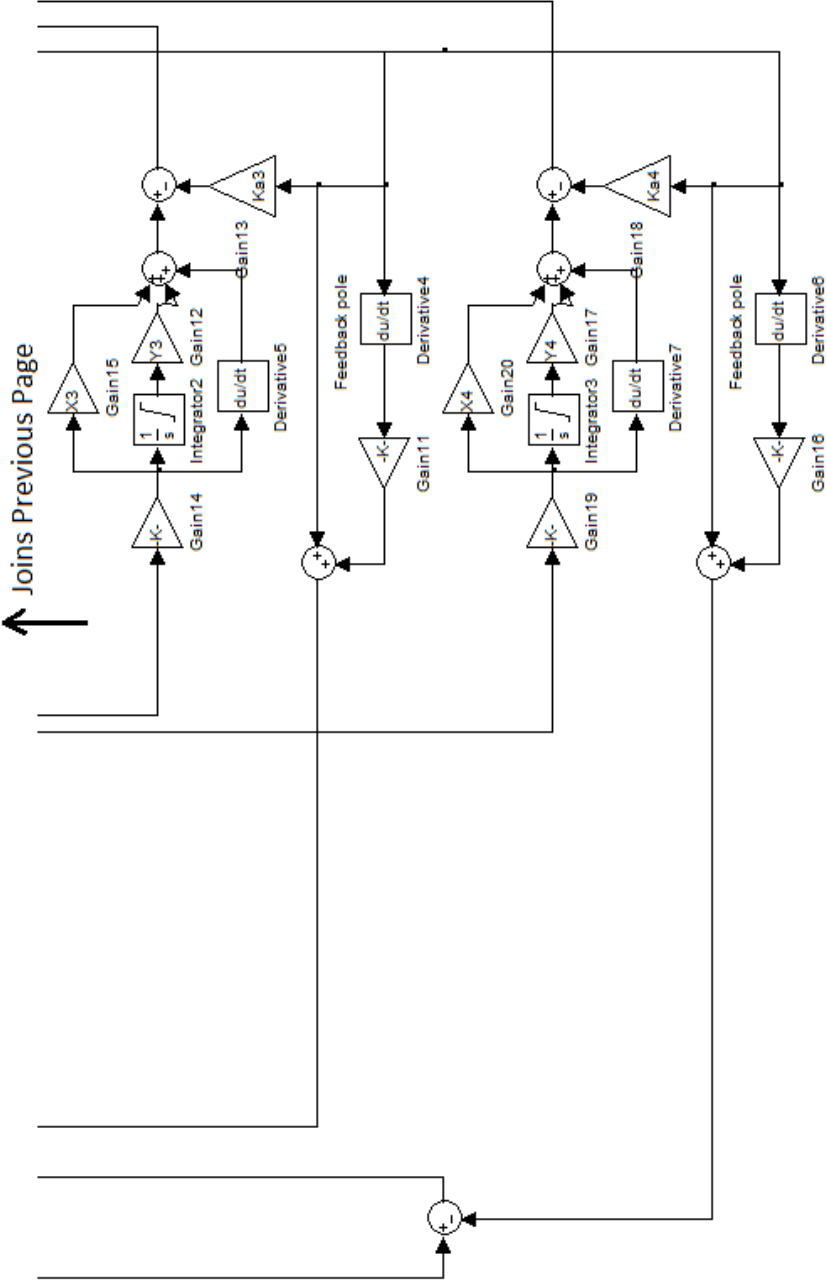
```
%*****Apply current limiting algorithm*****  
if (output>24000)  
control_effort1=24000  
endif  
if (output<-24000)  
control_effort1=-24000  
endif  
if (output>-24000 and output<24000)  
control_effort1=output  
endif  
  
plot=H1  
perrorplot=perror  
ierrorplot=ierror  
derrorplot=derror  
end
```

Appendix D

Simulink Models

1) Single Deadbeat Controller Simulink model (Called by the single deadbeat Matlab script)





Appendix E

Risk Assessment

The professional engineer has a duty of care to others, themselves and the environment. A risk analysis is required to identify the risks which are created by all tasks carried out by the project. The purpose of a risk analysis is to identify all possible risks and hazards, and reduce the likelihood and consequence severity of these issues to as low as reasonably practicable.

- A **hazard** is an object which has the potential to cause harm to someone or the environment.
- A **risk** is the likelihood of the stated hazard causing harm. Levels include:
 - Minor – Conceivable, but very unlikely
 - Moderate – Possible to occur however still unlikely
 - Major – Can happen and has happened in the past
 - Extreme – Very likely to happen
- The **exposure** is how often people are exposed to the risk
- The **consequences** are the repercussions which may occur should the hazard occur, levels include:
 - Low – Minor equipment damage may occur
 - Moderate – Major equipment damage or minor injury/illness (e.g. burns, cuts etc.)
 - High – major injury/illness to one or more people
 - Very high – death(s)

Table E.1: Risk Assessment Matrix is a matrix of risk severity. Steps must be taken to reduce risks in the yellow area to a lower level. If a task has a risk in the red area, the entire project must be redesigned to reduce this risk to an acceptable level.

Table E.1: Risk Assessment Matrix

		Consequences			
		Low	Moderate	High	Very High
Chance	Minor				

of Occurring (risk)	Moderate				
	Major				
	Extreme				

Using this matrix, the risks which are present in this project can be identified, analysed and reduced if necessary. The following page contains the risks which have been identified with this project.

Table E.2: Risk assessment for uncontrolled movement of the Maglev magnets

Risk Assessment	
Hazard Description:	Hazard is the uncontrolled movement of the magnet on the Maglev machine, which is exposed to the open world
Risk:	Major – When practicals are implemented, untrained students may attempt to operate the equipment without authorisation
Consequence:	Moderate – Uncontrolled movement can cause small crushing and pinching forces, bruises and cuts may result
Hazard Level:	Yellow. Steps must be taken to reduce the risk of this occurring
Steps to keep risk as low as reasonably practicable:	<ol style="list-style-type: none"> 1) Access to the apparatus must be restricted (achieved through access to control lab being restricted) 2) Appropriate training is required for the operators 3) Gloves are required for the manual handling of the magnet where required 4) When implemented, a guard could be installed to remove the entire risk
Result of Steps:	Hazard level is green if the above guidelines are adhered to

Table E.3: Risk assessment for falling objects

Risk Assessment	
Hazard Description:	The Maglev apparatus, associated control box and PC assembly inside the laboratory can fall if not handled carefully.
Risk:	Moderate – Students may unintentionally knock or move apparatus/control system to a position where it may fall
Consequence:	High – Falling equipment may cause severe damage to legs/feet
Hazard Level:	Yellow. Steps must be taken to reduce the risk of this occurring
Steps to keep risk as low as reasonably practicable:	<ol style="list-style-type: none"> 1) Access to the apparatus must be restricted (achieved through access to control lab being restricted) 2) Appropriate training for behaviour in laboratory must be undertaken before equipment is used 3) Closed in footwear (Preferably hard toed boots) must be used while in the laboratory
Result of Steps:	Hazard level is green if the above guidelines are adhered to

Table E.4: Risk assessment for potential exposed electrical wires

Risk Assessment	
Hazard Description:	Insulation may wear down, causing electric shock
Risk:	Minor – USQ protocol ‘test and tag’ already attempts to address risk
Consequence:	Very High – Exposure to electric shock can cause death
Hazard Level:	Yellow. Steps must be taken to reduce the risk of this occurring
Steps to keep risk as low as reasonably practicable:	<ol style="list-style-type: none"> 1) Access to the apparatus must be restricted (achieved through access to control lab being restricted) 2) Appropriate training for behaviour in laboratory must be

	<p>undertaken before equipment is used</p> <p>3) When implemented, a guard could be installed to remove the entire risk</p>
Result of Steps:	Hazard level is green if the above guidelines are adhered to

Table E.5: Risk assessment for laser radiation from the machine

Risk Assessment	
Hazard Description:	Laser radiation is present when the machine is in operation. Two lasers are required to detect the position of the magnet on the Maglev machine
Risk:	Moderate – The laser module is hidden from sight, however can be observed from one small angle from inside the machine
Consequence:	High – Laser radiation will burn tissue in the eye, causing irreversible damage
Hazard Level:	Yellow. Steps must be taken to reduce the risk of this occurring
Steps to keep risk as low as reasonably practicable:	<p>The steps to reduce the risk of this occurring are the same as those outlined in Table E.3: Risk assessment for</p> <ol style="list-style-type: none"> 1) Access to the apparatus must be restricted (achieved through access to control lab being restricted) 2) Appropriate training for behaviour in laboratory must be undertaken before equipment is used 3) When implemented, a guard could be installed to remove the entire risk
Result of Steps:	Hazard level is green if the above guidelines are adhered to