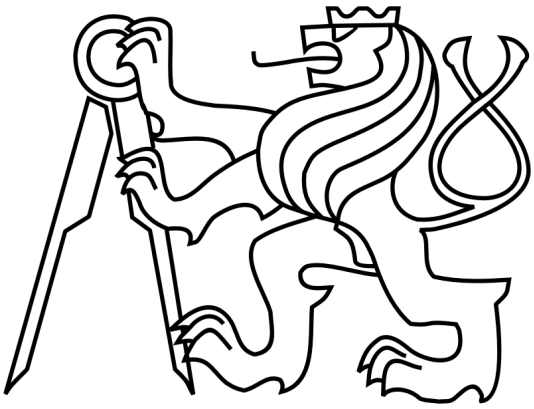
CZECH TECHNICAL UNIVERSITY IN PRAGUE

FACULTY OF MECHANICAL ENGINEERING

Department of Instrumentation and Control Engineering



Master's Thesis

Analysis of Industrial Hoist Mechanisms Braking Torque and Deceleration Control with Neural Networks

Author: Bc. Peter Mark Beneš

Academic Year: 2013/2014

Supervisor: doc. Ing. Ivo Bukovský, Ph.D.

CONTENTS

Abstract

This thesis introduces a new approach to analysis of braking torque of industrial hoist mechanisms, with particular focus to crane applications. This approach is unique as the necessary parameters of the braking torque are measured in real time, with an on the spot evaluation directly following measurement. This evaluation consists of analysis of the real braking torque values of the given hoist mechanism, followed by checking its compliance with the permissible limits as prescribed in standards. This analyse further highlights the need for control of the hoist deceleration, should the results of the braking torque be too high or low, as compared to desired and permissible values.

In particular, this thesis presents a low-cost hardware setup and open-source software application developed for real-time measurement of the braking torque for a general range of industrial hoist mechanisms and also for further analysis of hoist deceleration control via adaptive control methods. The software application acquires real time data and performs an analysis of the real braking torque followed by a test of compliance with the permissible limits as prescribed by standards. Then the measured data of the real hoist system along with data of a mine hoist, from a reviewed case study are analysed for neural network identification and possible feedback control by linear and higher order nonlinear neural units via real-time recurrent learning and batch-propagation through time learning algorithms. The conclusions are made in relation to the measurement, data analysis, and neural network control potentials with connotations to implementation on industrial electrically braked hoist mechanisms.

Acknowledgements

I would like to thank Mr. Ladislav Novák, former Maintenance Manager/Inspector for Montáže s.r.o. for facilitating the testing area, necessary for the experimentation throughout this thesis. I would also like to thank Ing. Jan Beneš, specialist in crane design from IBCE Pty.Ltd., for his valuable advice and expertise.

I would like to also extend my gratitude to my supervisor doc. Ing. Ivo Bukovsky PhD, for his help and cooperation throughout this thesis. Furthermore, to the reviewer Ing. Miroslav Mikulčík from Buhler Praha s.r.o. for his time and valuable suggestions on my thesis.

This thesis is also supported by project SGS12/177/OHK2/3T/12 “Non-conventional and Cognitive Signal Processing Methods of Dynamic Systems”.

1. Introduction

In today's industry, numerous industrial cranes are overlooked when it comes to inspection of the real braking torque of their hoist mechanisms. This, in accordance with crane standards, is quite a strict regulation that can lead to dangerous consequences, if not within the prescribed permissible limits. According to AS 1418.1 -2002 standards for safe crane use and design [1], the permissible limit of braking torque should not fall below a minimum of 1.2 times its full loading value for dynamic braking torque and 1.6 times its full loading value at static braking torque. Should this limit be breached, this can ultimately result in slipping or even loss of load due to too weak braking effect delivered by the hoist braking mechanism. Till today, no publications nor any real devices or systems are known, which provides measurement and analysis of real braking torque, taking into consideration all of the necessary factors which influence its true value.

Thus the first key objective of this thesis is to provide a unique solution to measurement and analysis of the real braking torque for a general range of industrial hoist mechanisms, as is presented in chapter 3 of this thesis. In particular, Section 3.1 presents a low-cost implementation of a hardware setup, for measurement of the hoist deceleration and gearbox casing temperature, which, are the two main real-time variables necessary to measure for determination of the real braking torque. In the presented setup, the angular displacement over time is measured as the hoist undergoes a braking process, via an optical reflective sensor type CNY 70 with a coded ring configuration and TCST2103 with coded wheel configuration, for comparison. The optical reflective sensor output is relayed through a PIC16F628A microcontroller to ensure real time measurement, which may then be relayed to an external processor for data analysis. In the sense of larger industrial hoists it is necessary to measure the gearbox casing temperature and oil temperature of the gearbox lubricant. Thus the DS18B120 temperature sensor was chosen due to its adequate temperature range and compatibility. Because of its low cost, portability and adequate functionality, the Raspberry Pi is presented as a means for processing the data measured by the employed sensor technology. The Raspberry Pi features GPIO pins which may be directly programmed to read the relayed values of the PIC16F628A microcontroller as well as inbuilt protocol for direct communication with specific types of temperatures sensors such as the DS18B120, making it a suitable choice for the scope of this work.
The following section 3.2, presents the second necessary component behind the first key objective which is that of analysis of the braking torque. This is achieved via the design and development of a new unique application for direct measurement of the hoist deceleration and gearbox temperature, given the same hardware configuration as presented in section 3.1, as well as a series of calculation and diagnostics functions for analysis of the real braking torque for the given measured data, which may also be uploaded from other means of measurement of the hoist angular displacement over time at is rotor, and gearbox casing temperature. After review of the necessary theories from industrial crane and hoist standards [1]-[3]&[7], the application was programmed via an open-source programming language Python 2.6 and further extended modules wxPython, Numpy, Matplotlib and RPi.GPIO [9]-[12]. The crane and hoist standards [1]-[3], published by the Standards Australia Association, used throughout this thesis, were chosen due to their accessibility. However, these standards also based on the same regulations prescribed by the ISO standards, and for the scope of this thesis are thus a good basis for the presented work. Following the development of this new application, experimentation on a real industrial hoist is presented. The Gude GSZ 200/400 is used

for the experimentation within the scope of this thesis, due to its manageability of size and excellent documentation. Section 3.3–3.4, highlights the functionality of the presented hardware setup up along with new application for analysis of the real braking torque on the investigated Gude GSZ 200/400 hoist mechanism.
Further to this is a second problem in regards to the operation of industrial hoist mechanisms, which is that of hoist deceleration control, which in turn also has a direct effect on the braking torque of the hoist mechanism. Should the mechanism feature too high deceleration, this can lead to large dynamic forces acting on the load, which can ultimately lead to damaging of the load and overstressing of the hoist structure. Should the deceleration value be too low, it may fall below the prescribed minimum according to standards and thus be a risk of safety. Till now hoist deceleration control is not a widely studied area. Several theoretical studies have been published as exampled in the recent works of [18] & [19], though featuring promising results as will be discussed in section 4.1 of this thesis, there is still a necessity for further optimisation of control, to achieve quicker response towards the desired behaviour of the hoist deceleration as well as improved adherence with the desired behaviour. Thus the second and final key objective of this thesis is to provide a new study into the potentials of adaptive control via neural network architectures as such the linear (LNU) and higher-order (HONUs) neural units with real-time-recurrent-learning (RTRL) and batch-propagation-through-time (BPTT) training methods. This investigation will be undertaken via application of the adaptive identification and control software application as presented in the work [22].

Investigation via means of an adaptive neural network based approach, arises from the already promising theoretical studies of higher-order neural units (HONUs), especially the quadratic neural unit for engineering problems [23]-[28]. These studies are focused on the use of supervised learning based approaches for polynomial structured neural units, also known as a class of HONUs, for adaptive identification and control of real engineering systems. Successful implementation of a quadratic neural unit controller (Neuro-controller) may be recalled from the work [21], used for control of a bathyscaphe system located in the automatic control laboratories of FME at CTU. Where, here, such controller adhered more closely to the desired behaviour of the system than the conventionally used PID controller. An extension on this result may also be recalled in the work [22]. Where, further study was made, via introduction of a new software application for adaptive identification and control, along with further testing on both a theoretical system and the previously mentioned bathyscaphe system. Given these results of real implementation, section 4.3 of this thesis thus aims to investigate via a similar approach the use of such adaptive linear and higher order neural unit based architectures for control of the hoist deceleration problem. The results of section 4.3 are based on two cases of study. The first utilises the measured data of the Gude GSZ 200/400 hoist mechanism as presented in section 3.3. However, due to the Gude GSZ 200/400 being a mechanically braked hoist, the only option for an electrical means of control of this hoist is via an extended single phase variable frequency drive or an additional electrically controlled braking system, extended onto the hoist rotor shaft. Thus, an equivalanced deceleration characteristic is modelled via a conversion table of the given speeds to corresponding input frequencies which may be altered in the sense of an electrically braked hoist as a means of control, such as the considered Gude GSZ 200/400 with extended variable frequency drive. This study highlights that although the initial characteristic for a standard braking process may be quite undesirable, the investigated form of adaptive control is still able to achieve desirable control of the hoist deceleration as is later shown in Figure 35 & Figure 36 of this thesis. The second case examples a reviewed case study of a mine hoist application from the work [20]. This particular hoist system features a dynamic braking system,

which may be controlled via influencing of field current, with the output being the hoist deceleration, corresponding to the lowering supply current. The capabilities of the quadratic neural unit with BPTT training are later shown in Figure 39, further showing the potentials that a neural network based adaptive control scheme has in application to hoist deceleration control of real industrial hoist mechanisms especially exampled with use of real hoist data, different to that of those presented in previous theoretical studies dedicated to the problem.

2. Description of the Industrial Problem in Relation to Hoist Mechanisms

Hoist mechanisms are one of the most crucial tools used in today's industry. These mechanisms widely vary in terms of size and application as such, from heavy lifting as in the crane industry to elevators, mining cabins etc. However when it comes to safety of this general class of industrial hoists a key area which is in need of further study, is that of analysis of the real braking torque present within the hoist mechanism. The braking torque is indeed a crucial element as it is ultimately this parameter that defines the strength of the hoist braking mechanism, prior to slipping of the load. Should the braking torque be overlooked, this can certainly lead to very large or even fatal consequences within the workplace and applications for which they are used in, the ultimate consequence being loss of load, due to poor inspection of the static braking torque present on the hoist.

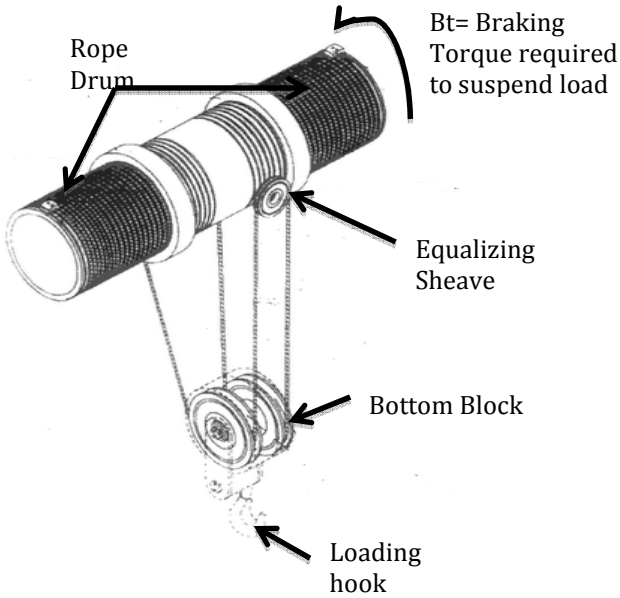


Figure 1: Hoist Mechanism – Principle of Braking Torque (modification from [17])

In Figure 1, the principle behind the braking torque of an industrial hoist mechanism is illustrated. Here we see that the braking torque is in fact the torque or rotational force required to suspend the load of the hoist in the air, should the braking mechanism be disregarded i.e.; the torque necessary to substitute for the braking mechanism, in order to keep the hoist load suspended.

According to crane industrial hoist standards as such that of the AS1418.1 -2002 Clause 7.12.8.1 [1]; Braking systems shall comply with a minimum braking torque of 1.6 times the rated capacity in a static condition and arrest a minimum of 1.2 times the rated hoist capacity, from the maximum

lowering speed, in a dynamical sense. Should the hoist mechanism suspend less than 1.6 times the rated capacity of the hoist mechanism, the load would be at serious risk of being lost by the brake pads, due to insufficient grasping at the brake lining. Currently the methods used in the industry in order to comply with the standards, feature for example over-tightening of the braking callipers, which in turn leads to larger wearing of the braking pads. This is a rather mechanical solution, which is only possible within a few times of application, not to mention the large estimation factor that takes place without actual measurement of the mechanism. Till this day the braking torque is often calculated in a simplified manner, often being calculated via only consideration of the overall average deceleration and not with consideration of the other areas of the braking process present in the deceleration characteristic. Furthermore, a disregard into the passive resistances within the mechanism, that subtract from the actual value of braking torque. One such significant factor is the passive resistances due to viscosity of the gearbox oil, for large application hoist mechanisms, which is variable with temperature of the gearbox. Thus till this day, there are no clear methods employed throughout the industry which are able to determine the braking torque in an effective, accurate manner.

A parallel problem to this, and the second area of focus in this thesis, is the area of hoist deceleration control, primarily for industrial hoists with electric braking systems. The necessity for investigation in this field arises from the increasing necessity for safety within in the industry. In terms of the hoist deceleration, here too a given industrial hoist must comply within a set regulation for the maximal and minimal allowable hoist deceleration, to provide safe braking of the hoist load. Such consequences are that too high deceleration may lead to overstressing of the hoist mechanism and structure, further, damaging of the load. Furthermore, too low deceleration can lead to incompliance with standards and thus being a safety issue, due to potential loss of load in the worst case. Till this day various solutions and devices have been proposed to achieve controlled deceleration of the industrial hoist braking. Such methods till this date include the conventional PID controller, where the set point deceleration and actual deceleration of the hoist are used as the input value of the control system. Or further a fuzzy logic based controller, with a rule based system defined for given limits of the deceleration as input and necessary force for output to manipulate the deceleration. A more recent method may be reviewed in the work [18], where a theoretical study into a fuzzy neural network based controller was proposed to achieve improved behaviour of the hoist deceleration. The results of this study were compared to the conventional PID controller and fuzzy controller, with tests on the hoist as an identified model, showing better behaviour via the proposed fuzzy-neural adaptive approach. However till now, hoist deceleration control is not a widely studied area. With certain solutions to this problem, in terms of produced devices, being in some cases uneconomical especially, for use on smaller sized hoist applications, or not being the most optimal solution. Similarly with the above mentioned controllers investigated in the industry, here too is a necessity for further study to either further optimize the proposed methods of control for more adequate functionality and also investigation into other alternative computational methods for control.

2.1 - Theories and Principles behind Calculation of Real Braking Torque

Referring back to Figure 1, we may see in brief the notion of braking torque. This may be defined as the torque (or moment) necessary to keep the load under suspension, given the absence of all the braking mechanism, which mechanically keeps the load suspended. With this we may set out to derive the true braking torque from the following classical definition;

$$T = J.\varepsilon \tag{1}$$

Equation (1) describes the classical definition for rotational torque, where J denotes the moment of inertia of a rotational mass and ε, denoting the angular acceleration of the rotational mass. From this definition we may then develop out the definition of the real braking torque in the following manner;

$$B_T + \zeta_{mech.} = (\sum J_{rot.}^{eq.} + \sum J_{lin.}^{eq.}).\varepsilon \tag{2}$$

Here B_T [N.m] denotes the real braking torque, where $\zeta_{mech.}$ represents the total mechanical torque such to add to the braking torque of the whole hoist mechanism and ε, representing the angular acceleration [rad/s^2]. With equation (2) we see that in a real hoist system, there is a combination of rotational and linearly moving masses which must be equivalanced for evaluation of the braking torque. Further to this, there exists a total mechanical torque which helps the braking of the hoist mechanism i.e.; passive resistances of the rope winding, viscosity of the gearbox oil, all of which should be considered when evaluation braking torque of a hoist mechanism. Thus the goal is now to derive, these key variables necessary for evaluation of the real braking torque as per equation (2).

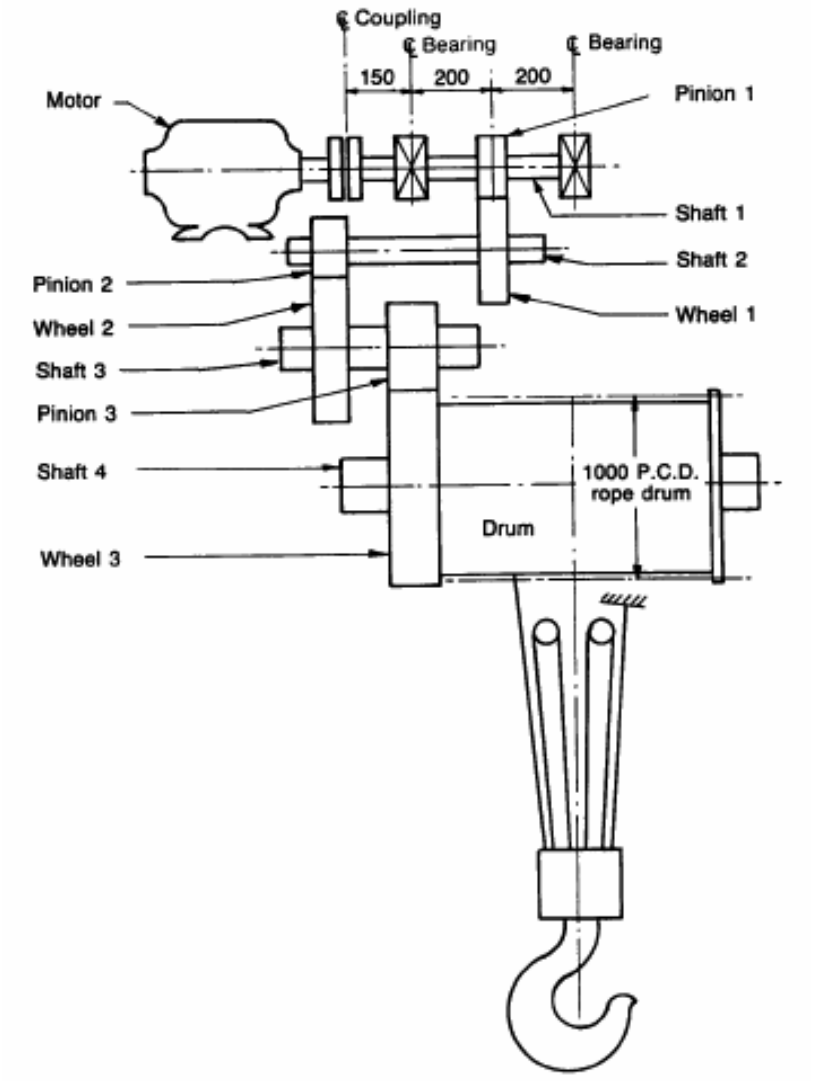


Figure 2: Hoist Mechanism – Components of hoist (adopted from Figure E1 ref. [3])

The above figure (Figure 2) illustrates the key components of the hoist mechanism, necessary for consideration for the braking torque calculation. In equation (2), we see two terms related to moment of inertia, one is related to an equivalanced moment of inertia of rotational masses and the other, related to total linear moving masses on the hoist mechanism. As per Figure 2, we see the hoist mechanism is divided into gearbox stages. The first shaft features the motor, shaft coupling and 1st pinion, all masses present on the first shaft must be added together to give a total inertia on the common shaft. Similarly, the 2nd shaft must summate the 2nd pinion and 1st gear wheel respectively. In case where the calculated shaft is the last shaft, the hoist rope drum must also be taken into account. We may denote these total inertias of these individual shafts as J_i [kg.m^2], where i=1,2,3,4 of each shaft respectively. Then the equivalent moment of inertia for each shaft with respect to the first shaft may be calculated via the following expression;

$$J_{eq.} = J_i \left(\frac{N_i}{N_{drive}} \right)^2 \tag{3}$$

Equation (3) applies to a general layout of a hoist mechanism, where the 1st pinion is driven on the same shaft line as the hoist motor. N_i denotes the rotational speeds of the respective shafts [rev/s], with N_{drive} being the rotational speed of the motor or in case where the motor is applied via a chain or belt drive configuration, the rotational speed of the 1st shaft [rev/sec].

In the sense of linear moving masses present on the hoist, we must also equivalence this as follows;

$$J_{eq.\_lin} = m_i \left(\frac{V_i}{2\pi N_{drive}} \right)^2 \tag{4}$$

Equation (4) denotes the equivalanced moment of inertia of the linearly moving masses as individual components. Here, m_i denotes mass of the component and V_i is the moving (lifting) speed of the load or mass element [m/s]. Once established, these moments of inertia in a linear moving and rotational moving sense, respectively, may be summed together to provide the final form of total moment of inertias as in equation (2).

A further consideration may be drawn into evaluation of the braking torque over various sections of the hoist mechanism, assuming we know the overall braking torque or a set value of the total braking torque. As the braking torque itself as a whole, is present over the braking elements (i.e; brake lining). Thus, to individually assess the braking torque over the individual hoist components, we may break the hoist down to the being and ends of the hoist shafts, or inputs and outputs of the hoist shafts, respectively. With this, each input and output of the various hoist shaft stages may be calculated as follows;

$$B_{Ti\_in} = \left[T_{out(i-1)} \right] . \frac{N_{i-1}}{N_i} . \frac{1}{\eta_{i-1/i}} \tag{5}$$

Equation (5) depicts the braking torque of the individual shaft (B_{Ti_in}) at input, where $\eta_{i-1/I}$ represents the mechanical losses between the previous shaft and the current shaft under calculation. Furthermore, $\left[T_{out(i-1)} \right]$ stands for the braking torque of the previous shaft output, considering the subtraction of torque due to inertias of the previous shaft. Thus, the 1st shaft would simply feature

the input of the total braking torque itself (B_T) of the hoist mechanism, and the output braking torque at the end of the first shaft would be as follows;

$$B_{T1\_out} = B_{T\_total} - (\sum J_{rot.\_1} + \sum J_{lin.\_1}).\varepsilon \tag{6}$$

Where on knowing the total braking torque as a measured or set value, it may be subtracted by the sum of rotational and linear moving inertias, with respect to the given shaft. Thus, the general form for braking torque at the outputs for the various stages of the hoist mechanism may be derived as follows;

$$B_{Ti\_out} = \left[T_{out(i-1)} \right].\frac{N_{i-1}}{N_i}.\frac{1}{\eta_{i-1/i}} - (\sum J_{rot.\_i} + \sum J_{lin.\_i}).\frac{N_1}{N_i}\varepsilon \tag{7}$$

Here from equation (7), we see that on each output of the hoist shaft under calculation, the value of braking torque is further subtracted by the influences of inertia of the given shaft under calculation. Where, all variables are analogical to the above.

Till now the calculation of the dynamical braking torque has been reviewed, however, another form of braking torque for analysis on an industrial hoist mechanism, is the static braking torque. The method behind determination of this static braking torque is rather simple once knowing the dynamic braking torque as a total. According to standards [1]-[3] the static braking torque may be determined as a factor of the dynamical braking torque, as follows;

$$B_{T\_total\_st.} = \frac{1.6}{1.2}.B_{T\_total\_dyn.} \tag{8}$$

Equation (8) depicts the relation between the static and dynamical braking torque. This is prescribed via standards due to the complex nature of determining the static braking torque. In theory, the values of static and dynamic braking torque, is in similar proportion to the coefficients of friction at static and dynamic states. However, these values are complex to determine concretely due to wearing, temperatures and humidity varying on the brake lining. Thus, equation (8) is empirically determined for a range of types of brake lining and is thus an acceptable relation for determining static braking torque in the scope of this thesis.

From here we now have two missing elements in accordance with (2), the angular deceleration and the torque of mechanical components, which will subtract from the braking torque to give us the real value. These components must be measured, and it is indeed the focus of the following subsections of this thesis, to determine and analyse these values.

3. Solutions to Real Braking Torque Measurement and Analysis

This chapter presents the experimental setup behind measurement and analysis of the braking torque of an industrial hoist mechanism. Section 3.1 presents the hardware which I have setup and programmed for experimentation on the investigated hoist mechanism, the Gude GSZ 200/400. The presented hardware setup is of a low-cost setup (estimated cost of the electrical equipment=1250Kč including cabling, Breadboard and the Raspberry Pi processor (at 900Kč). The presented sensors of which were purchased accordingly; PIC16F628A=56Kč, CNY70=19,40Kč, TCST2103=32,10Kč, DS18B120=55,00Kč and the used resistors valued at 2Kč per piece), whilst still providing adequate measurement of the necessary hoist mechanism data. Section 3.2 presents the new application which I have designed and programmed for measurement and analysis of the braking torque for a general sense of industrial hoist mechanisms, produced in Python 2.6 with wx.Python, Numpy, Matplotlib and RPi.GPIO Modules ([9]-[12]). With the experimental results of the measurement following in section 3.3 and the resulting braking torque analysis from measured data produced in section 3.3, shown later in section 3.4.

3.1 – Hardware Setup for Measurement

This subsection outlines the experimental setup of used hardware in this thesis, with focus on the principle of measurement for determining the real braking torque of a typical industrial hoist mechanism. Industrial hoist mechanisms range from several hundred kilograms of rated capacity into the order of tonnes due to the size of the application in the industry. For the scope of this thesis I will perform my experimentation on the Gude GSZ 200/400, rated for 200kg maximum lifting capacity on single reeving. This type of hoist mechanism was chosen due to the accessibility for measurement, good documentation, along with the size of the hoist being manageable for conducting the experimentation, which will follow in the next sections of this thesis. The specifications of the used hoist are given as follows;

Max. Rated Capacity	200kg (single reeving)/400kg (2/1 reeving)
Lifting Speed	8m/min
Height of lifting (single reeving)	11m (single reeving)/5.5m (2/1 reeving)
Hoist Motor Power	750 W
Supply Voltage	230V – 50Hz

Table 1 – Gude GSZ 200/400 Hoist Specifications – (modification of Table 1)

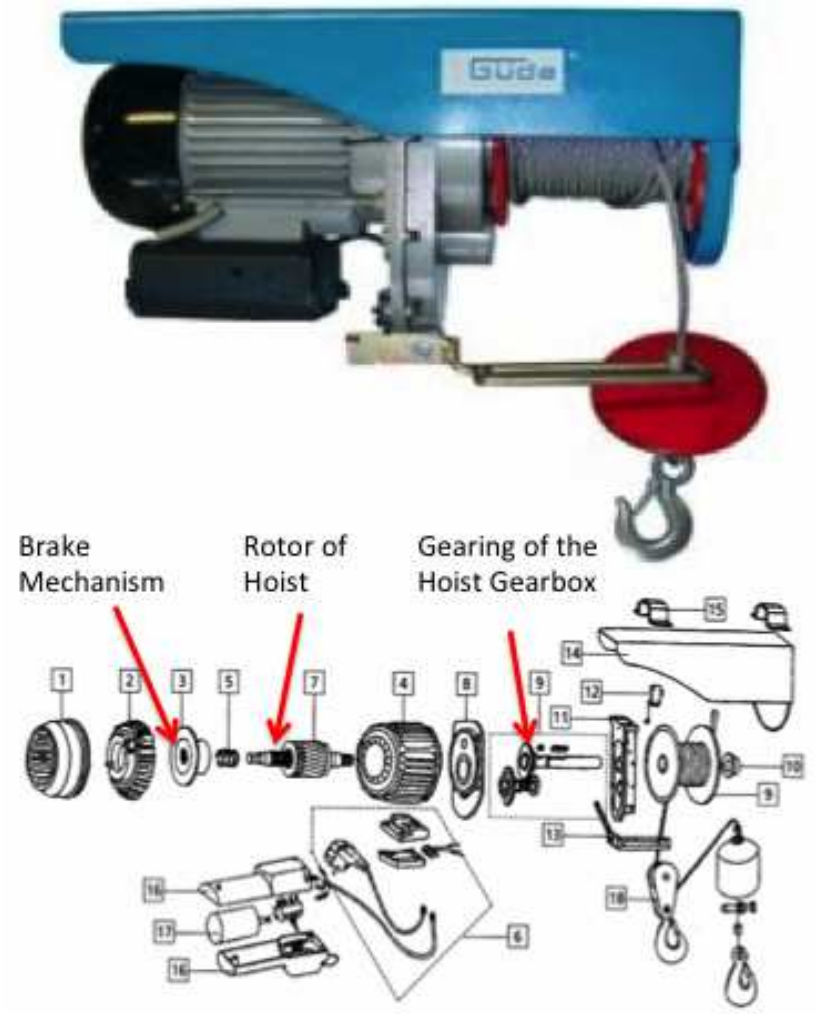


Figure 3: Assembly View and Full View of Gude GSZ 200/400 Hoist Mechanism (adopted and modified from [6])

In the above figure (Figure 3), the assembly view and full view of the Gude GSZ 200/400 hoist mechanism is shown. For the scope of this thesis, the areas of key interest lie at the hoist motor and brake system, marked from items 2-8, along with the gearbox elements at 9. In this hoist mechanism a single phase AC motor is used rated at 750 W/3.3 A, over 230V – 50Hz frequency. From the mechanism, we may identify that the used motor is in fact a size 71 standardised AC motor, with 2 pole configuration at 3000 rpm speed. It is also necessary to determine the used gearing configuration for correct determination of the reduction ratio, at each stage of the gearbox mechanism. This is vital in correctly computing the rotationally moving masses as required for determination of the braking torque (ref. equation (2)), thus access to the gearbox casing is an important feature of the chosen hoist. In larger industrial hoist mechanisms, the gearbox features an inspection hole, at the gearbox casing and this would be an important element for access during temperature measurement of the gearbox casing and oil temperatures.

In this thesis, there are two key components to measure, the first is the deceleration of the hoist with the second being temperature of the gearbox, for analysis of influences of gearbox oil viscosity on the value of braking torque in larger industrial hoist mechanisms.

Thus two measurements must be implemented. The first is focused on capturing the revolutions of the motor as the speed of the motor decreases with time, during the deceleration process.

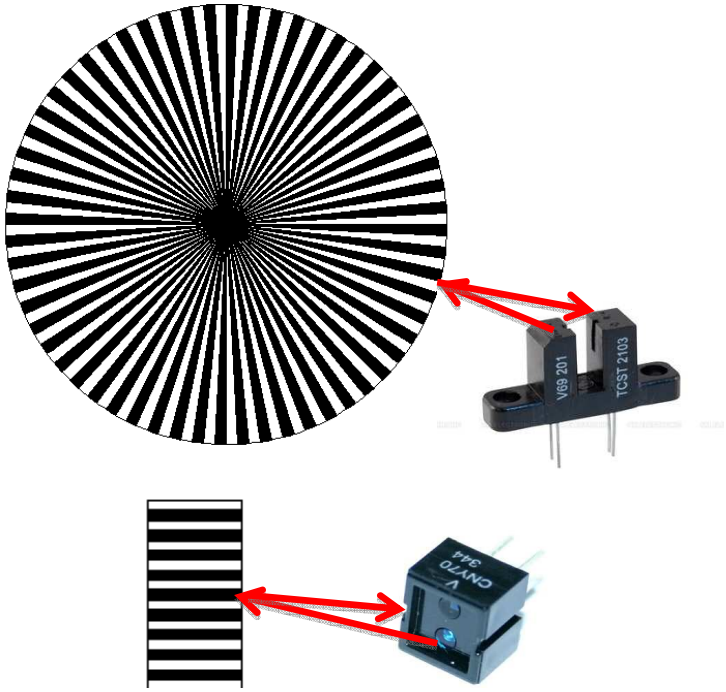


Figure 4: Principle of Measurement – Angular distance in time measurement

The above figure (Figure 4), shows the principle behind measurement of the angular distance of the hoist motor through time of deceleration. The hoist can be mounted with one of the two possible configurations as seen in Figure 4, the upper image is that of a coded wheel, spaced with equally divided increments of 3 degree, spaced squares. As the hoist motor rotates the transitions are detected via the optical reflective sensor as seen on the right. In principle, the optical reflective sensor works by emitting a low intensity light source repeatedly to the squares on the coded wheel, the collector end receives the incoming reflections with a setting of low voltage being triggered at every black square and high voltage on every white square of the coded wheel, or may be programmed vice versa. Similarly, the second configuration on the bottom of Figure 4 is possible, with the coded track rather than being mounted as a wheel, would be mounted as a coded ring on the end of the hoist motor shaft. Depending on the best configuration due to accessibility on the hoist motor, the most suitable configuration may be chosen. In the scope of this thesis, both methods will be used in the results section of this work also as a comparative measure to ensure functionality of the used hardware (for specifications of the used optical reflective sensors see appendix section).

As the hoist motor undergoes deceleration, real time measurement must be made to capture each coded square of the coded wheel (or ring) and detect the time between the transitions, as the hoist decelerates to a full stop. Thus, to ensure fast and accurate operation of the used sensor, I have implemented a connection board featuring a PIC16F628A microcontroller [15] for connection with the CNY 70 or TCST2103 optical reflective sensors [13]-[14], as used in this work.

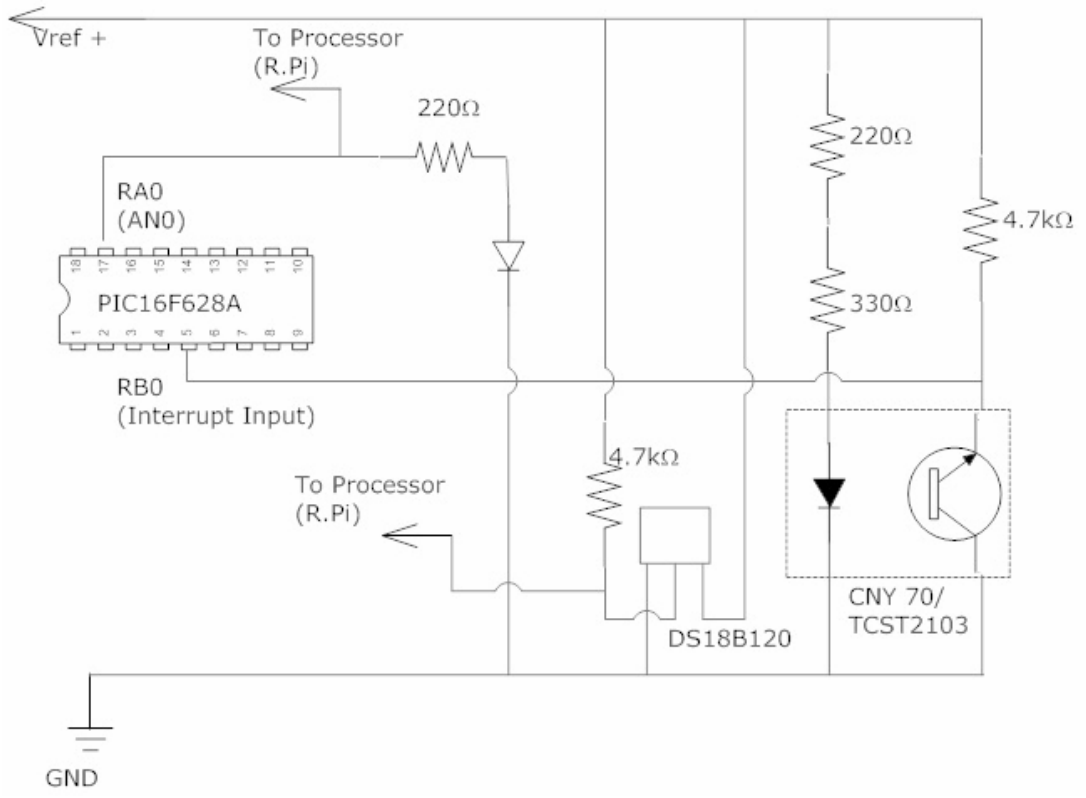


Figure 5: Circuit Diagram of the Connection Board for Connection of Sensors to Processor (Raspberry Pi)

The above figure (Figure 5), shows the circuit diagram of the connection board, which I have implemented via a breadboard, for connection of the necessary sensors to the Raspberry Pi processing unit, after review of the necessary hardware requirements in accordance with the CNY 70, TCST2103 and DS18B120 manufacturers datasheets [13]-[16]. To ensure sufficient response time of the optical reflective sensor (CNY 70 & TCST2103) with the coded divisions, 550Ω ohms was chosen at the anode according to the response characteristic from the manufactures documentations [13] & [14]. Once this optical reflective sensor is triggered, these pulses are relayed to the PIC 16F628A microcontroller. This microcontroller is programmed to raise an interrupt flag, every time the optical reflective sensor is triggered, via a transition between the coded squares on the hoist motor shaft. This is then able to be programmed to one of the PIC output pins in this case pin RA0 and be relayed to a processor for data recording and analysis, in real time. As a further visual aid, a simple LED is connected in series to the RA0 pin, to glow when high levels are triggered at the PIC microcontroller output.

As for the second component, the temperature measurement, we may directly use Raspberry Pi. Here the sensor may be read through direct connection with the GPIO pins on the Raspberry Pi and connection with the temperature sensor port located on the implemented connection board. The implemented temperature sensor in this thesis is the Dallas DS18B120 [16], this sensor was chosen for its adequate temperature range and good compatibility with the other used hardware. The operating range of this sensor is -55° to +125° C, with operating voltage range of 0.5 V – 6 V.

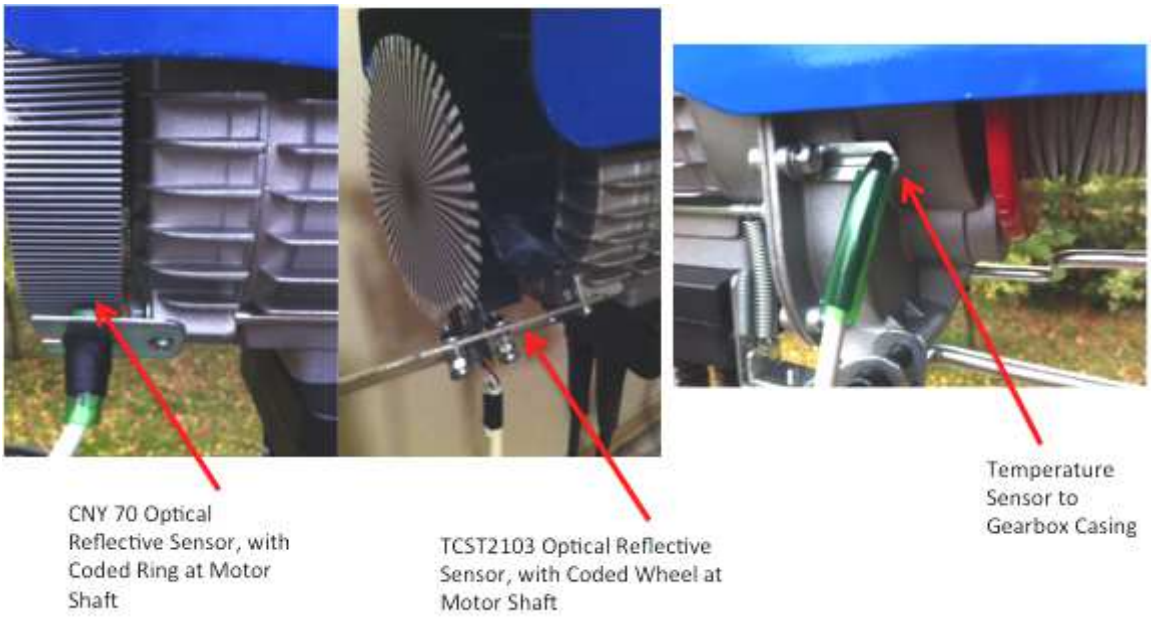


Figure 6: Sensor configuration on Hoist Mechanism

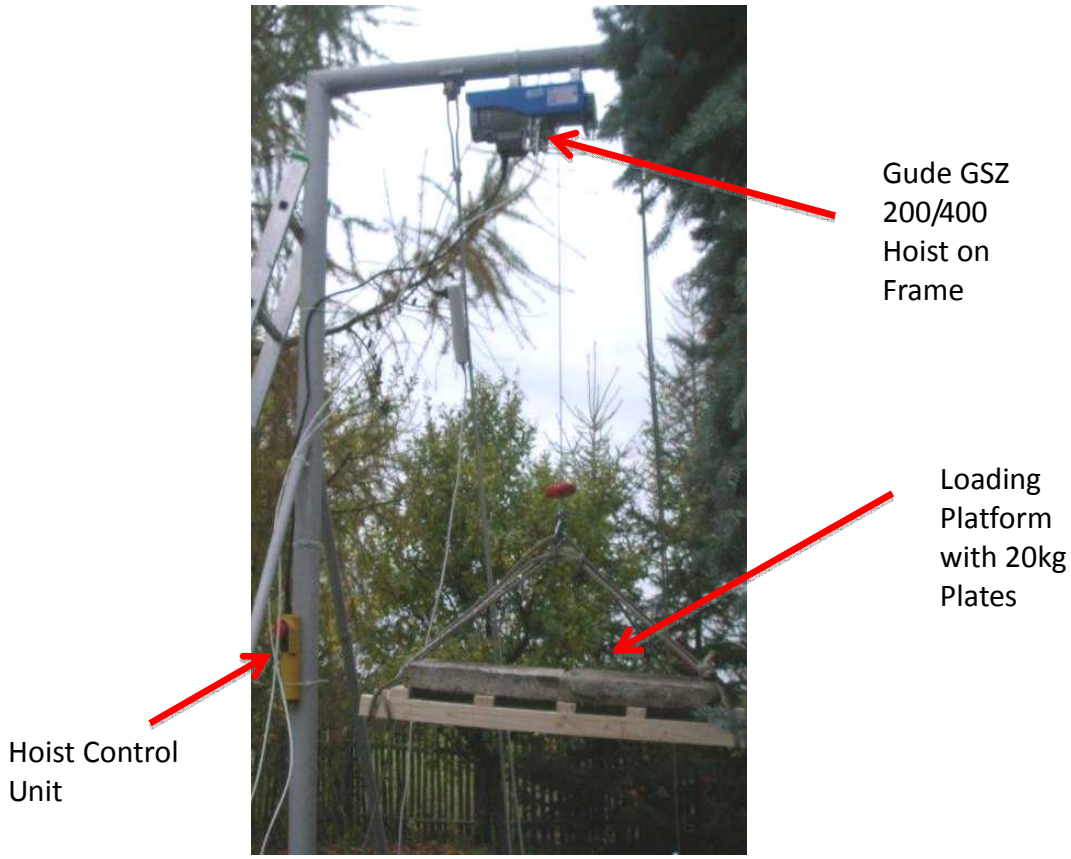


Figure 7: Experimental Setup – Hoist Mounted on Frame with Load

The above figure (Figure 6), shows the setup of the used sensors with the Gude GSZ 200/400 hoist mechanism. Here the methods of measurement used were via a coded ring, with the CNY 70 sensor used to detect the increments of coded track around the hoist motor shaft. Furthermore, as a comparative measure, the TCST2103 optical reflective sensor with coded wheel was implemented. On the right, we see the DS18B120 temperature sensor mounted on the hoist mechanism. Here the sensor was insulated and mounted to the gearbox casing. In practise, with this used hoist, the gears mounted in the hoist gearbox are lubricated via grease, rather than oil. Thus, for the scope of this thesis it is enough to measure the gearbox casing temperature for the corresponding deceleration measurement, as the deceleration characteristic would be negligibly changed. In larger industrial hoist applications where there are varying operating temperatures with an oil lubricated gearbox, it would be important to consider the temperature measurement and hence why it is implemented in this measurement (see appendix section, under braking torque application code lines 338-387). However, the results of this thesis will be focused on a single temperature setting, with focus on the effects of braking torque and deceleration due to varying of load.

From Figure 7, we can see the overall setup behind the measurement of the hoist mechanism. Here the hoist is mounted on a 4m high frame, structurally well within the strength limits of loading for this measurement. It is necessary to ensure a good working height in order that the hoist may fully accelerate as well as decelerate within a safe working distance. The hoist is configured with a single reeved configuration (direct winding onto the rope drum). The Gude GSZ 200/400 is also possible to configure as a 2/1 reeving with 400kg loading capacity, however for safety and handling, this thesis will focus on measurement with a single reeving setup, meaning a maximum working load of 200kg. The bottom block of the hoist thus features a loading platform, capable of holding this maximum rated load, loaded with 20kg concrete plates accordingly, as the linearly moving mass at measurement.

3.2 – Overview in Design & Use of Application

The purpose of this subsection is to outline the design and development of this new unique braking torque analysis application for real-time measurement and analysis on industrial hoist mechanisms, in regards to their real braking torque. Further to this, I will also demonstrate the usability of this new application, with the results as presented in section 3.3-3.4, performed on the Gude GSZ 200/400 hoist mechanism. The goal is to create an application where the engineer or trained user, who is investigating the braking torque of an industrial hoist mechanism, may either initially measure or load previously measured data into the application and perform the necessary calculations in determination of the braking torque. Here too the application should provide visualisations of the measured data as well as results of the braking torque analysis, and to also compare whether the hoist under investigation is within the prescribed limits of braking torque, both in a static and dynamic sense as according to standard hoist codes.

Recalling equation (2), we can note that in order to determine the real braking torque of a hoist mechanism, we must determine the deceleration of the hoist mechanism and further to this the torque due to the mechanical components of the hoist. Thus, the essence of measurement lies within determination of the hoist deceleration. Separately to this, various factors such as the overall mechanisms resistance in suspending the load, in the event of its braking system being disengaged also influences the accuracy of the braking torque value. Furthermore, operation of the hoist mechanism at substantially different temperatures can too also affect the overall braking torque in

the sense of viscosity of the gearbox fluid being different and thus too affects the value of the deceleration of the hoist mechanism as well as the torque holding the load due to the mechanical components of the mechanism. In this application, two main measurements are thus to be analysed. The first is the deceleration of the hoist and the second being the temperature of the hoist gearbox at operation. Given these key pieces of data, the braking torque may be determined via programmed algorithms in the application that are based on the previously reviewed theory in section 2.1.

Figure 8, depicts the overall layout of the application. Here the main interface is divided into several key panels, these panels correspond to the process behind determination of the braking torque. The first panel in Figure 8 is dedicated to all necessary general data about the hoist mechanism under investigation. The user must specify the hoist motors speed as well as the lifting speed, furthermore the rope drum diameter and mass of the load at testing.

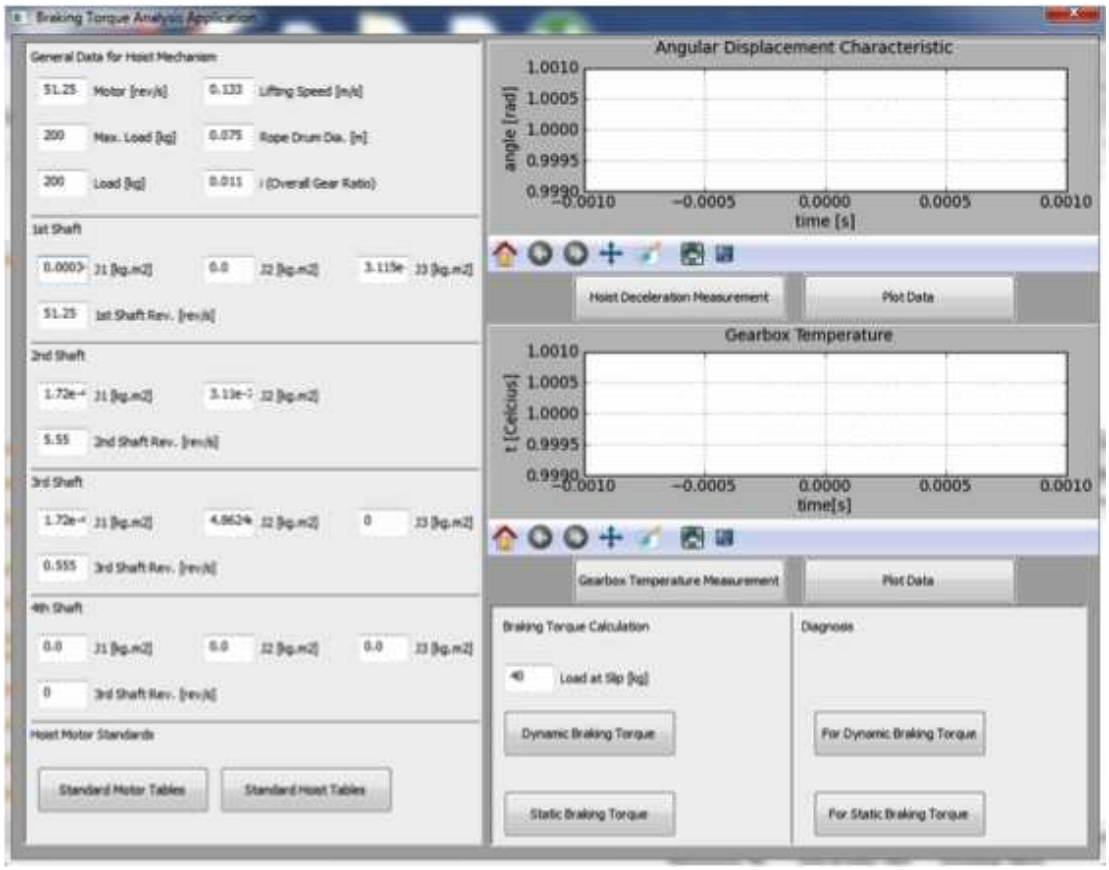


Figure 8: The Designed and Developed Application for Braking Torque Measurement (Braking Torque Analysis Application)

Another key piece of data required is the moment of inertia of each rotational mass featured on the several stages of the hoist gearbox. It is necessary to compute an equivalence of these rotational inertias with respect to the 1st shaft as according to equation (3). Thus all speeds as well as all moments of inertia on the respective shafts must be considered. As for computation of the inertias of linearly moving masses (equation (4)) on the hoist mechanism, the necessary data is supplied in the general data section of the application. Should the user be unable to identify the hoist motor on their investigated hoist mechanism, standard motor tables are featured in this application to cover a general range of typically used industrial hoist motors.

On the right side of this interface, we see two internal plots. These plots were designed to visualise the measured data of the hoist system. On the upper half of the screen the user may analyse the measured deceleration characteristic through the plot of degrees of rotation against time. The second plot features the temperature measurement, for the working temperature of the hoist gearbox, in sense of heavy industrial hoist mechanisms where oil is present in the gearbox and may be subject to largely varying temperatures. To these plots two key buttons were thus incorporated.

These plot buttons, allow the user under the same hardware setup as I have proposed to measure the characteristics from the hoist, directly. However if the user already has data from any other means of measurement, they may directly use the plot data button for visualisation and further the calculation functions for determination of the braking torque (the loaded data must be written under the file names “rad.txt”, “time.txt”, “t.txt” and “temp.txt”, corresponding to the angular distance measurement over time and temperature measurement respectively). The lower half of the interface shows the calculation functions. It is here that I have programmed a series of algorithms, corresponding to the calculation of the real braking torque for both static and dynamical torques. A diagnosis button is also featured for both cases to compare whether the investigated hoist mechanism complies with the industrial standards of static and dynamic braking torque or not.

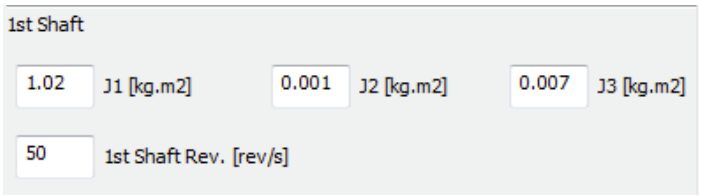


Figure 9: Shaft Data Segment of the Braking Torque Analysis Application

In the above figure (Figure 9), we see a segment of the interface corresponding to the data entry of the first shaft, according to the layout of typical industrial hoist mechanisms. In general, the first shaft includes the motor or other drive elements which may be incorporated in inertia J1, from here the user must also know via documentation of their hoist mechanism, the kind of coupling used, having an inertia corresponding to J2 with the edit titled J3, used for the inertia of the 1st pinion. In principle each shafts pinion and gear wheels must be considered as rotational masses, which adds to the overall rotational inertia of the hoist mechanism and furthermore influences the overall value of the braking torque. The user should also know the size of these gear elements and thus also the reduction ratios for each stage of the gearbox mechanism, to supply an accurate equivalence of the respective shaft to the first shaft of the hoist as in accordance with equation (3). All other rotational inertias, such as the rope drum on the final stage of the gearbox should be considered per respective shaft of the hoist mechanism.

Figure 10, depicts the main interface featured with a plot of the deceleration characteristic of the used hoist in this thesis, at a full load test. This plot is featured to re-draw the newly measured values of angular displacement over time, rather than a real-time plot. This is due to the deceleration process being too fast in certain applications for accurate real-time plotting. According the principle of measurement, as in section 3, the used method features plotting of the change of time against equal radians on the coded wheel. Thus, as the hoist decelerates we may monitor over time how the speed of the hoist abates over the same increment of radians. In more detail, regarding the plotting functions featured in this application, Figure 11 shows an example plot of the deceleration characteristic, measured on the used hoist with full loading. Here angular increments

of distance are plotted against the time during the deceleration process. A navigation tool bar is thus incorporated to enable the user to zoom, pan, adjust or save the plot as required for their use.

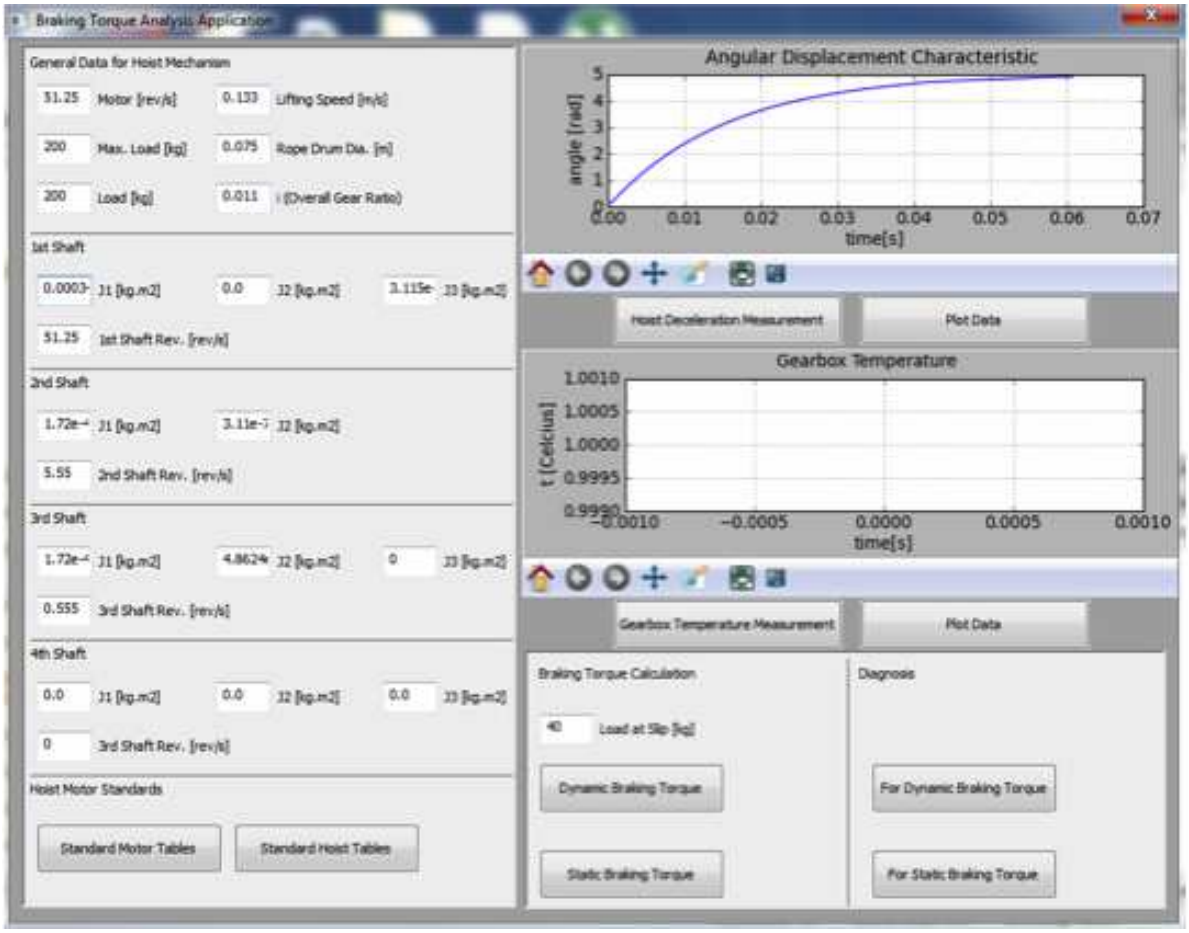


Figure 10: Braking Torque Analysis Application with Half Load Measurement

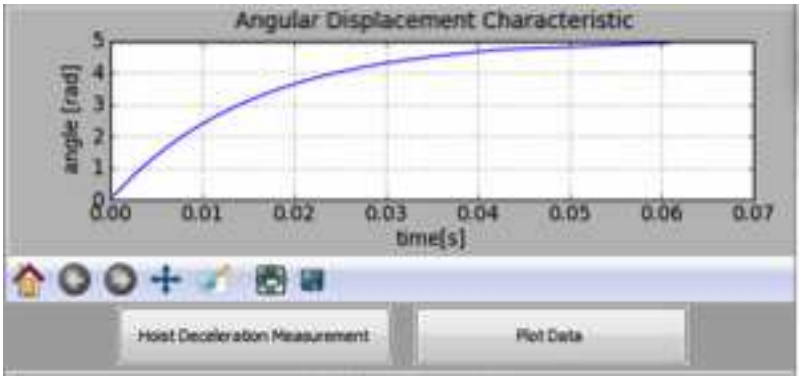


Figure 11: Braking Torque Analysis Application – Measured Angular Displacement Characteristic Due to Deceleration

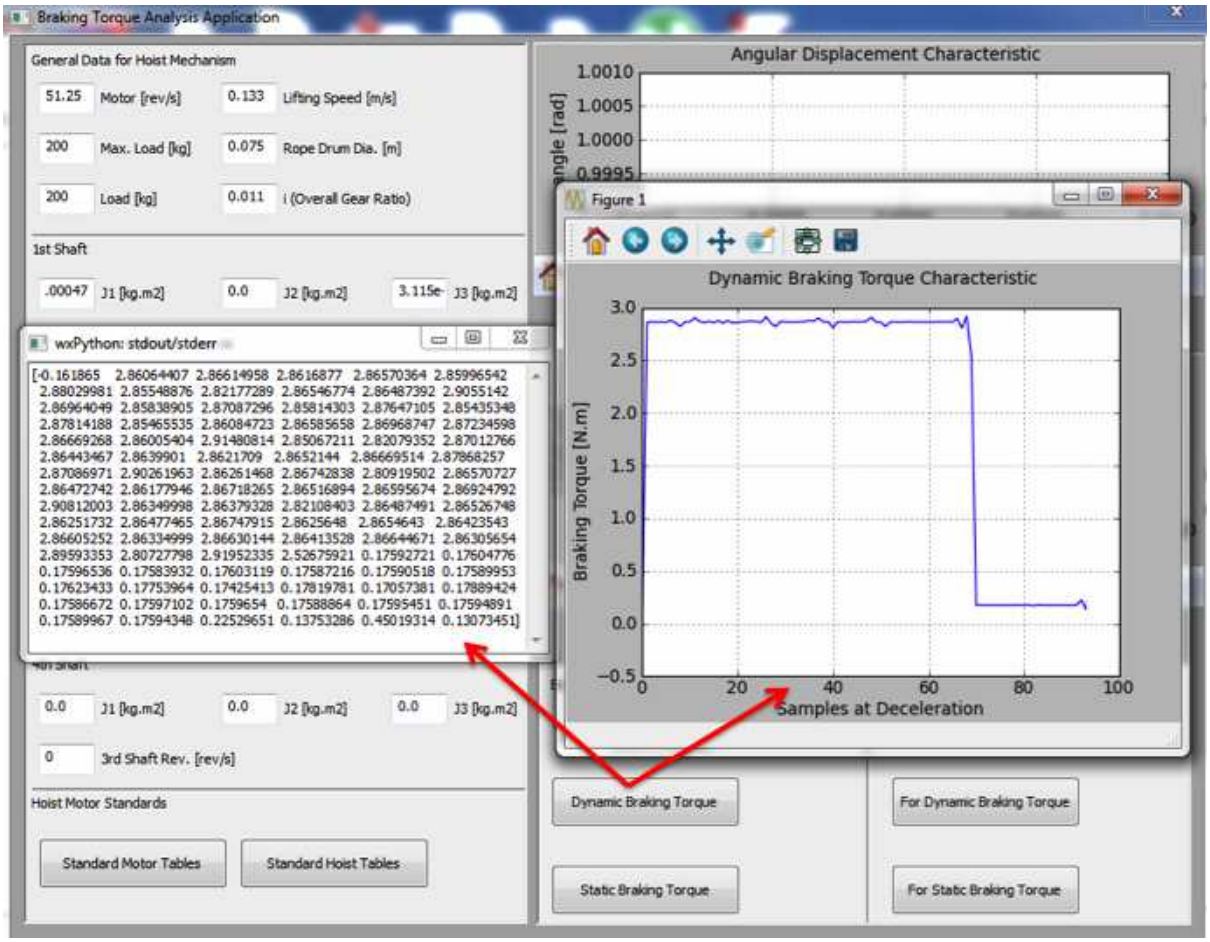


Figure 12: Braking Torque Analysis Application – Dynamic Braking Torque Characteristic

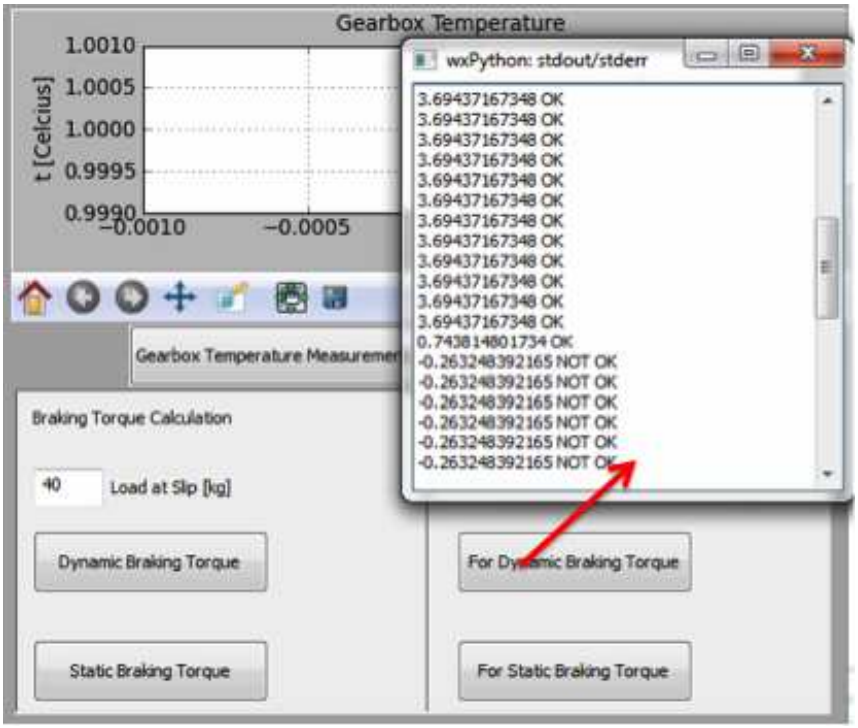


Figure 13: Braking Torque Analysis Application – Dynamic Braking Torque Diagnosis

Figure 12, shows the calculation function of the dynamic braking torque. Here the user may analyse both the static and dynamic braking torque of their hoist mechanisms via the programmed buttons on the interface. It is essential that the data files used if not directly measured via the braking torque application, should be in the path of the application folder. The user may then visualise the calculated braking torque of their hoist mechanism, via a separate external plot, which is activated via the calculation button and data window. The second result of this function is an external window, which depicts the values of the braking torque as they are calculated over each sample of the measured hoist data. Figure 13, depicts the diagnosis functions for braking torque analysis of the application. These functions are designed for both the dynamic and static braking torque respectfully. According to industrial hoist standards as such that of [1]-[3], the dynamical braking torque must be greater than 1.2 times the full rated load of the hoist mechanism used. Thus, this value must be compared with the real braking torque determined in the calculation function. Since the value prescribed in standards is the utmost minimum value allowed for the hoist mechanism, therefore each calculated real braking torque value is subtracted with this minimal value. If the result is negative, this implies that the hoist mechanism indeed has lower braking torque than what is prescribed by standards, a key reason for this may be too low deceleration thus it is another focus in this thesis to look at control of the hoist deceleration. Further to this, if the value is highly positive, this implies too high braking torque on the hoist mechanism.

3.3 – Experimental Results of the Measurement

Following the design and development of this application, for evaluation of real braking torque as presented in the previous subsection. The goal behind this component of the thesis is to analyse the experimental results, measured from a real hoist mechanism and to ultimately evaluate the real braking torque of the investigated system, via means of the previously introduced application.

The hoist used for investigation, in the scope of this thesis is the Gude GSZ 200/400. Here the maximum capacity for safe working, is rated at 200kg for single reeving. As a result of this, the experimentation is thus set for measurement at full loading, half loading and no loading respectively, with the aim to capture the difference in behaviour of deceleration at different loading conditions of the hoist, and its ultimate effect on the value of braking torque. The following results are obtained via the experimental hardware presented in section 3.1, with all measurements conducted for the lowering phase of the hoist lifting.

Figure 14, depicts the angular displacement of the hoist rotor during the deceleration process under full loading (200 kg) of the hoist mechanism. From this result depicted in Figure 14, we may see that under full loading the deceleration process took 0.15 seconds from full speed till stop. Furthermore, the overall angular distance travelled by the hoist rotor was 10.05 radians or 575°. Further, we may note the shape of the angular displacement characteristic. Here, during the first thousands of a second in the deceleration process, the hoist features quite large displacement, due to the higher speeds on the hoist rotor at the beginning of the braking process, as the process continues this time interval for travel of the hoist rotor over each division on the coded track grows. Towards the final division of the coded track, the hoist takes substantially larger time comparative to the last value, to travel the same 6° interval, until full stop is reached.

From this, we may see a direct reflection of this characteristic in the nature of the angular speed over time for this hoist mechanism under full loading.

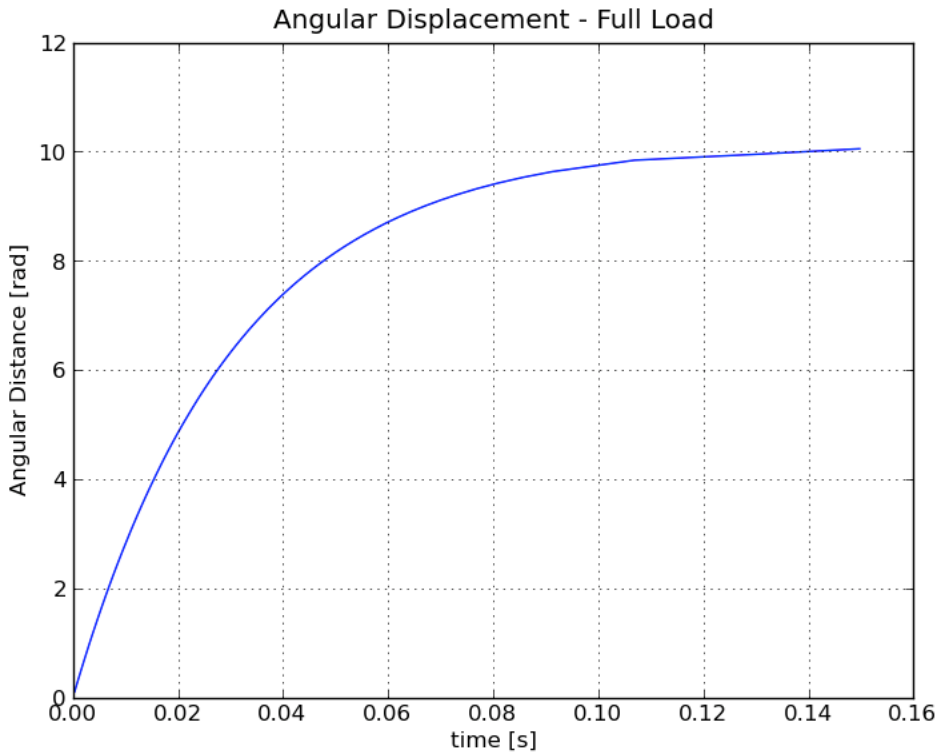


Figure 14: Braking Torque Analysis Application Results – Angular Displacement at Deceleration with Full Loading

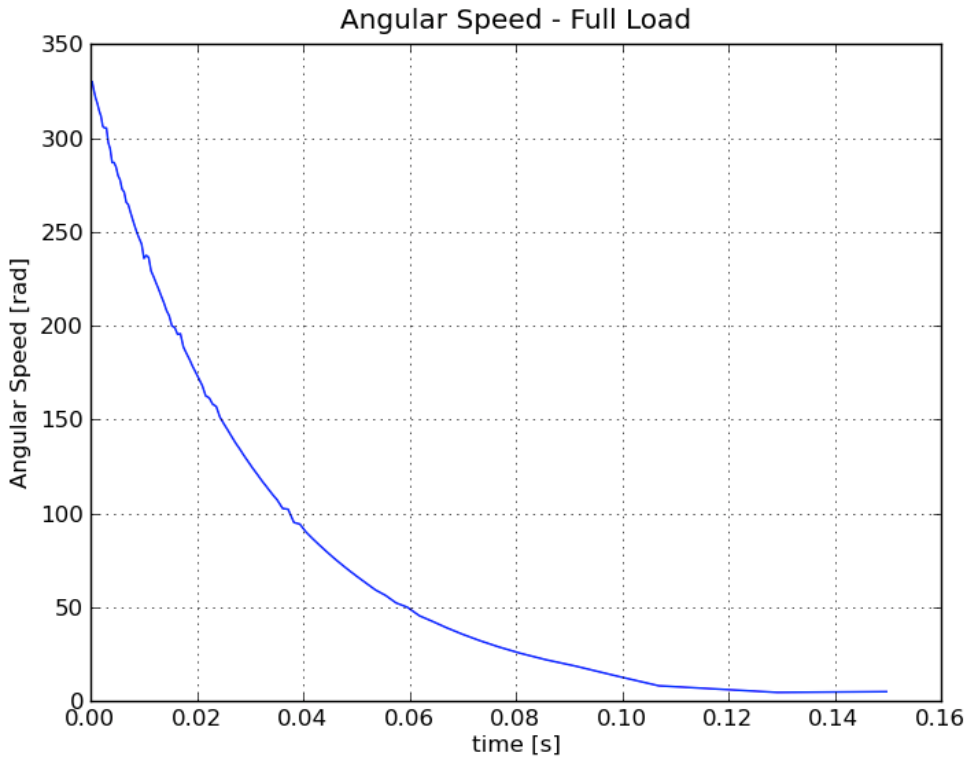


Figure 15: Braking Torque Analysis Application Results – Angular Speed at Deceleration with Full Loading

Figure 15, depicts the angular speed decay over the deceleration process for the same loading conditions as above. At the beginning of the braking process, the speed of the hoist rotor is in the region of 320 rad/s. This is due to the hoist at full speed, without any braking, being 329 rad/s at the rotor, for a 3150 rpm, taking in account the slightly increased speed due to lowering of the hoist with addition of the effects of full load. During measurement, we may observe a small set of values around the beginning of the deceleration characteristic, which corresponds to the initial samples before braking as well as reaction time of the hoist brake lining after de-energising. However, for the scope of this thesis, we are interested in the deceleration process itself and hence previous values before the deceleration are omitted. In the first few hundredths of a second, the hoist features quite a rapid drop in speed going from 329 rad/s to 50rad/s in the first 0.06 seconds of the deceleration characteristic. Following this, the speed abates to a shallow rate of deceleration till its final stop.

Overall the duration of the deceleration process is quite short, the whole process as seen in the full load measurement, lasting for only 0.15 seconds, as in the case of this hoist. This value significantly varies for the size and applications of various hoist mechanisms used in the industry. As exampled in [8], a dockyard container crane with an overall hoisting time of several minutes for the load to reach full height, may have a deceleration characteristic lasting up to 6 seconds overall, with large angular displacement after de-energising of the hoist motor. Another interesting observation from Figure 15, is the shape of the angular speed characteristic. What we may note is that at the initial stages of braking, the deceleration of the tested hoist is significantly higher than the final stages. Due to the speed drop of the first segment of this characteristic being approximately 75%, we may regard this as the main zone of braking effect. Following this, the second zone of the hoist speed characteristic features substantially lower speeds and thus, may be seen as in a creeping phase till the full stop is reached. In this phase, the deceleration seems to abate in less of a linear manner to the first component of the hoist speed characteristic. From a theoretical perspective, this variation of deceleration is a reflection of the changing coefficients of friction acting on the brake lining during the deceleration process. Where, a transition phase from lower dynamical friction coefficients to higher static coefficient values is realised, with the process being in a more static phase onto much lower speeds, corresponding to near full stop of the hoist motor.

For the purpose of evaluation of the resulting braking torque value of the investigated hoist mechanism an average of the deceleration during the various segments of the angular speed characteristic (corresponding to a line of best fit over the various portions of high and low speeds) will be taken. The key result of which will correspond to the main zone of braking in the first portion of the respective speed characteristics. Figure 16, depicts the angular displacement characteristic for half loading (100kg) of the hoist mechanism, during braking. Here we see a significant difference to the full load characteristic in terms of the overall time for deceleration and overall angular displacement during the deceleration process. From the obtained measurements on the tested hoist, the overall time of deceleration was 0.062 seconds under half load. Furthermore, the angular displacement is almost half of that under full loading, being 4.9 radians or 280°. From this observation, we may confirm this result to a general hypothesis, being that under full loading, we should expect larger angular distance travelled to reach full stop. This is due to the relation seen in equation (2), where the product of equivalent inertias with angular deceleration is such that for higher loading, in order to preserve a similar range of braking torque, the angular deceleration must be lower.

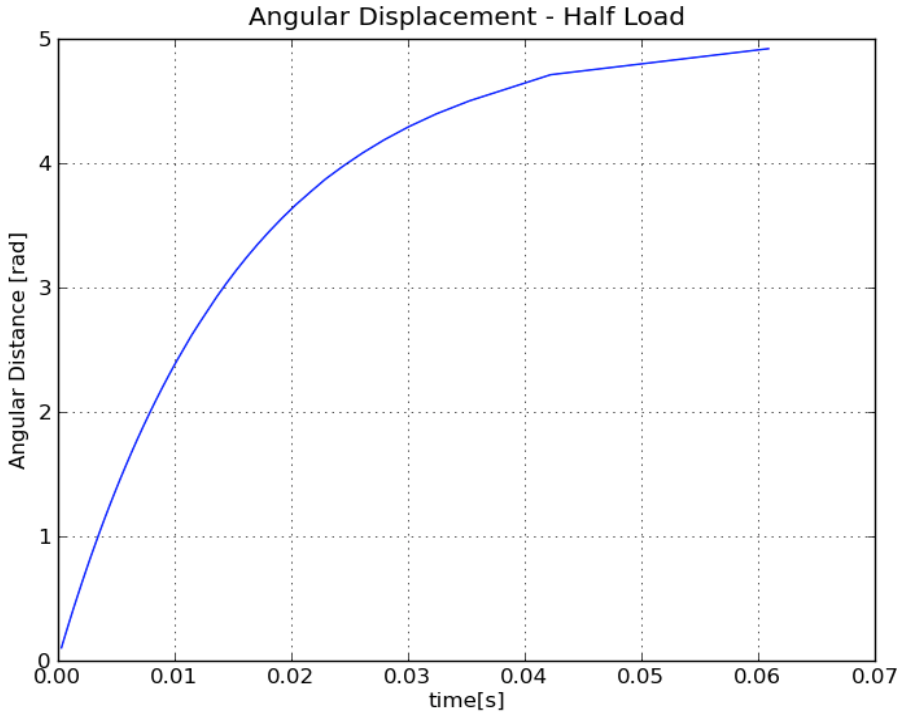


Figure 16: Braking Torque Analysis Application Results – Angular Displacement at Deceleration with Half Loading

Figure 17, shows the consequence of the previous angular displacement as an angular speed decay at half load. Similarly to the full load angular speed characteristic, the angular speed at half load follows a very similar shape. The first two hundredths of a second, during the deceleration features the hoist being in its main zone of braking. Where, from three hundredths of a second and onwards to final stopping time, the hoist reaches much lower speeds and consequently is in a creeping phase till full stop. From the measurement several minor discrepancies may be noted. Looking on the angular displacement graphs and comparing with the respective angular speed graphs, we may find that in both loading conditions, although the displacement characteristic appears quite smooth, the angular speed seem to show several discrepancies, where the value of speed between several intervals are very close or almost the same with one another. This may be due to time reading of the equipment featuring slight inaccuracies in reading. Due to the whole braking process being quite rapid, minor changes in order of few thousandths of a second may result in a significant discrepancy with respect to the angular speed characteristic. Or furthermore, may be the result of minor vibrations within the hoist mechanism or structure. In spite of this, the overall characteristic is quite well depicted in the results, clearly illustrating to us the behaviour of the hoist during its deceleration. Given the above results for both full and half loading conditions on the tested hoist mechanism, it is also interesting to test how the investigated hoist mechanism behaves under no load, being that of just the loading hook mass itself. In industrial practise this is not necessary due to there being no load present on the hoist, thus no danger can be imposed as a result of high deceleration or braking torque. However, for the scope of this thesis, it is interesting for us to note how the investigated hoist behaves during deceleration at no load, and especially to compare time and angular distance travelled by the hoist rotor for the given loading condition, in comparison with those previously investigated.

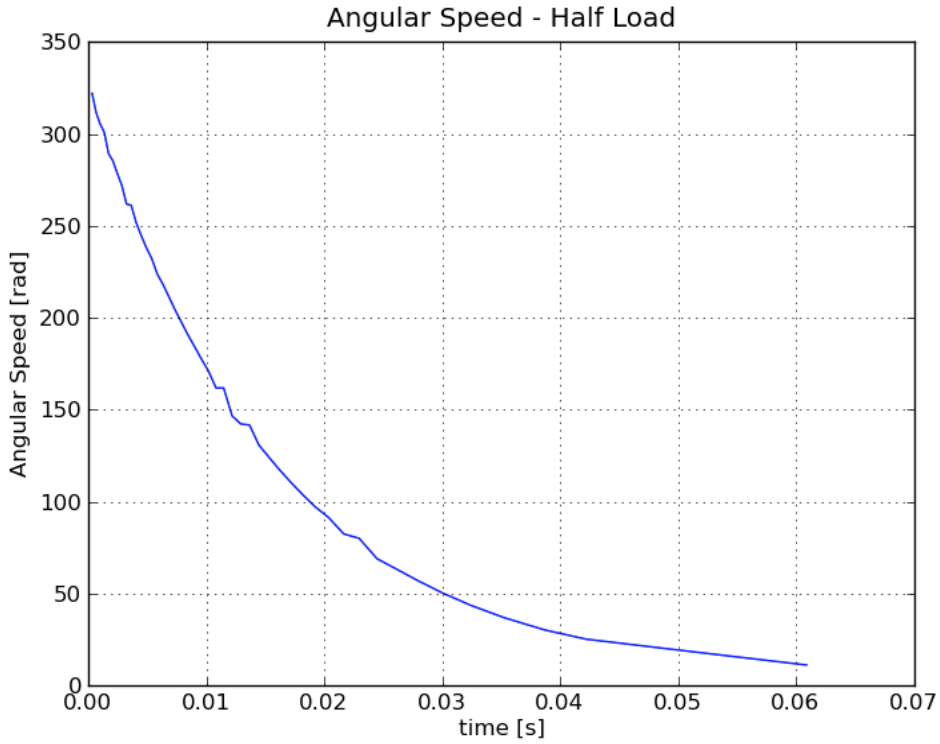


Figure 17: Braking Torque Analysis Application Results – Angular Speed at Deceleration with Half Loading

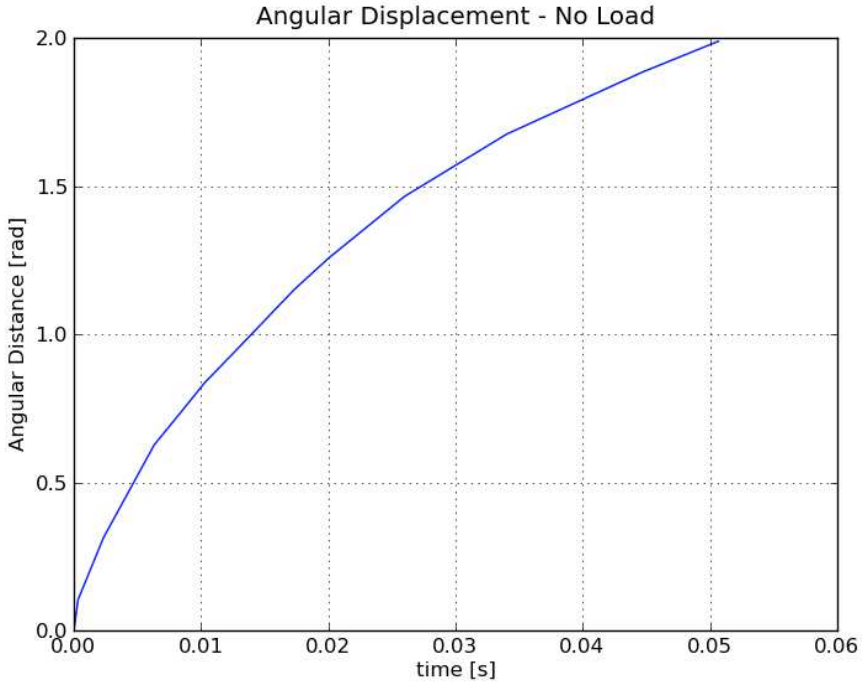


Figure 18: Braking Torque Analysis Application Results – Angular Displacement at Deceleration with No Additional Loading

Figure 18, shows the angular displacement characteristic under a no loading condition. What we may first observe is the overall time of deceleration under no load, was measured to be 0.051 seconds, being approximately one hundredth of a second faster than the half loading of the same hoist mechanism. Another key value is the overall angular distance, here we see on Figure 18, that the hoist reached 2 radians or 114° of angular displacement from the start of braking, till full stop.

Figure 19, depicts the resulting angular speed under the no loading condition. Here we may note that from the initial moments of braking, the hoist mechanism features a larger rate of deceleration as compared to the previously investigated loading conditions. Under no load, the hoist feature substantially short time of braking to reach full stop and also starting from a lower speed being in the region of 314 rad/s which corresponds to the nominal speed in rad/s of the hoist motor without loading. Here, the main zone of braking effect occurs almost instantly, the hoist mechanism within the first 0.01 seconds drops its speed to 50 rad/s and from there on, thus acts more so in a creeping phase, due to a rather shallow speed decay till full stop. Another point of interest as seen in Figure 19, as well as the previously analysed characteristics, is the final value of the angular speed results. In all graphs the final value, contrary to what we could expect to be at zero value, is here shown to finish slightly above zero. This is due to the fact that on the several last intervals of the coded wheel (or ring), travelled by the hoist rotor, featuring a slower and slower speed. Once the hoist enters the final interval, the hoist has an even lower speed than the final value plotted on the depicted charts. However, its final value is unable to be read, due to the inability to hit the next interval, where another transition and correspondingly time registration could be met.

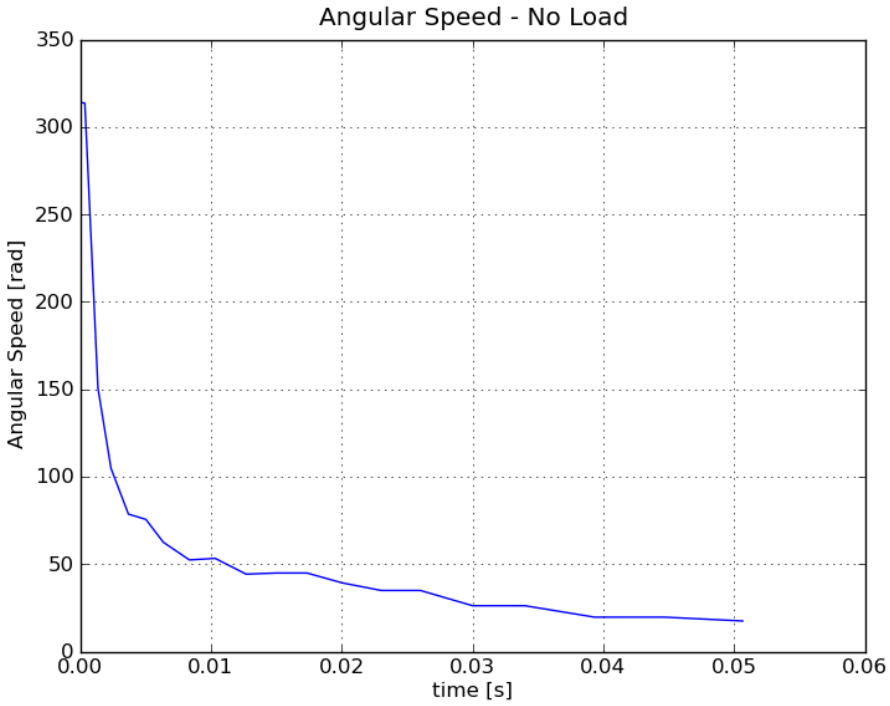


Figure 19: Braking Torque Analysis Application Results – Angular Speed at Deceleration with No Additional Loading

In the sense of Figure 19, a larger drop would have taken place from the final value to zero stop and consequently a slightly longer time. However the sensor was unable to reach the next marker

necessary to record this last value. In spite this, the characteristic in all cases is still clearly represented, with this discrepancy for purposes of braking torque evaluation, being negligible.

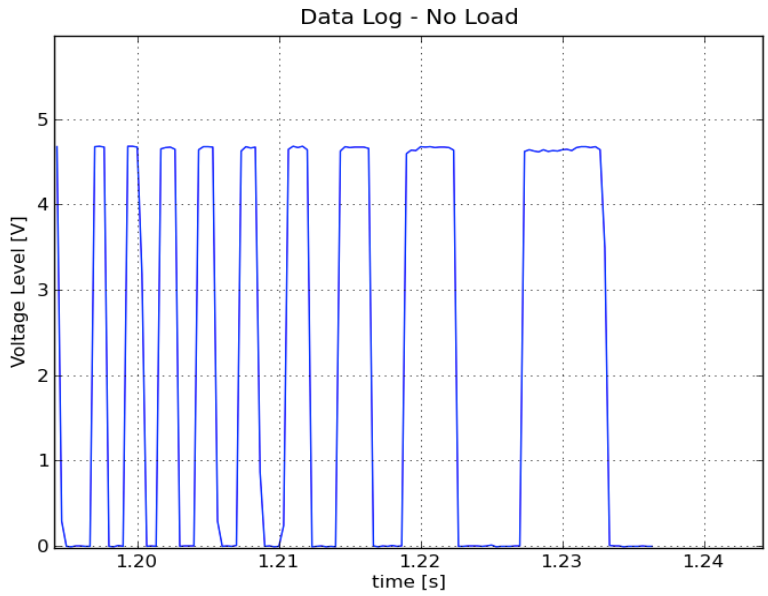


Figure 20: Braking Torque Analysis– Data Log at Deceleration with No Additional Loading (measured with Labjack U3-HV)

As a comparative measure, Figure 20 is included to show a more illustrative view of the measuring equipment during the deceleration process. Here, simultaneously as the results of Figure 18 & Figure 19 were measured, additionally the Labjack U3-HV DAQ module was used to see the behaviour of the used optical reflective sensor during this deceleration process. What we may note from this data log, is that as each interval is travelled by the hoist rotor, so too a transition is raised on the optical reflective sensor, from high to low (or low to high) voltage. As the sensor operates in the range of 0 – 5V, we may see the signal characteristic has a peak voltage at high range of +4.7 V and +0V on low range values. From Figure 20, we may note that the deceleration process happens from 1.19 seconds after start of the measurement, due to allowance of time for the hoist to reach full speed on lowering, before measurement of its deceleration characteristic till stop. Furthermore, with each interval of transition from high to low, or vice-versa, we see the gradual increase in time it takes between each interval, which is a direct consequence of the braking process.

3.4 – Experimental Results of the Braking Torque Analysis

Following the experimental results in the previous subsection, we may now determine the braking torque acting on the hoist mechanism during its respective deceleration processes. To begin, we must first establish the moments of inertias of all rotational components on the hoist mechanism, in order to develop the sum of equivalent inertias as reviewed in eq. (2)

The Gude GSZ 200/400 hoist mechanism, features a two stage reduction gearbox, with an overall gear ratio of 1/81, each stage featuring a 1/9 reduction. At first, this gearing ratio may seem quite high, however due to this hoist being a small application hoist mechanism, the manufacturing of such gearbox is aimed to be as economical as possible, and hence featuring a two stage reduction gearbox is one method of saving costs at production and is a likely reason for such design. With this said the calculations of the moment of inertias for the gearbox elements are as follows;

Calculation of moments of inertia for gearbox elements

For rotational cylindrical masses, the moment of inertia of the rotational element with its respective shaft may be calculated via the following relation;

$$J = R_g{}^2 + m \tag{9}$$

Where, $R_g{}^2$ is the radius of gyration of the respective rotational element and m, stands for the mass of the rotational element. For computation of the radius of gyration, we may refer to the following relation for cylindrical elements;

$$R_g{}^2 = \frac{R^2}{2} + \frac{L^2}{12} \tag{10}$$

Where, R is the radius of the cylindrical element and L standing for the length. Given this we may proceed to calculate the moments of inertias of the respective gear elements, in the following manner;

1st & 2nd Pinion

For R = 4.5 mm, L = 17.6 mm;

$$R_g{}^2 = \frac{(0.0045)^2}{2} + \frac{(0.0176)^2}{12} = 3.59\times10^{-5} m^2 \tag{11}$$

$$m = \rho.V = 7750\times(1.119\times10^{-6}) = 8.67\times10^{-3} kg \tag{12}$$

$$J = R_g{}^2.m = 3.59\times10^{-5}\times8.67\times10^{-3} = 3.112\times10^{-7} kg.m^2 \tag{13}$$

1st & 2nd Gear Wheel

For R = 41 mm, L = 5 mm;

$$R_g{}^2 = \frac{(0.041)^2}{2} + \frac{(0.005)^2}{12} = 8.425\times10^{-4} m^2 \tag{14}$$

$$m = \rho.V = 7750\times(2.6405\times10^{-5}) = 0.204kg \tag{15}$$

$$J = R_g{}^2.m = 8.425\times10^{-4}\times0.204 = 1.72\times10^{-4} kg.m^2 \tag{16}$$

Rope Drum

For R = 35 mm, L = 110 mm;

$$R_g{}^2 = \frac{(0.035)^2}{2} + \frac{(0.110)^2}{12} = 1.6208\times10^{-3} m^2 \tag{17}$$

Mass = 3kg (measured value);

$$J = R_g{}^2.m = 1.602\times10^{-3}\times3 = 4.8624\times10^{-3} kg.m^2 \qquad (18)$$

The final value necessary for calculation of the braking torque is the moment of inertia of the hoist motor. From standard hoist motor catalogues [4]-[5], we may obtain for the size 71 frame hoist motor as on the Gude GSZ 200/400, the moment of inertia is 0.00047 kg.m^2, including the inertia of the disc brake.

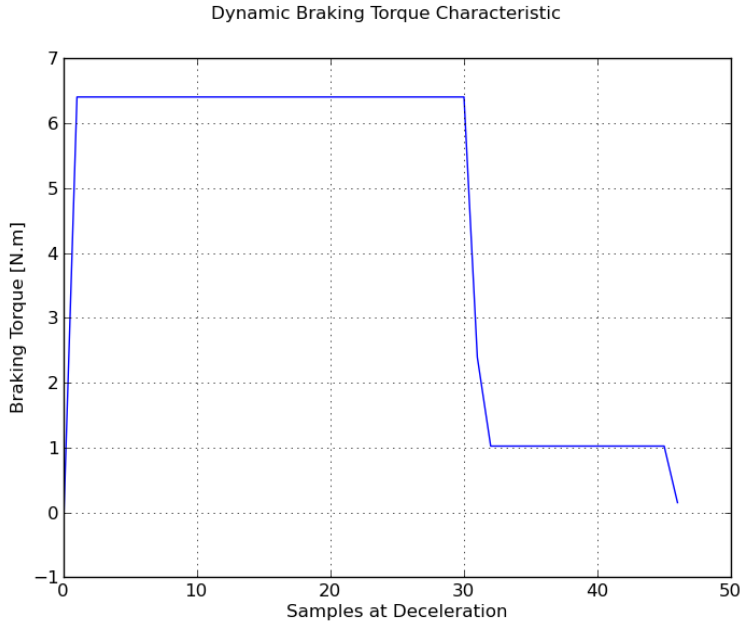


Figure 21: Braking Torque Analysis Application Results– Dynamic Braking Torque Characteristic at Half Load Measurement (considering the averaged components of the speed characteristic in different zones of braking)

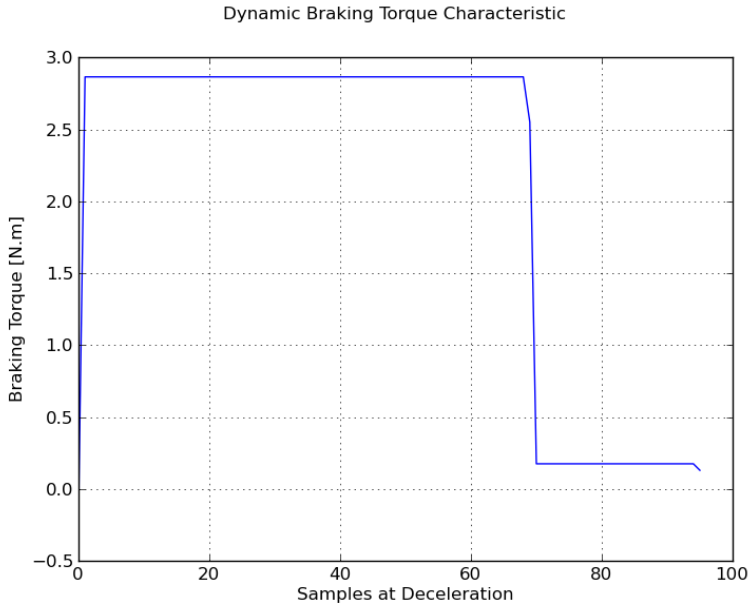


Figure 22: Braking Torque Analysis Application Results– Dynamic Braking Torque Characteristic at Full Load Measurement (considering the averaged components of the speed characteristic in different zones of braking)

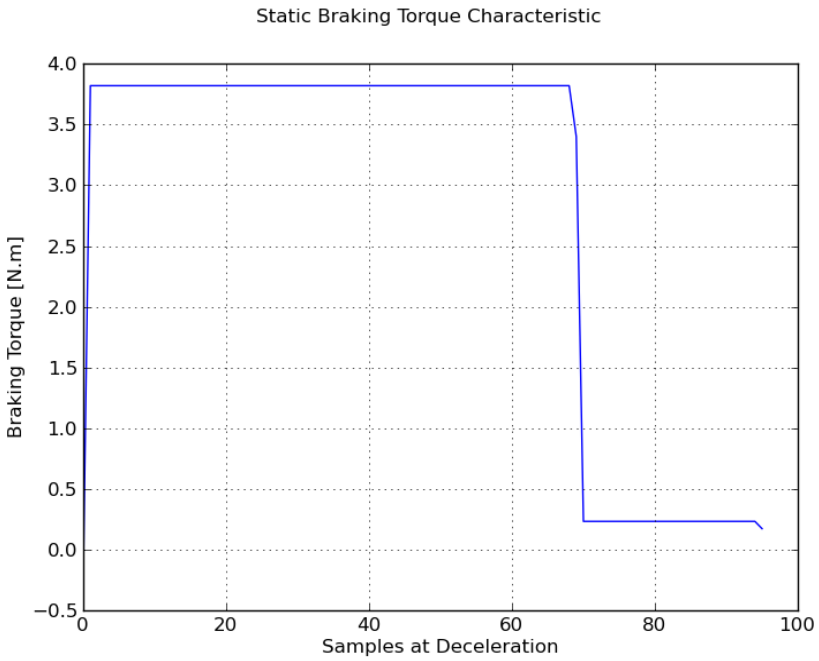


Figure 23: Braking Torque Analysis Application Results– Static Braking Torque Characteristic at Full Load Measurement (considering the averaged components of the speed characteristic in different zones of braking)

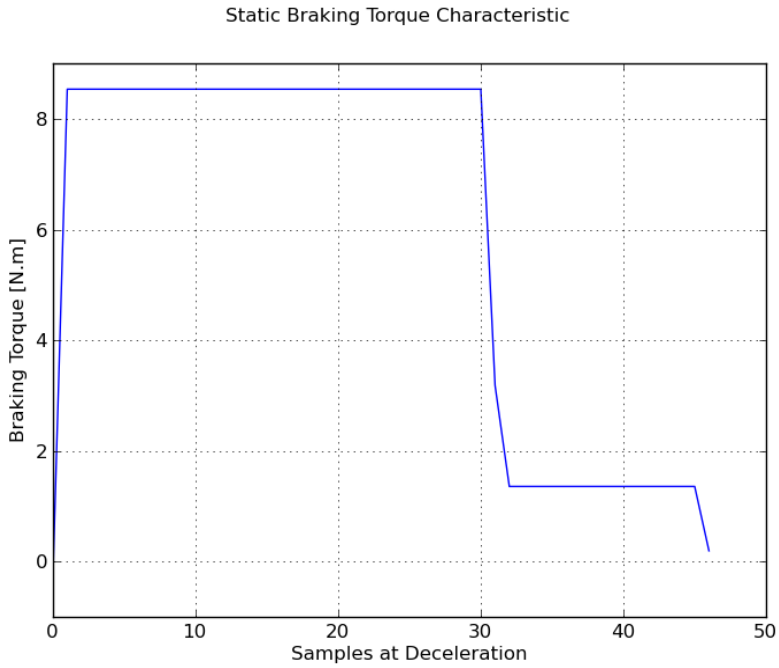


Figure 24: Braking Torque Analysis Application Results– Static Braking Torque Characteristic at Half Load Measurement (considering the averaged components of the speed characteristic in different zones of braking)

The above results depicted in Figure 21 & Figure 22, represent the dynamic braking torque characteristics at half and full load respectively. Due to the respective results of angular speed over time depicted in Figure 15 & Figure 17, we may simplify these produced characteristics into two linear components, resembling an average deceleration for the given sections of the hoist speed

characteristic, for better clarity in the braking torque analysis. In Figure 21 & Figure 22, the hoist mechanism features high deceleration at the initial stages of braking, in its main zone of braking effect and hence a high value of braking torque. According to the permissible values of dynamical braking torque, as prescribed in standards, this hoist mechanism at dynamic braking torque should not fall below 0.97 N.m, as computed for the hoist rotor, where the deceleration measurement has been made. Looking at the dynamical braking torque characteristics, the several last samples of the measured values fall below this prescribed minimum value, however at this stage the hoist mechanism is already considered to be past its braking stage, where the final values are almost at stopping point. Thus the samples of deceleration past 50 radians per second may be considered to be after the real braking effect of the hoist mechanism. With this, the overall value of braking torque would thus correspond to 6.5 N.m for half loading and 2.8 N.m for the hoist mechanism at full loading, with all values of the deceleration characteristic in the main braking effect zone, being in compliance with the prescribed standard values.

As for the static braking torque characteristics as seen in Figure 23 & Figure 24, the hoist mechanism features similar results for the braking torque. For static braking torque the prescribed permissible minimum value for the investigated hoist, is 1.308 N.m. Recalling equation (8), we see that for calculation of the static braking torque as prescribed by standards, we may compute this value as a scaling from the dynamic braking torque. Thus, a direct reflection in the behaviour of the braking torque may be drawn at a static condition. At half loading the investigated hoist mechanism has an overall static braking torque value of 8.4 N.m, in its main zone of braking. At full loading the same hoist mechanism resulted in 3.8 N.m static braking torque, which is almost three times larger than the prescribed minimum value. Though the half load characteristic better complies with the prescribed minimum value as in accordance with industrial standards, its deceleration characteristic is significantly large at high speed, and in certain engineering applications, would be undesirable. Similarly the same may be seen in the sense of the full load speed characteristic. Thus, the next chapter of this thesis is dedicated to an investigation into control of the hoist deceleration.

4. Study of Hoist Deceleration Control via a Neural Network Control System

This chapter of the thesis aims to investigate the problem of hoist deceleration control. This section is motivated by promising theoretical studies to the problem as will be reviewed in the works [18]-[19], as well as the previously presented results from section 3 of this thesis. As presented in these works the key goal is to control the rate of deceleration of the hoist mechanism, such to achieve as gentle and eased stopping as possible, which also has a direct effect on the overall braking torque of the hoist mechanism. However, till now the field of this problem still is open to further research in more optimally behaving solutions to hoist deceleration control, and especially further investigation in application with real hoist mechanisms. Thus, another key contribution behind this thesis is a new study into the potentials of a neural network based approach for solution to the hoist deceleration problem.

4.1 – Brief Review of Recent Achievements in Hoist Deceleration Control

Till this current day, investigation into the problem of hoist deceleration control is still ongoing. With recent works in this time aimed to provide further optimised solutions, to those previously presented. As far as real application of control systems extend in the industry, Those that are implemented for AC induction motors, are mostly controlled via extension of a variable frequency drive. Such drives may be programmed to gently alter the frequencies fed to the hoist motors, with the aim to decelerate the hoist motor via manipulation of hoist motor speed, and to only engage braking at a certain speed during the deceleration.

In itself, such solution is advantageous in the sense of reduced wearing of the hoist motor brake lining, which can be quite an economical loss in terms of cost for regular maintenance and repair. Along with the controlled deceleration contributing to a gentler rate of descent in speed of the hoist load, which ultimate leads to less stress build up in the hoist wire rope, reducing issues associated with fatigue and furthermore, reduction in bobbing of the hoist load and prevention of possible load damage. With this said, further advancements are under investigation to enhance the performance of this type of motor control, to achieve as optimal behaviour of the hoist deceleration as possible. With its necessity becoming more significant in terms of safety as the demands in various industrial applications are at an increase.

The first key work necessary to mention, is a study published in 2010, on variable frequency speed regulation based on a PLC fuzzy controller, for application to mining hoists. Written by Wu. Y, Tian Z. and Xie M., this work is focused on control of hoist deceleration on a mining hoist by means of a fuzzy rule based controller. In previous works to this, implementation of the conventional PID controller was realised, with the results showing that it featured high overshooting problems along with long modulation time and weak controlling effect. This work thus aimed to set out control of the mining hoist in the following manner;

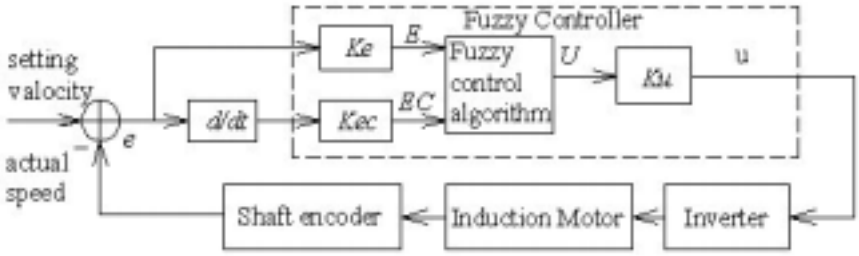


Figure 25: Structure of Fuzzy Control System for Hoist Deceleration Control (adopted from Figure 2. [19])

Figure 25 above, depicts the overall layout of the proposed fuzzy based control scheme, as applied to the mining hoist. As the hoist mechanism features a given desired behaviour of its speed or deceleration, this value is compared with the actual value of the hoist motor speed. This difference or error e, is fed into a fuzzy rule based controller along with a second input being an error variance ratio. Thus, this fuzzy based controller is a two input one output controller, with a control modifier as the output or conclusion of the rule based system. This modifier ultimately manipulates the frequency inverter to control its value as applied on the hoist motor mechanism. The conclusion

from this study was that on testing via a simulation model implemented in MATLAB SIMULINK, the behaviour of the fuzzy rule based controller featured promising results with the desired behaviour of the system, featuring very minor overshoot and more rapid response, than previously implemented the PID controller.

An addition of this study can be drawn in the work [18], written by Liu J., Wang F, published in 2010. This work features control of a mine hoist motor via a fuzzy neural network based controller, as depicted in the below schematic;

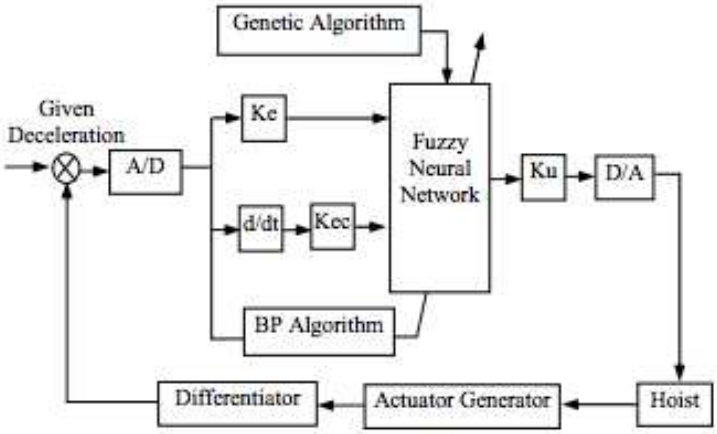


Figure 26: Structure of Fuzzy Neural Network Control System for Hoist Deceleration Control (adopted from Figure 1. [18])

Figure 26, depicts the overall layout of the proposed fuzzy based adaptive control scheme. Here instead of the hoist motor speed, the hoist motors deceleration is taken as the input values for the proposed controller. This controller is structured as a two input single output configuration, with the error in deceleration e, and change in error ec, being the first layer of the fuzzy neural network based controller.

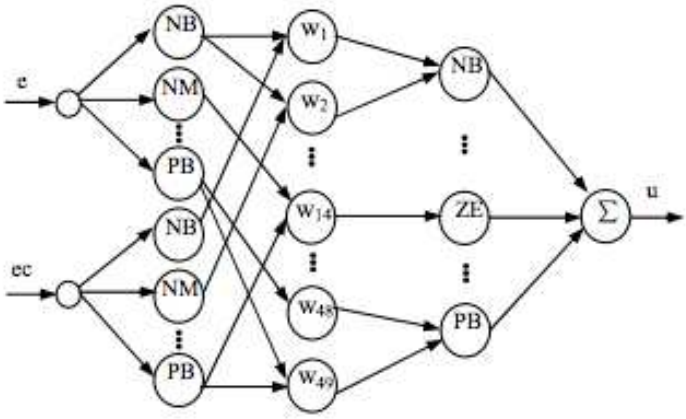


Figure 27: Inner Structure of Fuzzy Neural Network Control System for Hoist Deceleration Control (adopted from Figure 2. [18])

Figure 27 shows the inner structure behind the proposed adaptive fuzzy controller. Here the deceleration error and its change are immediately evaluated with a series of fuzzy sets corresponding to the scale of the deceleration. The third layer features the fuzzy rule layer, with weight calculation function, from this weight calculation function a corresponding output fuzzy set may be triggered. These connection weights are trained via offline pre-training and the learning parameters of the controller being online trained during simulation. The triggered result is then defuzzified to result in a control voltage force u, which may be converted to a voltage value for manipulation of the AC induction motor.

From the results of this study, superior behaviour on step response characteristic was evident, when compared to the conventional PID and fuzzy logic controllers (as seen in Figure 28 (left image)). However, an earlier study from 2003 on, higher order neural units for adaptive identification and fast state feedback control of unknown, non-linear systems, by I. Bukovsky, S. Redlapalli and M. Gupta, show that the use of high order neural units in state feedback control, can achieve even faster response for unknown non-linear systems, like the hoist mechanism in the work [18]. As seen in Figure 28 (right image), the high order neural unit gave almost three time faster response to the desired, step response characteristic as compared to other linear controller applied to the same system, without any overshoots.

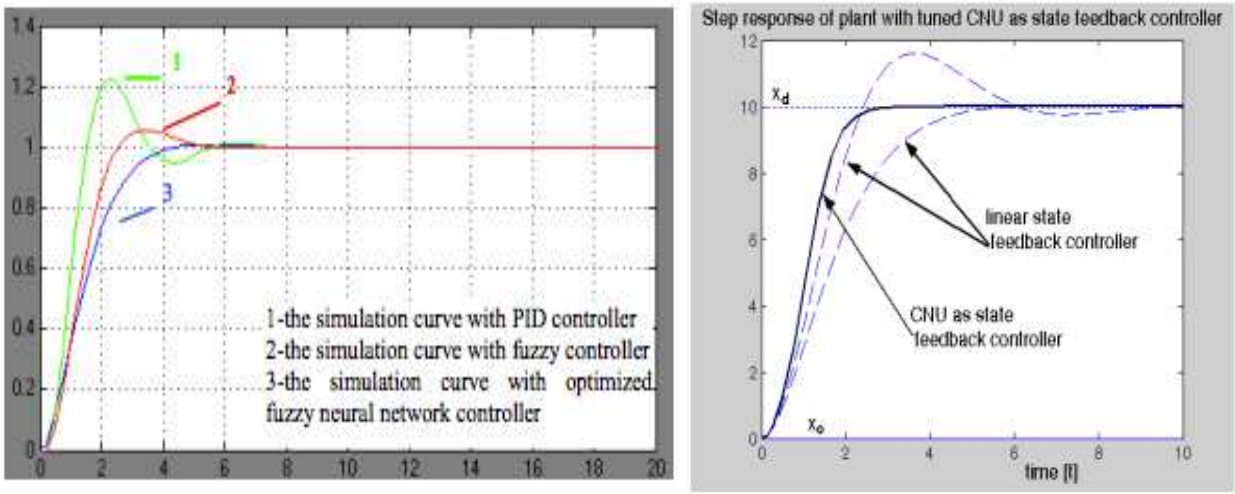


Figure 28: Comparison of Step Response with Adaptive Fuzzy Based and Neural Unit Based Controllers (adopted from Figure 4. [18] (left) & Figure 3. [27] (right))

The reason behind such response in work [27] as seen in Figure 28, can be drawn from definition of damping gains of the respective linear and non-linear state feedback controllers. Equation (19) (adopted from eq. 11 from [27]), describes a linear closed loop control equation. From this the position and velocity gains k_p & k_v, respectively, are constant and thus constrained in velocity to which it may reach its equilibrium position, with r(t) being the desired behaviour of the system. In other words we may see this as that its damping effect is defined by these constant gains.

$$u(t) = -k_p x(t) - k_v x(t)' + r(t) \qquad (19)$$

As for a non-linear closed loop control equation, a derived relation from work [27], describes that the damping effect is variable, with the concept that by gradually increasing this damping to a positive optimal value, such that the system does not surpass the desired value, faster and smoother response can be achieved. This is dictated by equation (20) (adopted from eq. 16 [27]).

$$r(t) = k_v.\left|1-\left(\frac{x_1(t)-r(t)}{x_1(0)-x_d}\right)^2\right|.x_2(t)-k_p.x_1(t) \qquad (20)$$

Here k_v represents the gain of the damping function (first component of the right side of equation (20)) and $x=x_1$, $x' = x_2$ for the non-linear equation. Furthermore, x_d is the desired behaviour for the non-linear system. This gain k_v, is thus dependant on the initial position and desired position of the system and is proportioned accordingly during application of the non-linear controller. Where, the absolute value brackets ensure stability of this controller. Thus, from this result an extended motivation arises in the potentials of application to this industrial problem of hoist deceleration control.

From the reviewed control methods of hoist motor deceleration control, though the mining hoist application does feature certain unique complexities during to the vast hoisting distances and hoisting at a slope. The principle structure of the studied hoist mechanisms are a single-input-single-output (SISO) system, with the hoist motor speed or deceleration being the output of the system and frequency or voltage variation in the sense of AC induction hoist motors, as input. From the above reviewed applications, it was desirable for the hoist to feature a constant deceleration. The dynamics of this case is unique because mining hoists feature very large motors, with correspondingly large rope drums sufficient to reel several kilometres of wire rope. Here the deceleration time is substantially large and as a consequence of variable deceleration or even too high deceleration, oscillatory motion may be induced in the rope and cabin, posing a danger to the mining operation and personnel. In other hoisting applications the desired value may not necessarily be constant however, what is key, is that the mechanism doesn't exceed the permissible limits of braking torque as prescribed by standards. Furthermore, during control, the load is not subject to too high deceleration where bobbing of the load may be induced, and in the sense of a large jib style crane application, could result in large values of dynamical forces that amplify the loads effect substantially, at deceleration.

4.2 –The Proposed Principle of Hoist Deceleration Control via a Neural Network Control System

In this section, the various neural network approaches, aimed for adaptive identification and control of the previously introduced hoist deceleration problem, will be described. These approaches are based on the well know Gradient Descent (GD) method, used as a tool for defining the learning rule of the applied neural units. These applied neural units (adaptive models) are trained via two methods, which will be focused on in this thesis. These methods are namely, the method of incremental training or Real Time Recurrent Learning (RTRL) [29] applicable for dynamic adaptive models or a batch form of training, which is a variation of the back propagation through time method[30], which in this thesis will be denoted as batch-propagation-through-time (BPTT) training, as an extension of RTRL training in combination with the famous Levenberg-Marquardt equation.

This subsection aims to review the important theories from works [21]-[23], behind the GD method and structures of the neural units used within this thesis. Firstly, we will begin by introducing the famous GD algorithm for the linear and quadratic neural units. To understand this, we must begin with the polynomial models of the Linear and Quadratic or (LNU and QNU), respectively as follows;

$$y = \sum_{i=0}^{n} x_i w_i = w_0.x_0 + w_1.x_1 + ... + w_n.x_n = \mathbf{w.x} \tag{21}$$

$$y = \sum_{i=0}^{n}\sum_{j=0}^{n} x_i x_j w_{i,j} = w_{0,0}x_0x_0 + w_{0,1}x_0x_1 + ... + w_{n,n}x_n^{\ 2} = \mathbf{rowx.colW} \tag{22}$$

Where, *y* represents the output of the LNU (21) and QNU (22) respectively (adopted from eq. 1-2 [22]). The long vector representations of **colW** and **rowx** can further be extended to achieve the CNU neural unit architecture and furthermore, even higher orders. In regards to the LNU, **w** stands for an updatable vector of neural weights and **x**, represents the input values of the engineering system in the case of a purely static model and in the sense of a dynamic model, a combination of inputs of the real system and neural model outputs. Looking at equation (22), **rowx** is a long-vector representation of the utilised input vector. Where, **colW** represents a weight matrix of the quadratic or higher order neural unit in general. From this we may then understand the GD algorithm as applied to such neural units.

$$w_{i+1} = w_i + \mu.e(k).\frac{\partial y(k)}{\partial w_i} \tag{23}$$

$$\mathbf{colW}(\mathrm{k}+1) = \mathbf{colW}(k) + \mu.e(k).\frac{\partial y(k+1)}{\partial \mathbf{colW}} \tag{24}$$

Here, equation (23) & (24) (adopted from eq. 4-5 [22]) depicts the GD algorithm for both the LNU and HONUs (as such QNU & CNU) respectively. Here the output of the GD algorithm is such, to incrementally over sample-by-sample, update the neural weights to adaptively teach the LNU or HONU model, to learn the engineering system. Here, μ represents the learning rate of the GD algorithm. This is analogical to humans where, setting μ relatively high, corresponds to faster learning of the human and consequently, means the less detail the human can remember and digest from their learning. Furthermore, setting this parameter to a smaller value, corresponds to a slowing rate of learning i.e. the human may remember the information learned, quite well, but less information overall about the subject. The next parameter is $e(k)$ (k representing the number of the sample), this represents the current error between the real and calculated output of the model. The final term $\frac{\partial y(k)}{\partial w_i}$ corresponds to the partial derivatives of the neural unit output, with respect to the individual neural weights.

Regarding the HONU in equation (24), we see that the GD algorithm is analogical, with exception of updating the matrix of neural weights as opposed to a vector in the sense of the LNU.

Till now, the structures of GD for LNU and HONU were reviewed in the format of RTRL or sample-by-sample method of learning. Where, the neural weights are updated over each new sample of the engineering system data. However for certain engineering processes, it is more advantageous to use the BPTT batch method of training these neural weights, over runs of the algorithm rather than over each sample. This is because the RTRL method focuses on the contemporary governing law of the system as opposed the BPTT which focuses more so on the main governing law of the input and outputs of the engineering system. The BPTT method is achieved via an extension of the RTRL method with combination of the famous Levenberg-Marquardt equation. It is also important to note

that this method is more preferable in cases where the data may be affected by noise. The following equation denotes the weight update rule for the BPTT method;

$$\Delta \mathbf{w} = (\mathbf{J}^T.\mathbf{J} + \frac{1}{\mu}.\mathbf{I})^{-1}.\mathbf{J}^T.\mathbf{e} \qquad (25)$$

Here, the neural weights are updated over each run of the algorithm (or batch trained) in the following way w=w+Δw. Here equation (25) (adopted from eq. 16 [22]) describes the change, necessary for the update of the batch trained weights. Where, **J** represents the Jacobian matrix of derivatives for the neural unit. This may be the complete partial derivatives of the neural model with respect to each neural weights, or in practical applications it seems useful to simply introduce this Jacobian matrix as the input vector or matrix itself, being **x** and **colx** for LNU and HONUs respectively. Where it is important to note that, $\mathbf{colx} = \mathbf{rowx}^T$.

Often in such adaptive neural units, it is apparent that a modification of the normalised learning rate may be used to solve issues of instability of learning. In practise it is possible to employ the simplified normalised learning rate as presented in the work [22], as follows;

$$\eta = \frac{\mu}{\mathbf{x}(\mathrm{k})\mathbf{x}(\mathrm{k})^T + 1} \qquad (26)$$

Where, equation (26) (adopted from eq. 9 [22]) represents the normalised learning rate in the sense of LNU. This is analogically represented in the HONUs, by replacement of **x**(k) (i.e. each k-th sample of the input vector x) with **colx**(k). It should be noted that the above representation holds for RTRL training for dynamic adaptive models, where the BPTT method of training, algorithms used in this thesis, take the learning rate μ itself.

Adaptive Identification and control

The previous was focused on usage of neural units in the sense of adaptive identification of a real engineering system. This subsection will now extend on these neural units as a method of control, with brief review from works [21] & [22] and extension as applied to the hoist deceleration problem, which is indeed the key focus behind this chapter of the thesis.

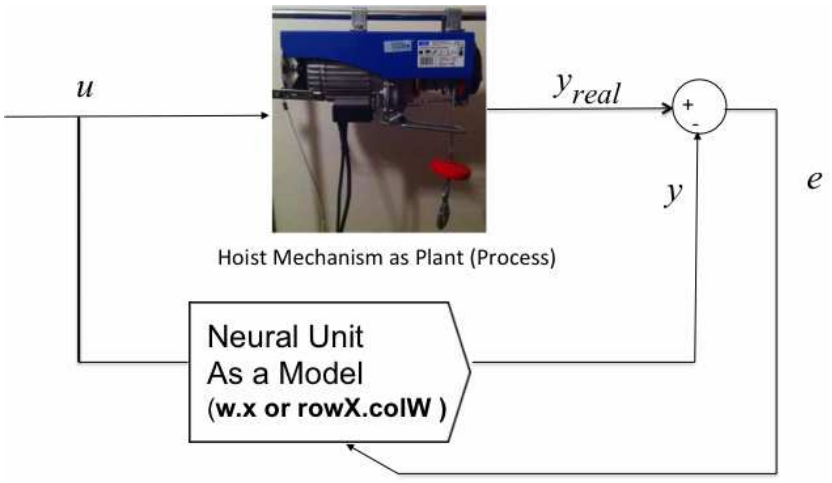


Figure 29: Adaptive Identification with supervised learning of neural networks (**w.x**=LNU, **rowX.colW** =QNU or CNU)

Figure 29, depicts the identification scheme for the previously reviewed neural network architectures. Where, for the scope of this thesis, measured data from the Gude GSZ 200/400 hoist along with a further case study data, will be used to investigate the various neural network approaches for control. *u,* represents the control variable of the hoist (i.e.; frequency, voltage or current supply to the rotor or stator windings of the hoist motor or external braking system). The output y_{real}, corresponds to the hoist deceleration over time, and *y*, being the output of the neural unit, with the difference being the error, *e*.

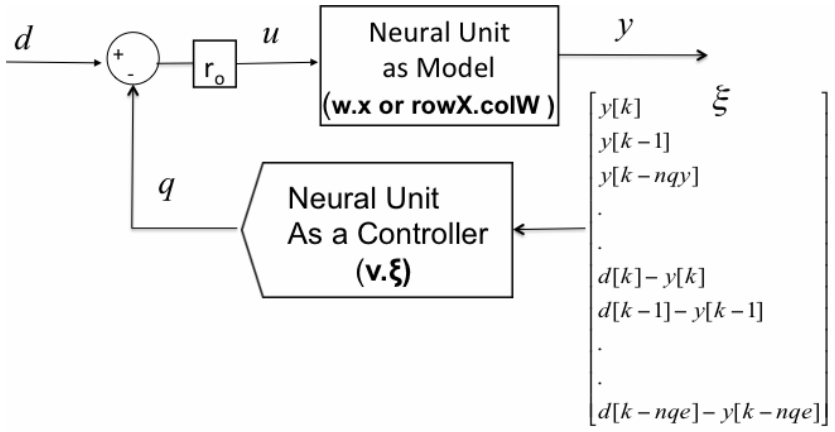


Figure 30: Adaptive control loop for experimental study of a neural network controller (**w.x**=LNU, **rowX.colW**=QNU or CNU) (modification of Figure 2 [22])

Figure 30, shows an extension of the discussed neural architectures for application to the hoist mechanism. Here, once the neural unit as a model is identified, it may be utilised as the foundation for a neural network based control setup. For the scope of this thesis, I will propose an offline tuned control scheme, as the goal here, is to investigate the potentials of applying a neural network based control, for this application. However, the extension to online control is indeed an aim for future work of mine on this problem, beyond this thesis. Figure 30, depicts use of a second neural unit as a controller. Similarly to the previously mentioned architectures, as a controller here too the neural unit may take shape to that of the LNU or HONU adaptive models. However, in this case the adaptive neural weights are tuned differently to that of those used for the neural unit as a model and hence, depicted as *v.* Analogically, should a HONU (as such QNU or CNU) structure be applied, these neural weights would be represented in a matrix form. Further to this, Figure 30 introduced a new vector **ξ**. This vector comprises of a combination of outputs from the neural unit as a model and the difference between the desired behaviour (in our case desired speed or deceleration of the hoist) and the output of the neural model. **v. ξ** or collectively, the variable *q*, thus serves as a manipulator for the newly fed samples of the neural unit as an identified model. Here the GD algorithms are employed in the following manner to achieve sample-by-sample adaptation of the neural weights for the controller, as follows (adopted from eq. 4 [22]);

$$v_{i+1} = v_i + \mu.ereg(k).\frac{\partial y(k)}{\partial v_i} \qquad (27)$$

Where v_i, are adaptable neural weights of the neural unit as a controller and $e_{reg}(k)$ is the error between the desired value of the real system (in this case the hoist, where the desired value will be

denoted as *d*) and the real system output value at sample *k*. $\frac{\partial y(k)}{\partial v_i}$, is the partial derivative of the output of the neural unit as a model, with respect to the individual adaptive neural weights of the neural unit as a controller. An extension of this weight update scheme for BPTT training would result in the following form v=v+Δv, where the change of neural weights for each batch would be analogical to equation (25).

In certain engineering processes, it may also be necessary to adapt not only the neural weights of the neural unit as a controller, but also the gain r_o. For the scope of this thesis, only adaptation of r_o for use on the LNU architectures of the neural unit as a controller, will be investigated. From Figure 30, we may note that the input u(k), to the neural unit as a model is in fact equal to r_o.(d(k)-q(k)). This may more simply be denoted as r_o.e(k), furthermore, ereg(k)=d(k) – y(k), as in the update rule for the neural unit as a controller. With this said, we may setup up the following derivation of the adaptive gain r_o.

First we will establish the update of the r_o over each sample via the gradient descent method, as follows;

$$\Delta r_o = -\frac{1}{2}.\mu r_o.\frac{\partial e_{reg}{}^2(k)}{\partial r_o} \tag{28}$$

$$\Delta r_o = -\frac{1}{2}.\mu r_o.e_{reg}(k).\frac{\partial e_{reg}(k)}{\partial r_o} \tag{29}$$

From Figure 30, ereg(k) = d(k) - y(k) (from vector **ξ**), thus we may proceed with the following substitution;

$$\Delta r_o = -\frac{1}{2}.\mu r_o.e_{reg}(k).\frac{\partial[d(k)-y(k)]}{\partial r_o} \tag{30}$$

The partial derivative of the desired behaviour, with respect to the adaptive r_o, yields to be zero, due to the desired behaviour being independent of the adaptive gain, thus;

$$\Delta r_o = \frac{1}{2}.\mu r_o.e_{reg}(k).\frac{\partial y(k)}{\partial r_o} \tag{31}$$

Furthermore, from the neural unit as a model, we may substitute that y(k) = **w.x**;

$$\Delta r_o = \frac{1}{2}.\mu r_o.e_{reg}(k).\mathbf{w}\frac{\partial x(k)}{\partial r_o} \tag{32}$$

Here, the partial derivative of the input vector of the neural unit as a model $\frac{\partial x(k)}{\partial r_o}$, may be expressed as follows;

$$\Delta r_o = \frac{1}{2}.\mu r_o.e_{reg}(k).\mathbf{w}\left[1 \quad \frac{\partial y(k-1)}{\partial r_o} \quad \frac{\partial y(k-2)}{\partial r_o} \quad .. \quad \frac{\partial u(k-1)}{\partial r_o} \quad \frac{\partial u(k-2)}{\partial r_o} \quad ..\right] (33)$$

As mentioned above, we may substitute u(k) with a difference between the desired behaviour and the output of the neural unit as a controller, including the multiplication with r_o. Thus;

$$\Delta r_o = \frac{1}{2}.\mu r_o.e_{reg}(k).\mathbf{w}\left[1 \quad \frac{\partial y(k-1)}{\partial r_o} \quad \frac{\partial y(k-2)}{\partial r_o} \quad .. \quad r_o.\frac{\partial [d(k-1)-q(k-1)]}{\partial r_o} \quad . \quad .\right] (34)$$

From here, we reach the previous simplification for the partial derivative of the desired behaviour with respect to r_o, being zero, thus;

$$\Delta r_o = \frac{1}{2}.\mu r_o.e_{reg}(k).\mathbf{w}\left[1 \quad \frac{\partial y(k-1)}{\partial r_o} \quad \frac{\partial y(k-2)}{\partial r_o} \quad .. \quad -r_o.\frac{\partial [q(k-1)]}{\partial r_o} \quad . \quad .\right] (35)$$

Therefore, after the above derivation, we yield equation (35). This relation represents the update rule for each respective gain r_o over its incremental adaptation. Similarly to both the neural unit as a model as well as neural unit as a controller the final relation for the newly calculated r_o, over each sample would be in the following form $r_o = r_o + \Delta r_o$.

4.3 –Simulation and Results

After review of all necessary theories behind the various neural architectures for adaptive identification and control, we may now analyse the potentials of these reviewed architectures within the problem of hoist deceleration control. For the scope of this thesis, all simulations will be produced via a software application which I developed in my previous work [22], for investigation of adaptive identification and adaptive controller tuning (this software is open-source from [23]. The version used in this thesis is included with the electronic version of this document).

Throughout this section two cases will be investigated. The first is derived from the angular speed measurements at full loading of the Gude GSZ 200/400 hoist mechanism (ref. Figure 15). From these characteristics we can see that for the initial moments of braking on the tested hoist mechanism, rather large values of deceleration were present, followed by rather low values towards the end of the braking process. For certain industrial applications (e.g; large jib crane applications) such characteristic is not desirable from a functional point of view, where by too large deceleration at the beginning of the braking process, may lead to bobbing or oscillatory behaviour in the hoist ropes during the deceleration, which in turn not only could damage the load, but significantly overstress the mechanism and supporting structure. Therefore, a consideration into control of this deceleration would be necessary. Such control may be achieved via frequency manipulation at the hoist motor input, which is possible in the sense of hoist motors with an electric braking system, via a frequency converter. Therefore, the corresponding output results to speed and furthermore deceleration of the hoist mechanism. Due to the accessibility of data, this section will utilise the previous result of angular speed decay at full load from chapter 3, to analyse the potentials of a neural network approach to this problem. Due to the nature of the Gude GSZ 200/400 hoist mechanism, being at standard a mechanical braking system, the mechanism undergoes braking via application of braking

callipers and not via manipulation of frequency as in the sense of an electrically braked hoist. However, we may suppose an equivalent setup present in such an electrically braked hoist mechanism, where by the first half of samples in our investigated hoist mechanism would be replicated via an electrical brake (as such, via means on a variable frequency drive at the motor input), with corresponding frequencies for the measured speeds of the hoist rotor. Typically in electrically braked hoist applications through the industry (i.e large crane and mining hoist applications), the electric brake operates during the initial stages of the hoist deceleration, once a lower speed is reached, the mechanical brake may take over (as depicted in Figure 31). This is also an advantageous situation due to the minimisation of wear on the mechanical brake lining.

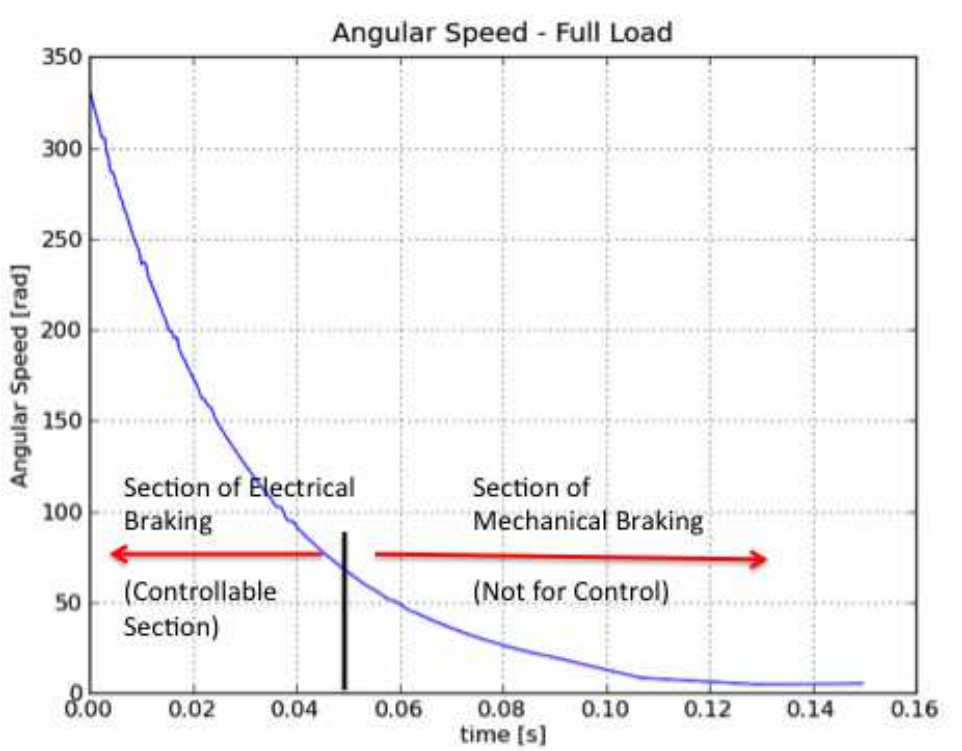


Figure 31: Considered Hoist Data of Deceleration Characteristic (The left side features too high deceleration and would be desired to control via electrical means)

Thus, we will begin investigation of the first case by establishing an equivalent setup using the deceleration characteristic measured on the investigated hoist mechanism (left side of characteristic depicted in Figure 31), with derived input values of frequency, which correspond to the same output as the mechanically braked hoist behaviour that was measured.

$$n = \frac{2.60.f}{no.poles} \tag{36}$$

$$n_{rotor} = n_{stator}.(1 - slip) \tag{37}$$

Equation (36) & (37) above, denote the classical relation for synchronous speed an AC induction motor. From this relation we may note that in order to achieve the same values of speed as measured in the angular speed data of chapter 3, we may use the above conversion, to obtain the various supply frequencies, necessary to achieve the same output behaviour, for the sense of an electrically braked hoist, via variable frequency. Thus, considering that in the investigated hoist

mechanism we have a 2 pole arrangement with a slip value at 0.05, the following table may be derived and further extended for intermediate values;

Variable Frequency [Hz]	Synchronous Speed (ns)[rpm]	Asynchronous Speed (na) of rotor[rpm]
50	3000	2850
40	2400	2280
30	1800	1710
20	1200	1140
10	600	570

Table 2 – Conversion table of speed to corresponding frequency

From Table 2, we may now establish our overall model data for a hoist mechanism undergoing the studied braking characteristic, via electrical braking as follows;

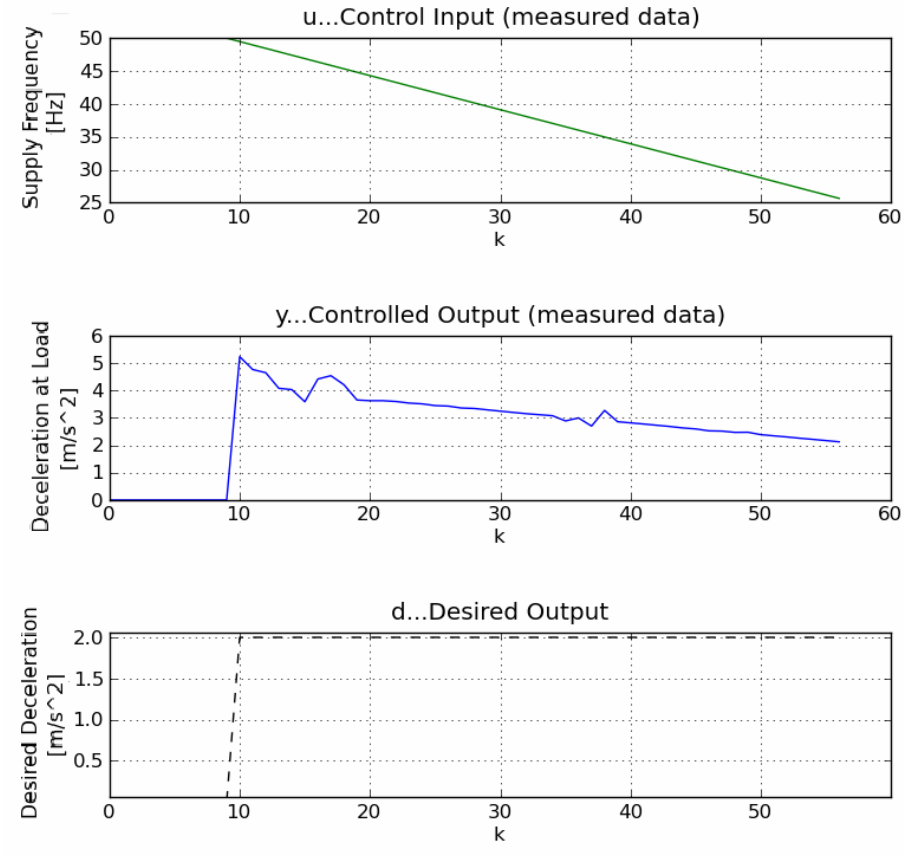


Figure 32: Hoist Model Data for Deceleration at Full Loading

Figure 32, depicts the necessary data for investigation of hoist control via the previously reviewed neural network methods. The data was chosen for the given area of the angular speed characteristic at full load, for which we would desire to control. Thus, the upper plot depicts the corresponding frequencies for the values of speed during the considered deceleration process. The central plot

shows the calculated deceleration from the measured angular speed characteristic from Figure 15, chapter 3, with the final plot depicting the desired output values for the deceleration.

Due to the dynamics of the hoist motor under loading, the central plot in Figure 32, features several samples of non-steady behaviour in the hoist deceleration. With the small scale imperfections being a likely cause of inaccuracies of measurement, the larger changes in the deceleration, especially at the beginning of the deceleration process may be due to several key phenomenons. The first is that during the initial moments of deceleration as the hoist pendant is pressed off by the operator, the de-energising of the hoist stator current retains a certain impedance, which results in a minor counter current in the rotor windings, which may result in a discrepancy in the braking process. Another consideration is in the sense of the mechanically braked hoist mechanism. From the initial moment the hoist motor is de-energised, a vibration is induced due to the sudden engagement of the braking lining with its retraction spring. And in the sense of electrically braked hoists, a substantial change of the input frequencies during the deceleration process, can result in a similar non-steady behaviour.

Figure 32, depicts the desired behaviour of the hoist mechanism for this given deceleration characteristic. This value is usually dictated from standards, in this case crane hoist standards [1] , suggests that in order to maintain a safe level of dynamical hoisting factor of 120% of the load, we should have a deceleration being $1/5^{th}$ of g. which corresponds to $2m/s^2$ maximum deceleration on the load. With all data established, we may proceed with analysing the adaptive identification process of the dynamical hoist system as follows;

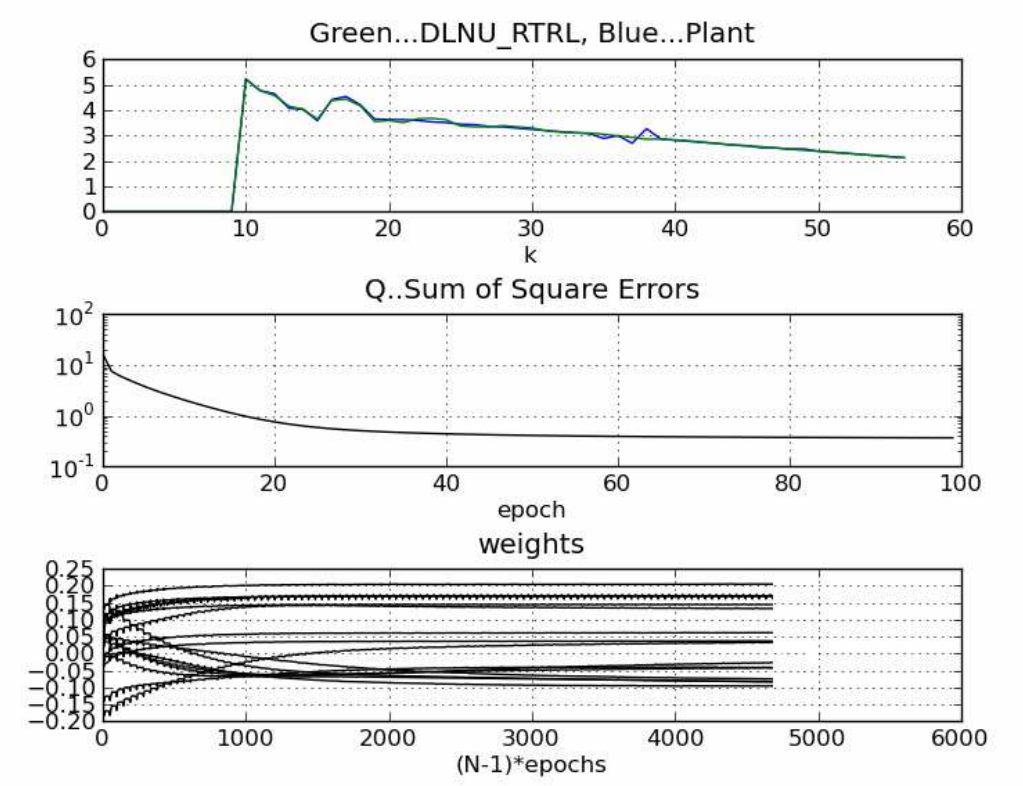


Figure 33: Adaptive Identification of Hoist Mechanism with DLNU and RTRL Training, ny=9 (No of previous samples of neural model output), nu=5(No. of previous samples of real process input), mu (learning rate)=0.2, epochs (runs of algorithm)=100

Figure 33, depicts the adaptive identification of the investigated hoist data, with the DLNU with RTRL training. From this figure we may note that the adaptive model (green line) follows the hoist real data almost exactly. The central plot depicts the decreasing sum of square errors throughout the learning process, its steady decay implies stable learning however this is more so evident in the weights, plotted on the lower graph. This graph depicts the learning of each individual adaptive weight in the neural model, here we may note the gradual increase of each weight for the linear neural model until its stable value is shown, this stable value corresponds to the trained coefficients of the polynomial model as seen in equation (21), represented by vector **w**. We may now investigate the behaviour with a higher order neural unit (HONU) with BPTT training.

Figure 34 depicts the adaptive identification of the hoist data with DQNU trained via the BPTT learning algorithm. What we may notice is that the identification is even more precise to the real behaviour of the hoist data and with faster learning, achieved over just a few epochs or runs of the learning algorithm. As mentioned in the reviewed theory of this thesis, the BPTT algorithm is advantageous in learning data affected by noise, as such here where a minor disturbance is present, the model focuses on the main governing law as opposed to learning this behaviour.

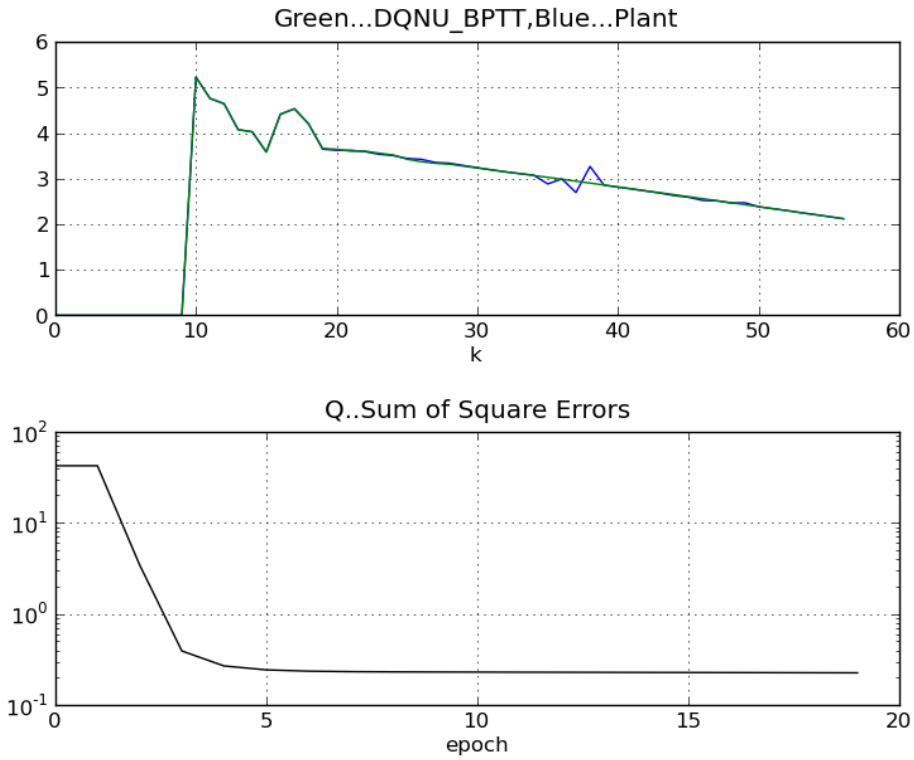


Figure 34: Adaptive Identification of Hoist Mechanism with DQNU and BPTT Training, ny=9 (No. of previous samples of neural model output), nu=5 (No. of previous samples of real process input), mu (learning rate)=0.2, epochs (runs of algorithm)=20

Due to the successful dynamical modelling of the hoist system behaviour, we may now investigate the behaviour of a Neuro-controller, as applied to the identified neural model in Figure 34, as follows;

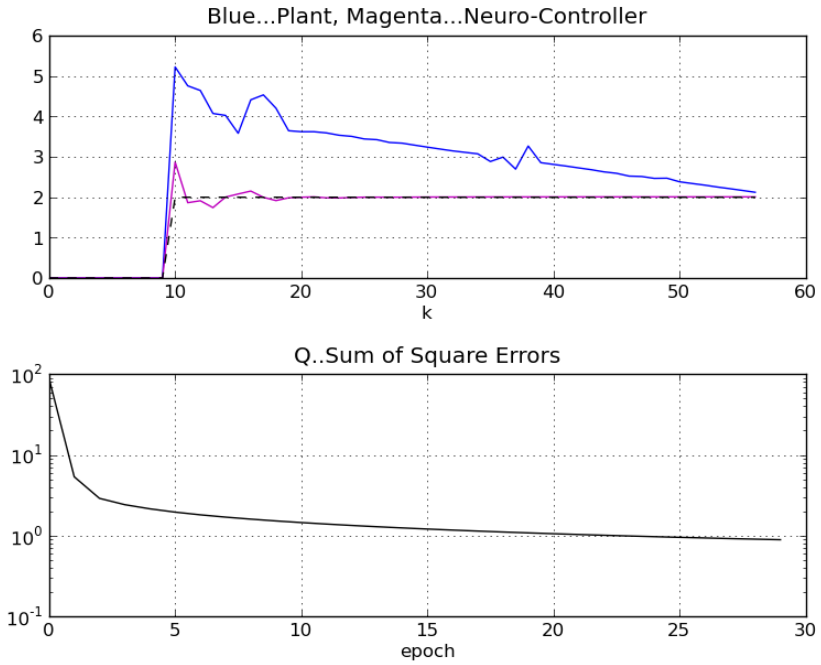


Figure 35: Adaptive Control of Hoist Mechanism with QNU BPTT Training, nqy=9 (No of previous samples of neural model output), nqe=0 (No. of previous samples of difference between desired and output of the neural model), mu (learning rate)=1, epochs (runs of algorithm)=30 (Trained with DQNU BPTT as per Figure 34)

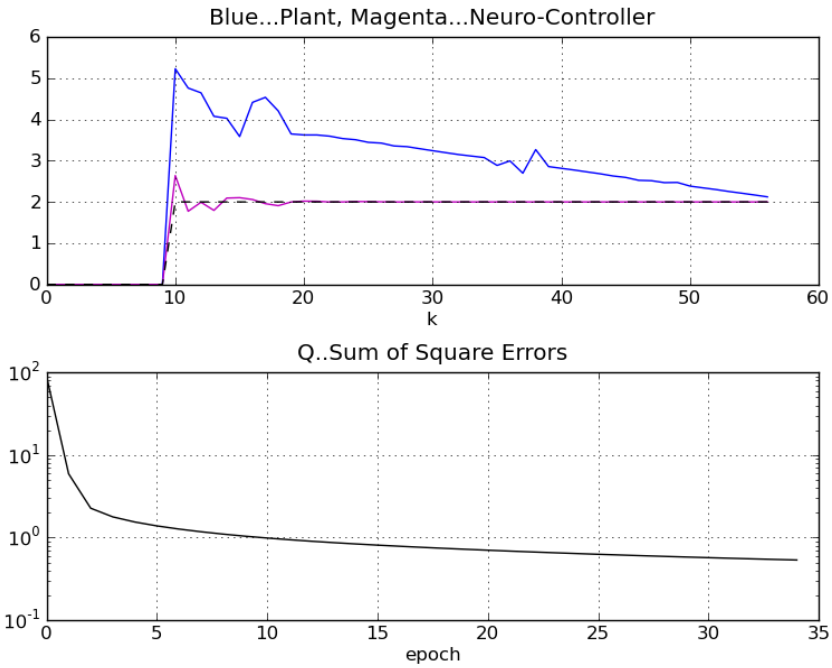


Figure 36: Adaptive Control of Hoist Mechanism with CNU BPTT Training, nqy=9 (No. of previous samples of neural model output), nqe=0 (No. of previous samples of difference between desired and output of the neural model), mu (learning rate)=1, epochs (runs of algorithm)=35 (Trained with DQNU BPTT as per Figure 34)

Following training via the DQNU with BPTT as depicted in Figure 34, the extension of a Neuro-controller in state feedback was applied to the hoist data, with the black line depicting the desired behaviour, the blue being the real system and the magenta line being the Neuro-controller output. Figure 35 & Figure 36, thus depict the application of the QNU and CNU Neuro-controller with BPTT training methods, respectfully. As we may notice, the QNU controller achieved adequate control, following closely to the desired behaviour of the hoist deceleration, within the first several samples of application. However, we may notice from Figure 36 that in the initial moment of the braking effect, corresponding to the largest deceleration value. The CNU Neuro-controller achieves more desirable behaviour peaking at 2.6 m/s^2 as opposed to the QNU controller with 2.9m/s^2, whilst still adhering closely to the desired behaviour and with slightly faster convergence as seen in the respective sum of square error graphs. Thus, from these results, the CNU shows to provide slightly better performance for this given hoist deceleration data. Furthermore, from these produced results we may see the potentials that the investigated neural architectures have in controlling the hoist deceleration to achieve close response to the desired deceleration value. Another key consideration is that for the above tested case, a large variation is present from the real deceleration of the hoist and the desired behaviour. In spite this, the Neuro-controller is still able to quickly respond to the desired behaviour of the system within the first few samples, and maintain a steady desirable output value even when a change in the dynamics of the system, or large disturbance may be occur.

Following this study, we may see the potentials that the various neural architectures have in control of the hoist deceleration during the hoist braking process, with desirable control achieved on the hoist model data from Figure 31. Following this we will now investigate the second case study for the scope of this section, which is that of a mining hoist system (the data of which was adopted from [20]). The scale of such mining hoist is significantly larger than the previously investigated hoist mechanism and even so to average sized crane hoist applications. However, as reviewed in the works of [18] & [19], hoist deceleration remains as an area of study within the field due to the complexities in the dynamics of the hoist mechanism during the deceleration process. In this particular case study a DC hoist motor is used as the main hoist drive. In the sense of mining hoists, various methods of braking are possible, being via mechanical, electrical and hydraulical means, as well as a combination in certain cases of application. However, for the scope of this section being focused on single-input-single-output form of control, this investigation will be more case specific to hoist mechanisms with a single control variable in the sense of an electrical control signal. This may be achieved in hoist mechanisms via two key approaches. The first is via control of frequency by means of a variable frequency controller in the sense of the AC driven hoist motors. The other form, as will be shown from this case study is braking via a dynamical braking system. In such case variation of an external dynamical resistor, or stator field current of the dynamical brake may be controlled in order manipulate the braking process of the hoist mechanism and consequently its speed and furthermore, deceleration.

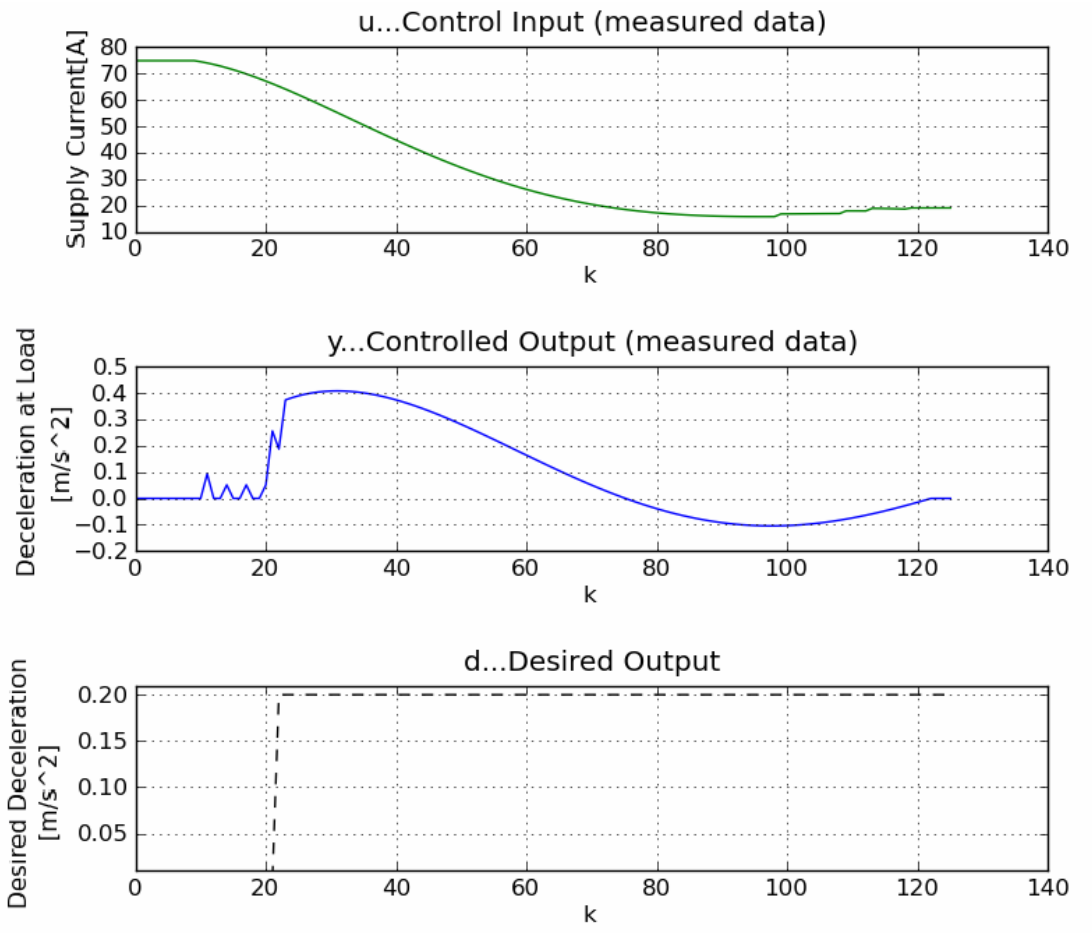


Figure 37: Mine Hoist Deceleration Characteristic (adopted from [20])

Figure 37, depicts the case study data for the mine hoist application. Here the upper plot shows the input value for the hoist mechanism, which is that of the field current applied to, in this case a dynamical braking system at the hoist motor shaft. The deceleration characteristic is depicted in the central plot. In this case the overall deceleration characteristic lasted 11 seconds from a speed of 1.7 m/s to 0.3m/s at the load, where, from here, the mine hoist motor may then undergo further braking via mechanical means. Either via a rope brake system or disc or drum braking mechanism. In this case due to mining safety regulations, the desired hoist deceleration is 0.20 m/s^2. Due to the dynamics of the mine hoist mechanism, the several initial samples corresponding to the application of braking, results in relatively little effect on the hoist deceleration, with minor fluctuation due to the nature of the hoist mechanism. In sense of adaptive identification, the primary goal is to identify the overall characteristic as closely as possible, whilst still retaining the general profile of the characteristic. It was found that of the tested neural network architectures for adaptive identification. The DQNU with BPTT training featured the best performance, with re-sampling of every second sample where, ny=9, nu=5, mu=0.35, epochs=30 (ref. Figure 38). Following the adaptive identification, an extension of the QNU Neuro-controller with BPTT training featured the best performance (Figure 39). Here a simpler structured model as opposed to the adaptive identification neural model was sufficient, with only two samples back of the regulator error required to provide desirable control, as depicted in the magenta line.

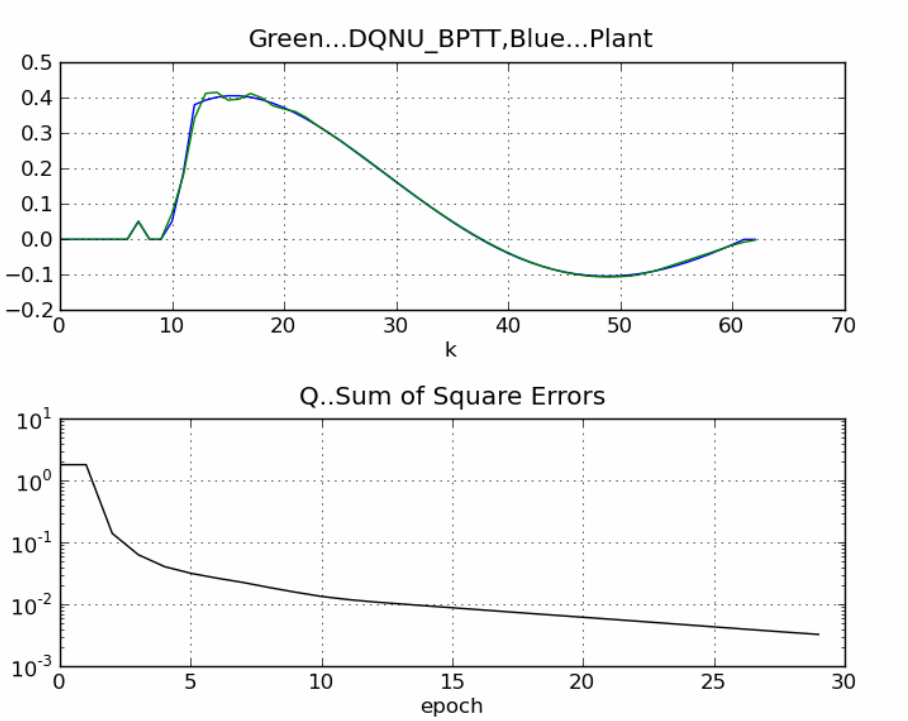


Figure 38: Adaptive Identification of Mine Hoist Mechanism with DQNU BPTT Training, ny=9 (No. of previous samples of neural model output), nu=5 (No. of previous samples of real process input), mu (learning rate)=0.35, epochs (runs of algorithm)=30, re-sampling=2

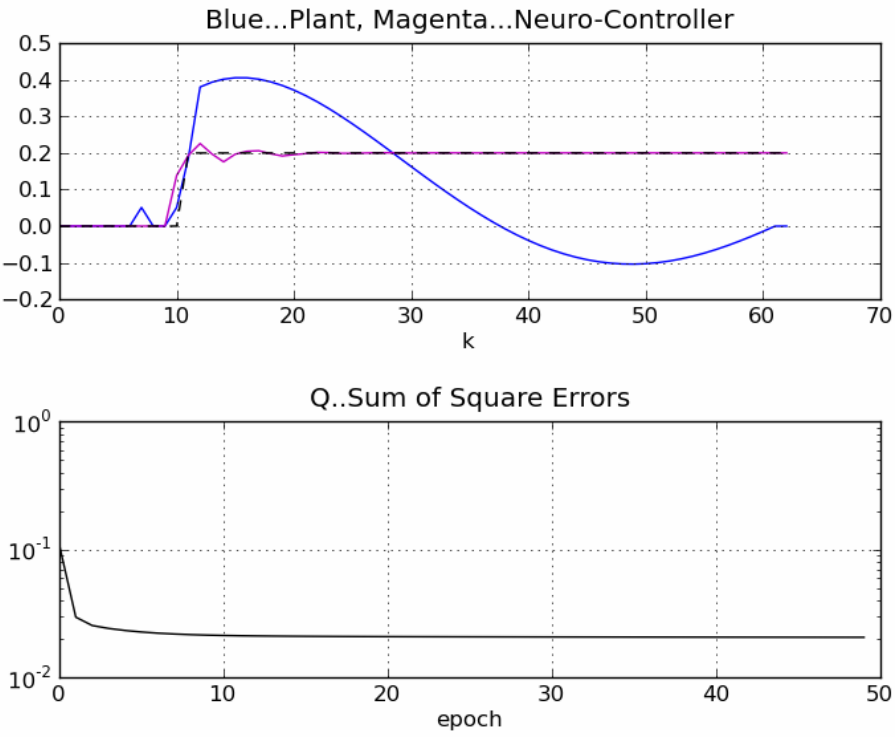


Figure 39: Adaptive Control of Mine Hoist Mechanism with QNU BPTT Training, nqy=9 (No. of previous samples of neural model output), nqe=2(No. of previous samples of difference between desired and output of the neural model), mu (learning rate)=0.03, epochs (runs of algorithm)=50, re-sampling = 2, (Trained with DQNU BPTT as per Figure 38)

4.4 –Braking Torque Analysis Following Control

Following the above investigation of a neural network approach to hoist deceleration control, this section aims to analyse the corresponding braking torque in both static and dynamical cases, for the simulated control output of the first case in the previous section (Figure 36). It is interesting to analyse the behaviour of the hoist mechanism in terms of its braking torque characteristic once the controller has been applied to the hoist mechanism data, and compare it with the previously calculated with the absence of control. As the Gude GSZ 200/400 was used as the basis of the controlled model, all inertias of the hoist mechanism are analogical to those calculated in chapter 3 of this thesis. Keeping in mind that the produced data was measured for full loading, we may recalculate the dynamic and static values of braking torque for the controlled deceleration output via the new braking torque application as demonstrated in chapter 3, as follows;

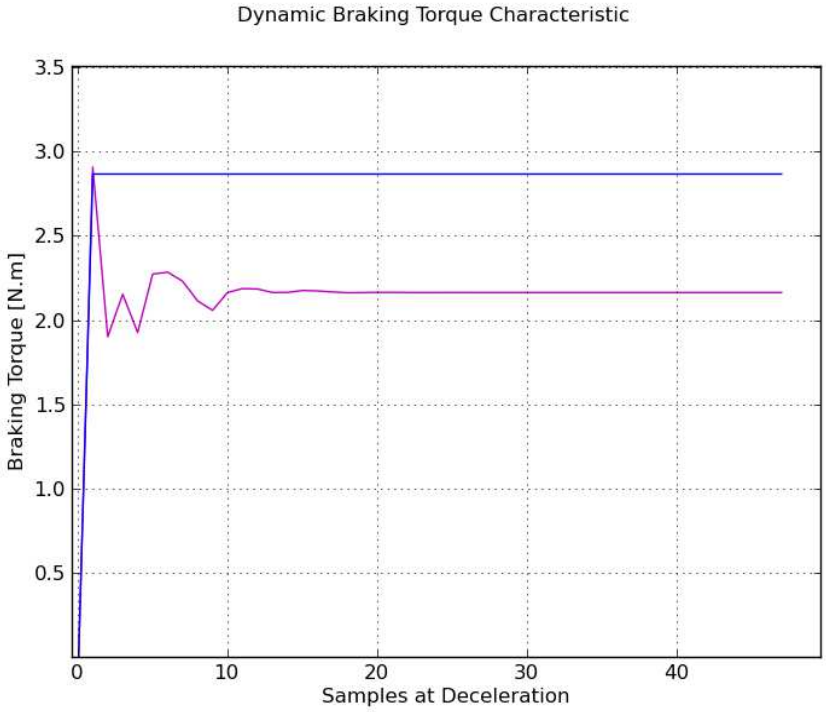


Figure 40: Dynamic Braking Torque Analysis - Adaptively Tuned Neuro-Controller with Hoist Mechanism data (magenta), and Original Hoist Mechanism Data (blue)

Figure 40, depicts the calculated dynamic braking torque at the zone of braking, for the investigated hoist mechanism before and after application of the Neuro-controller. Here we see a reflection in the values of the braking torque characteristic with respect to the controlled deceleration values. During the initial stages of control, the Neuro-controller featured a few samples of deviation from the desired behaviour of the hoist mechanism and correspondingly the braking torque values too, represent this component of the behaviour. However, in spite this, the calculated values of dynamical braking torque are far more desirable, considering the average deceleration produced by the Neuro-controller, bearing in mind the blue characteristic in Figure 40 represents the braking torque due to an averaged deceleration of the main braking zone portion of the measured hoist mechanism data, for the purpose of concluding an overall value of braking torque for the hoist mechanism. Here too may be concluded that the overall value of braking torque for the controlled

hoist mechanism data, is 2.2 N.m, which is a desirable value. Thus the deviation from the minimum prescribed value is 1.23 N.m, which is more desirable, than being too many folds higher, corresponding to too high deceleration, during the main zone of the braking effect of the hoist mechanism. However, this result is also motivation for testing on a real industrial hoist mechanism, where a look onto even further optimisation may also be considered, should the performance on the real system with control indeed be the same of that presented in these results.

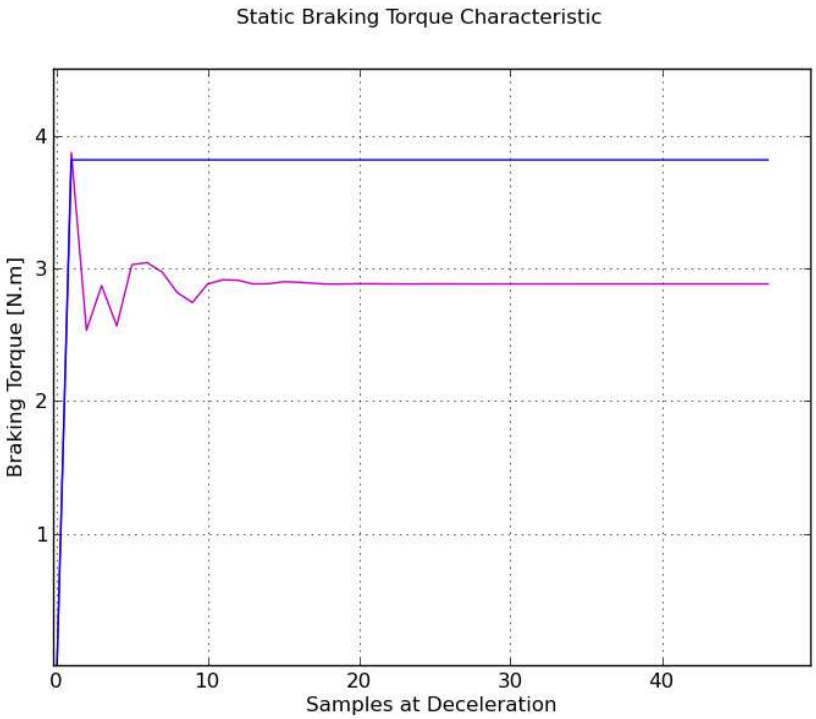


Figure 41: Static Braking Torque Analysis - Adaptively Tuned Neuro-Controller with Hoist Mechanism data (magenta), and Original Hoist Mechanism Data (blue)

Figure 41, depicts the static braking torque characteristic at the zone of braking, before and after the Neuro-controller application on the hoist mechanism data. Here we may note a similar characteristic to that of the dynamic braking torque characteristic, with the static braking torque also reflecting the deceleration characteristic of the controlled hoist mechanism. Here the static torque in its settled stage, deviates from the permissible braking torque by 1.592 N.m, which here too is more desirable compared with the previously calculated values as depicted in blue (Figure 41), showing that the hoist mechanism in its braking phase, is not subject to too high values of deceleration and correspondingly too large braking torque values.

From these results as depicted in Figure 40 & Figure 41, we may see the direct effect the hoist deceleration has on the real braking torque of the hoist mechanism, in both static and dynamical cases. This thus justifies the impact the hoist deceleration has on the hoist mechanism and supporting structure during its braking process, as seen from this analysis, too large values of braking torque and also too low values, corresponding to too slow rate of deceleration due the braking phase can lead to overstressing of the hoist mechanism and structure, furthermore incompliance with the permissible limits prescribed by standards and thus, ultimately being a risk of safety.

5. Discussion

After completion of the main objectives of this thesis, a summarisation of the produced results behind this work along with its key contributions may be drawn. As stated in the introduction section, two parallel objectives were set, the first was focused on braking torque analysis of an industrial hoist mechanism, with the output of this investigation being the development and testing of a new application that can effectively calculate and diagnose whether the investigated hoist mechanism is in compliance with the required permissible limits of braking torque or not. In addition to this, was also the goal to implement a simple, economical hardware setup for measuring the deceleration characteristic and temperature of the gearbox casing as needed for braking torque analysis for a general sense of industrial hoist mechanisms.

As shown in chapter 3 of this thesis, the application which I have designed and developed for braking torque measurement and calculation, was successfully implemented. Testing of this application with the various braking torque characteristic calculations and furthermore measurement and visualisation functions, proved to be a key tool in the analysis of the real braking torque of the investigated hoist mechanism. And furthermore, shows its usefulness and potentials for application on other industrial hoist mechanisms, as an efficient means of investigating the real braking torque of a hoist mechanism. Looking at the produced results as seen in chapter 3 of this thesis, the measured values from the angular distance and speed characteristics were overall depicted quite well. The functionality of the implemented hardware was also investigated via comparison with the Labjack DAQ module, to compare readings in respect of the angular distance measurement over time. From the produced results of this comparison, the implemented hardware seems to be in agreement with the results produced by the Labjack DAQ module, however minor inaccuracies were present in the implemented hardware, which corresponded to several samples of the respective angular speed characteristics being affected. In spite this, a simple easy to use hardware setup was successfully implemented which could be applicable on small to mid range hoist mechanisms with the extension of a coded wheel or coded ring configuration for measurement. Further to this, the overall characteristics for the respective loadings were captured quite well, with the produced results seemly in accordance with what may be theorised from mathematical and physical considerations, concerning the deceleration characteristic of the hoist mechanism. An interesting phenomenon seen from these results is the rather larger rate of deceleration produced by the investigated hoist mechanism during its main zone of braking effect. However, due to the investigated hoist being a small application low capacity hoist. The overall design is achieved to be as economical as possible and thus featuring only a two stage reduction gearbox with a standardised small hoist motor. Hence, large values of deceleration are not of such a concern due to smaller loads and low duty application being considered. However, testing on the Gude GSZ 200/400, allowed for testing and representation of all the same principals and considerations necessary for analysis of real braking torque on larger industrial hoist mechanisms. With the overall conclusion that in investigation of this particular hoist design, though adequate values of braking torque were measured in the sense of both static and dynamic braking torque. The Gude GSZ 200/400 tends to feature too abrupt deceleration, which for other engineering applications could be quite undesirable. This was thus also a further motivation for investigation of hoist deceleration control, as was the second key objective behind this thesis.

The second key goal of this thesis, was to study to problem of hoist deceleration control. From the produced results of chapter 3, we can note that the relatively abrupt deceleration produced by the hoist mechanism, particularly at smaller loads would be undesirable in certain industrial application. Though the investigated hoist was of rather smaller scale, similar concerns in the undesirable behaviour of hoist deceleration may be seen throughout larger industrial application, as also recalled from the review works in chapter 4 of this thesis. Where, too large values or too high fluctuations in the deceleration value may in turn lead to damaging of the load, or overstressing of the hoist mechanism and structure, which also leads to a direct effect on safe operation and safety of personnel. Hence, chapter 4 of the thesis was thus aimed at a study into the potentials of hoist deceleration via a neural network based approach. This followed from the promising, previous studies on neural network based dynamical system identification and control, as already implemented throughout various other single-input-single-output natured engineering systems. After review of the necessary theories of adaptive identification and control, an extension to the problem of hoist deceleration control was realised.

This investigation of hoist deceleration control featured two case studies, the first was based on the measured results from the Gude GSZ 200/400 mechanism at full loading. Due to the nature of this hoist being only mechanically braked, a consideration was extended to modelling the equivalent output behaviour via an electrically braked means of the hoist mechanism (with extension of a single-phase frequency inverter in the sense of the Gude GSZ 200/400) featuring the same deceleration characteristic. The results of this first case showed that although quite large deviation in the output data of the hoist mechanism with respect to its desired behaviour was shown, the adaptive identification via a neural network based model particularly the DQNU with BPTT, proved to model the behaviour of the hoist deceleration characteristic data quite well. Following this an extension of the Neuro-controller, via QNU & CNU with BPTT training, quite desirably followed the behaviour of the desired hoist mechanism output during simulation. As seen in Figure 35 & Figure 36, though adequate control was achieved, the initial values at the beginning point of braking had minor fluctuations from the desired value, it is thus an important future direction to test behaviour of such proposed neural network setup (particularly the CNU with BPTT training as a Neuro-controller) on the real hoist mechanism, as may be recalled in work [21], the results of the Neuro-controller after real application indeed proved to be of better performance as compared with its simulation. Following experimentation of such control on the real system, one may also identify any potentials for further optimisation, should the results produced be similar to that presented in Figure 35 & Figure 36. Another consideration is the overall deceleration time, which in this case study was over 0.15 seconds, which is quite rapid. This is however, due to the small scale of the Gude GSZ 200/400 with relatively small stopping distance of the hoist load being present. Thus, the capability of actuation for control in such quick cases of deceleration timing should be checked with the real application, if such control can be implemented for the specific real application, however in spite of this, the produced results are quite promising. Should such a hoist mechanism setup already feature an initial means of control (i.e; conventional PID or fuzzy control), the proposed neural network control also features potentials for further optimisation and thus, arises even further motivation for research on other hoist mechanisms in the field of this study.

The second following case study analysed a mine hoist case study, in the sense of an example of a real case, large scale industrial hoist mechanism. In this study the QNU with BPTT training in the sense of both adaptive identification and control, deemed to be the best performing architecture for the sense of this specific application. Though, this case study was quite specific to a dynamically

braked mine hoist mechanism, the principal structure for the control scheme is analogical to that studied in the first case and even for application to AC induction hoist motors with variable frequency drives serving as the control variable. The results of this study showed that the application of the Neuro-controller to this case study delivered desirable behaviour in terms of deceleration of the hoist mechanism and is thus motivating for further research into application for the scope of the mining hoist application and also crane based applications where hoist deceleration control may be necessary either initially or may need further optimisation for even further enhanced behaviour of the hoist mechanism.

As a final component to this thesis, a relation between the braking torque and hoist deceleration was drawn. Using the newly developed braking torque application, this study shows the effect the deceleration has in the overall values of the hoist braking torque, for both static and dynamic cases. After calculation of the braking torque for the controlled hoist deceleration as simulated in the first case of the hoist deceleration control subsection (ref. Figure 36), the hoist mechanism correspondingly showed desirable values of the braking torque in both static and dynamic values, being in reasonable deviation from the prescribed minimum and thus confirming that the deceleration value after control was not too high from a braking torque point of view.

6. Conclusions and Future Directions

After completion of the key objectives within the scope of this thesis, we may conclude the following. The first is that the hardware setup implemented in chapter 3 of this thesis was aimed to be an economical solution to measurement of the hoist deceleration. With the presented setup featuring adequate functionality for small to medium range hoists where a coded wheel or coded ring could be implemented on the end of the hoist rotor shaft. However, a future consideration may also be to implement such setup in the form of an external device which can be extended to the hoist rotor shaft without any additional design of a joining coupling for connection. Further to this, the implemented time measurement from the hardware setup, did show to have minor discrepancies. Though still adequate, an even more accurate method of the time measurement can be considered, due to the rather rapid timing between the initial intervals of the applied coded wheel, being within the order of microseconds measurement. In spite of this, a functional hardware setup was presented with the produced results being adequate for accurate analysis of the real braking torque on the investigated hoist mechanism.

Another key point is of the new unique application which was presented in chapter 3 of this thesis. From the produced results of the braking torque analysis, we may conclude that this newly developed application is an efficient software tool for braking torque analysis and diagnosis. And with the correct values of the hoist mechanism data being input to the application edits, the calculation of the braking torque can be realised instantly following measurement of the hoist mechanism deceleration and gearbox temperature. A future consideration would be to test this application on a larger scale industrial hoist mechanism and possibly extending further functions to the program. However, this tool still provides a good platform for engineers and trained users to investigate adequately the true braking torque of the hoist mechanism, which ultimately ensures the safety of use of the particular hoist mechanism at investigation.

The final point of conclusion is that of the hoist deceleration control, presented in chapter 4. From this new study, we may see the potentials that especially the QNU & CNU with BPTT training has in providing desirable behaving control to the hoist mechanism data. In terms of the adaptive identification, the BPTT training method was able to identify the process data with better performance, particularly with the noised regions within the data being ignored and the model focusing only on the main governing law of the hoist mechanism output data. Following this QNU with BPTT identification, it was shown that within a relatively short amount of runs of the Neuro-controller algorithm, the Neuro-controller was tuned to provide desirable control of the hoist mechanism, following closely to the desired behaviour almost exactly with exception of the first several samples at the beginning of the controller's application. We may also note the excellent speed of convergence of the Neuro-controller, as seen in Figure 35 & Figure 36 where the QNU & CNU with BPTT achieved fast convergence in its sum of square errors, even more so in the sense of the CNU Neuro-controller. A future goal following these results is testing of this proposed neural network setup for real time control of an industrial hoist mechanism. As may be recalled in the work [21], the real application of the tuned Neuro-controller provided better performance than that of its simulation, following proper setup of the adaptive control algorithm. With real application on an industrial hoist mechanism, further optimisation on the real system may also be necessary and thus also leading to a further point in research.

7. References

[1] Australia Standards Association, AS 1418.1-2002 – Cranes, hoists and winches Part 1 –General requirements, Fourth Edition, Sydney, Australia ISBN 0 7337 4372 2, 2002

[2] Australia Standards Association, AS 1418.14-1996 –Requirements for cranes subject to arduous working conditions, Fourth Edition, Sydney, Australia, 1996

[3] Australia Standards Association, AS 1403-2004 –Design of rotating steel shafts, Sydney, Australia, 2004

[4] Mannesmann Demag -Komponenten Handbuch (Demag components handbook), Printed in Germany, 1986.

[5] Electric Motors Catalogue –SEW Eurodrive URL: http://www.sew eurodrive.com/download/pdf/16936426.pdf, Visited 20/11/13

[6] Gude GSZ 200/400 Hoist Manual URL: http://www.gude.cz/upload/File/el-lanove-navijaky.pdf, Visited 22/9/13

[7] Cranes Design, Practise and Maintenance, URL http://www.scribd.com/doc/15587793/Cranes-Design-Practice-And-Maintenance-Malestrom, Visited 28/9/2013

[8] Electronic Load Controlled Rail Mounted Quay Crane – Haldia Dockyard URL: http://www05.abb.com/global/scot/scot293.nsf/veritydisplay/885dd14854cdd4c2c12576a50048c311/$file/elc%20sts%20haldia%20dockyard%20flyer%203ast003458d0087%20rev-.pdf, Visited 29/9/13

[9] Programming Language Python URL: http://www.python.org/, Visited 5/10/13

[10] Numpy Documentation Page URL: http://docs.scipy.org/doc/, Visited 5/10/13

[11] wxPython Documentation Page URL: http://www.wxpython.org/, Visited 5/10/13

[12] Matplotlib Documentation Page URL: http://matplotlib.org/, Visited 5/10/13

[13]Vishay TCST2103 Optical Reflective Sensor Documentation URL: http://www.vishay.com/docs/81147/tcst2103.pdf, Visited 22/9/13

[14]Vishay CNY70 Optical Reflective Sensor Documentation URL: http://www.vishay.com/docs/83751/cny70.pdf, Visited 22/9/13

[15] PIC 16F628A Microcontroller Data Sheet URL: http://ww1.microchip.com/downloads/en/DeviceDoc/40044G.pdf, Visited 22/9/13

[16] Dallas DS18B120 Temperature Sensor Dataheet URL: http://dlnmh9ip6v2uc.cloudfront.net/datasheets/Sensors/Temp/DS18B20.pdf, Visited 22/9/13

[17]Electrical Wire Rope Hoist URL: http://technical-portuguese.blogspot.cz/2008_08_01_archive.html, Visited 28/9/13

[18]Liu Jingyan, Wang Fuzhong, "The Fuzzy Neural Network Control of Hoist Constant Deceleration Braking System Based on Genetic Algorithm," icdma, vol. 2, pp.687-689, 2010 International Conference on Digital Manufacturing & Automation, 2010

[19]Wu Yanxiang, Tian Zhanfei, Xue Minjie, "Study and Simulation on Variable Frequency Speed-Regulating System of Mine Hoist Based on PLC Fuzzy Control," CISP 2010, 3rd International Congress on Image and Signal Processing, 2010

[20] Mine Hoist Dynamic Braking System Study URL: http://www.msha.gov/S&HINFO/TECHRPT/HOIST/PAPER6.HTM, Visited 22/11/2013

[21]Ladislav Smetana: *Nonlinear Neuro-Controller for Automatic Control Laboratory System*, Master's Thesis (sup. Ivo Bukovsky), Czech Tech. Univ. in Prague, 2008

[22]Peter M. Benes (2013): *Software Application for Adaptive Identification and Controller Tuning* Student's Conference STC, Faculty of Mechanical Engineering, CTU in Prague. 2013 (Awarded paper: STČ + Bosch + STOČ Zlín)

[23]Adaptive Identification and Control Software Application URL: http://users.fs.cvut.cz/ivo.bukovsky/Research_and_Applications/projects/SGS12177OHK23T12/index.htm, Visited 16/11/13.

[24]Gupta, M., M., Bukovsky, I., Homma, N., Solo M. G. A., Hou Z.-G. (2012): "*Fundamentals of Higher Order Neural Networks for Modeling and Simulation*", in Artificial Higher Order Neural Networks for Modeling and Simulation, ed. M. Zhang, IGI Global,.

[25] Bukovsky, I., Bila, J., Gupta, M., M, Hou, Z-G., Homma, N. (2010): "*Foundation and Classification of Nonconventional Neural Units and Paradigm of Nonsynaptic Neural Interaction*" in Discoveries and Breakthroughs in Cognitive Informatics and Natural Intelligence within the series of the Advances in Cognitive Informatics and Natural Intelligence (ACINI), ed. Y. Wang, IGI Publishing, Hershey PA, USA, ISBN: 978-1-60566-902-1, pp.508-523.

[26] Bukovsky, I., Homma, N., Smetana, L., Rodriguez, R., Mironovova M., Vrana S. (2010): "*Quadratic Neural Unit is a Good Compromise between Linear Models and Neural Networks for Industrial Applications*", ICCI 2010 The 9th IEEE International Conference on Cognitive Informatics, Tsinghua University, Beijing, China, July 7-9.

[27] Bukovsky I., S. Redlapalli and M. M. Gupta (2003): *Quadratic and Cubic Neural Units for Identification and Fast State Feedback Control of Unknown Non-Linear Dynamic Systems*, Fourth International Symposium on Uncertainty Modeling and Analysis ISUMA 2003, IEEE Computer Society, , Maryland USA, ISBN 0-7695-1997-0, p.p.330-334

[28] Rodriguez , R., Bukovsky, I., Homma, N.: "*Potentials of Quadratic Neural Unit for Applications*", in International Journal of Software Science and Computational Intelligence (IJSSCI) ,vol 3, issue 3, IGI Global, Publishing, Hershey PA, USA ISSN.

[29]Bukovsky, I., Lepold, M., Bila J.: "*Quadratic Neural Unit and its Network in Validation of Process Data of Steam Turbine Loop and Energetic Boiler*",WCCI 2010, IEEE Int. Joint. Conf. on Neural Networks IJCNN, Barcelona,Spain, 2010.

[30]P. J.Werbos (Oct. 1990) "Backpropagation through time: What it is and how to do it" Proc. IEEE, vol. 78, no. 10, pp. 1550–1560.

Appendix

Basic Characteristics of TCST2103 Optical Reflective Sensor

BASIC CHARACTERISTICS (1)							
PARAMETER	TEST CONDITION	PART	SYMBOL	MIN.	TYP.	MAX.	UNIT
INPUT (EMITTER)							
Forward voltage	I_F = 60 mA		V_F		1.25	1.6	V
Junction capacitance	V_R = 0 V, f = 1 MHz		C_j		50		pF
OUTPUT (DETECTOR)							
Collector emitter voltage	I_C = 1 mA		V_{CEO}	70			V
Emitter collector voltage	I_E = 10 µA		V_{ECO}	7			V
Collector dark current	V_{CE} = 25 V, I_F = 0 A, E = 0 lx		I_{CEO}			100	nA
SWITCHING CHARACTERISTICS							
Turn-on time	I_C = 2 mA, V_S = 5 V, R_L = 100 Ω (see figure 2)		t_{on}		10		µs
Turn-off time	I_C = 2 mA, V_S = 5 V, R_L = 100 Ω (see figure 2)		t_{off}		8		µs

Note

(1) T_{amb} = 25 °C, unless otherwise specified

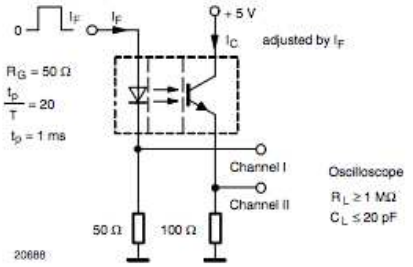


Fig. 2 - Test Circuit for t_{on} and t_{off}

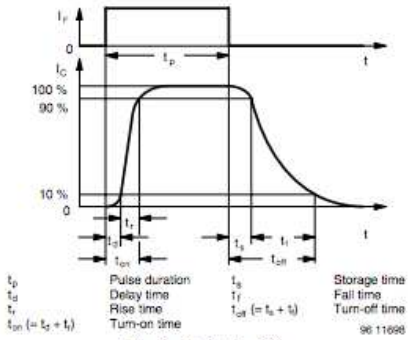


Fig. 3 - Switching Times

Basic Characteristics of CNY70 Optical Reflective Sensor

BASIC CHARACTERISTICS (T_{amb} = 25 °C, unless otherwise specified)						
PARAMETER	TEST CONDITION	SYMBOL	MIN.	TYP.	MAX.	UNIT
COUPLER						
Collector current	V_{CE} = 5 V, I_F = 20 mA, d = 0.3 mm (figure 1)	I_C (2)	0.3	1.0		mA
Cross talk current	V_{CE} = 5 V, I_F = 20 mA, (figure 2)	I_{CX} (3)			600	nA
Collector emitter saturation voltage	I_F = 20 mA, I_C = 0.1 mA, d = 0.3 mm (figure 1)	V_{CEsat} (2)			0.3	V
INPUT (EMITTER)						
Forward voltage	I_F = 50 mA	V_F		1.25	1.6	V
Radiant intensity	I_F = 50 mA, t_p = 20 ms	I_e			7.5	mW/sr
Peak wavelength	I_F = 100 mA	λ_P	940			nm
Virtual source diameter	Method: 63 % encircled energy	d		1.2		mm
OUTPUT (DETECTOR)						
Collector emitter voltage	I_C = 1 mA	V_{CEO}	32			V
Emitter collector voltage	I_E = 100 µA	V_{ECO}	5			V
Collector dark current	V_{CE} = 20 V, I_F = 0 A, E = 0 lx	I_{CEO}			200	nA

Braking Torque Application – Software Code

```
# Analysis of Braking Torque Application ================
import wx
import time
import datetime
import os
import glob
import webbrowser as wb
from numpy import savetxt,loadtxt,pi,zeros,multiply,sum
from matplotlib.pyplot import plot,figure,subplot,show,grid,xlabel,ylabel,suptitle,close
from matplotlib.backends.backend_wxagg import FigureCanvasWxAgg as FigureCanvas, NavigationToolbar2WxAgg as NavigationToolbar
##import RPi.GPIO as GPIO

#Initialisation of the frame

class console(wx.Frame):
    def __init__(self,parent,id,title):
        wx.Frame.__init__(self,parent,id,title,size=(953,745),style= wx.SYSTEM_MENU | wx.CAPTION | wx.CLOSE_BOX)
##        panel1 = wx.Panel(self)
        panel12 = wx.Panel(self, -1, pos=(10,5),size=(400,700),style= wx.BORDER_SUNKEN)
        Title1 = wx.StaticText(panel12, label='General Data for Hoist Mechanism', pos=(5, 5))
        self.text1 = wx.TextCtrl(panel12, -1, pos=(10,30),size=(45,25),value='51.25')
        Caption1 = wx.StaticText(panel12, label='Motor [rev/s]', pos=(60,35))
        self.text2 = wx.TextCtrl(panel12, -1, pos=(155,30),size=(45,25),value='0.133')
        Caption2 = wx.StaticText(panel12, label='Lifting Speed [m/s]', pos=(205,35))

        self.text3 = wx.TextCtrl(panel12, -1, pos=(10,70),size=(45,25),value='200')
        Caption3 = wx.StaticText(panel12, label='Max. Load [kg]', pos=(60,75))
        self.text4 = wx.TextCtrl(panel12, -1, pos=(155,70),size=(45,25),value='0.075')
        Caption4 = wx.StaticText(panel12, label='Rope Drum Dia. [m]', pos=(205,75))
        self.text5 = wx.TextCtrl(panel12, -1, pos=(10,110),size=(45,25),value='200')
        Caption5 = wx.StaticText(panel12, label='Load [kg]', pos=(60,115))
        line1 = wx.StaticLine(panel12, -1, pos=(5,150), size=(395,3))
        self.text22 = wx.TextCtrl(panel12, -1, pos=(155,110),size=(45,25),value='0.011')
        Caption22 = wx.StaticText(panel12, label='i (Overall Gear Ratio)', pos=(205,115))

        Title2 = wx.StaticText(panel12, label='1st Shaft', pos=(5, 155))
        self.text6 = wx.TextCtrl(panel12, -1, pos=(10,185),size=(45,25),value='0.00047')
        Caption6 = wx.StaticText(panel12, label='J1 [kg.m2]', pos=(60,190))
        self.text7 = wx.TextCtrl(panel12, -1, pos=(155,185),size=(45,25),value='0.0')

        Caption7 = wx.StaticText(panel12, label='J2 [kg.m2]', pos=(205,190))
        self.text8 = wx.TextCtrl(panel12, -1, pos=(290,185),size=(45,25),value='3.115e-7')
        Caption8 = wx.StaticText(panel12, label='J3 [kg.m2]', pos=(340,190))
        self.text9 = wx.TextCtrl(panel12, -1, pos=(10,225),size=(45,25),value='51.25')
        Caption9 = wx.StaticText(panel12, label='1st Shaft Rev. [rev/s]', pos=(60,230))
        line2 = wx.StaticLine(panel12, -1, pos=(5,260), size=(395,3))

        Title3 = wx.StaticText(panel12, label='2nd Shaft', pos=(5, 265))
        self.text10 = wx.TextCtrl(panel12, -1, pos=(10,295),size=(45,25),value='1.72e-4')
        Caption10 = wx.StaticText(panel12, label='J1 [kg.m2]', pos=(60,300))
        self.text11 = wx.TextCtrl(panel12, -1, pos=(155,295),size=(45,25),value='3.11e-7')
        Caption11 = wx.StaticText(panel12, label='J2 [kg.m2]', pos=(205,300))
##        self.text8 = wx.TextCtrl(panel12, -1, pos=(290,295),size=(45,25),value='0.01')
##        Caption8 = wx.StaticText(panel12, label='J3 [kg.m2]', pos=(340,300))
        self.text12 = wx.TextCtrl(panel12, -1, pos=(10,335),size=(45,25),value='5.55')
        Caption12 = wx.StaticText(panel12, label='2nd Shaft Rev. [rev/s]', pos=(60,340))
        line3 = wx.StaticLine(panel12, -1, pos=(5,370), size=(395,3))

        Title4 = wx.StaticText(panel12, label='3rd Shaft', pos=(5, 375))
        self.text13 = wx.TextCtrl(panel12, -1, pos=(10,405),size=(45,25),value='1.72e-4')
        Caption13 = wx.StaticText(panel12, label='J1 [kg.m2]', pos=(60,410))
        self.text14 = wx.TextCtrl(panel12, -1, pos=(155,405),size=(45,25),value='4.8624e-3')
        Caption14 = wx.StaticText(panel12, label='J2 [kg.m2]', pos=(205,410))
        self.text15 = wx.TextCtrl(panel12, -1, pos=(290,405),size=(45,25),value='0')
        Caption15 = wx.StaticText(panel12, label='J3 [kg.m2]', pos=(340,410))
        self.text16 = wx.TextCtrl(panel12, -1, pos=(10,445),size=(45,25),value='0.555')
        Caption16 = wx.StaticText(panel12, label='3rd Shaft Rev. [rev/s]', pos=(60,450))
        line4 = wx.StaticLine(panel12, -1, pos=(5,480), size=(395,3))

        Title5 = wx.StaticText(panel12, label='4th Shaft', pos=(5, 485))
        self.text17 = wx.TextCtrl(panel12, -1, pos=(10,515),size=(45,25),value='0.0')
        Caption17 = wx.StaticText(panel12, label='J1 [kg.m2]', pos=(60,520))
        self.text18 = wx.TextCtrl(panel12, -1, pos=(155,515),size=(45,25),value='0.0')
        Caption18 = wx.StaticText(panel12, label='J2 [kg.m2]', pos=(205,520))
        self.text19 = wx.TextCtrl(panel12, -1, pos=(290,515),size=(45,25),value='0.0')
        Caption19 = wx.StaticText(panel12, label='J3 [kg.m2]', pos=(340,520))
        self.text20 = wx.TextCtrl(panel12, -1, pos=(10,555),size=(45,25),value='0')
        Caption20 = wx.StaticText(panel12, label='3rd Shaft Rev. [rev/s]', pos=(60,560))
        line5 = wx.StaticLine(panel12, -1, pos=(5,590), size=(395,3))
```

```
79
80          Title6 = wx.StaticText(panel12, label='Hoist Motor Standards', pos=(5, 595))
81          button1 = wx.Button(panel12,label="Standard Motor Tables",pos=(10,630),size=(150,40))
82          button2 = wx.Button(panel12,label="Standard Hoist Tables",pos=(170,630),size=(150,40))
83          self.Bind(wx.EVT_BUTTON, self.motor_tables, button1)
84          self.Bind(wx.EVT_BUTTON, self.hoist_tables, button2)
85
86          panel15 = wx.Panel(self, -1, pos=(417,496),size=(520,210),style= wx.BORDER_SUNKEN)
87          Title7 = wx.StaticText(panel15, label='Braking Torque Calculation', pos=(10, 10))
88          line5 = wx.StaticLine(panel15, -1, pos=(260,5), size=(3,235))
89          self.text21 = wx.TextCtrl(panel15, -1, pos=(10,50),size=(45,25),value='40')
90          Caption21 = wx.StaticText(panel15, label='Load at Slip [kg]', pos=(60,55))
91          button3 = wx.Button(panel15,label="Dynamic Braking Torque",pos=(10,90),size=(150,40))
92          button4 = wx.Button(panel15,label="Static Braking Torque",pos=(10,160),size=(150,40))
93          Title8 = wx.StaticText(panel15, label='Diagnosis', pos=(270, 10))
94          button5 = wx.Button(panel15,label="For Dynamic Braking Torque",pos=(280,90),size=(150,40))
95          button6 = wx.Button(panel15,label="For Static Braking Torque",pos=(280,160),size=(150,40))
96          self.Bind(wx.EVT_BUTTON, self.dyn_torque, button3)
97          self.Bind(wx.EVT_BUTTON, self.stat_torque, button4)
98          self.Bind(wx.EVT_BUTTON, self.dyn_diagnosis, button5)
99          self.Bind(wx.EVT_BUTTON, self.stat_diagnosis, button6)
100
101 ## Initial Plot 1 - Angular Displacement Characteristic
102         close("all")
103         panel16 = wx.Panel(self, -1, pos=(415,5),size=(275,250))
104         panel16.figure = figure()
105         panel16.figure.set_size_inches((7.0,2.1))
106         panel16.figure.set_dpi(80)
107         panel16.axes = panel16.figure.add_subplot(111)
108         panel16.figure.subplots_adjust(top=0.86, bottom=0.22)
109         #rad0 = 0.0
110         #t0 = 0.0
111         rad = loadtxt('rad.txt')
112         t = loadtxt('time(1).txt') #With individual time file
113 ##        rad,t =loadtxt('rad.txt',unpack=True) #Unpacking of variables from measurement file
114         panel16.y_max=max(rad)
115         panel16.axes.plot(t,rad),grid(),xlabel('time[s]'),ylabel('angle [rad]'),suptitle('Angular Displacement Characteristic')
116 #       panel16.axes.axis((0.0,0.0,1.0,1.0))
117         panel16.canvas = FigureCanvas(panel16, -1, panel16.figure)
118         panel16.toolbar = NavigationToolbar(panel16.canvas)
```

```
119         panel16.toolbar.Realize()
120         panel16.sizer=wx.BoxSizer(wx.VERTICAL)
121         panel16.sizer.Add(panel16.canvas, 1, wx.LEFT | wx.TOP | wx.GROW)
122         panel16.sizer.Add(panel16.toolbar, 0, wx.LEFT | wx.EXPAND)
123         panel16.SetSizer(panel16.sizer)
124         panel16.Fit()
125
126 ## Initial Plot 2 - Temperature Characteristic
127         close("all")
128         panel17 = wx.Panel(self, -1, pos=(415,252),size=(275,250))
129         panel17.figure = figure()
130         panel17.figure.set_size_inches((7.0,2.1))
131         panel17.figure.set_dpi(80)
132         panel17.axes = panel17.figure.add_subplot(111)
133         panel17.figure.subplots_adjust(top=0.86, bottom=0.22)
134 #        t=loadtxt('t.txt')
135         temp=loadtxt('temp.txt')
136 ##        panel17.y_max=max(temp)
137         panel17.axes.plot(temp),grid(),xlabel('Samples'),ylabel('t [Celcius]'),suptitle('Gearbox Temperature')
138         panel17.canvas = FigureCanvas(panel17, -1, panel17.figure)
139         panel17.toolbar = NavigationToolbar(panel17.canvas)
140         panel17.toolbar.Realize()
141         panel17.sizer=wx.BoxSizer(wx.VERTICAL)
142         panel17.sizer.Add(panel17.canvas, 1, wx.LEFT | wx.TOP | wx.GROW)
143         panel17.sizer.Add(panel17.toolbar, 0, wx.LEFT | wx.EXPAND)
144         panel17.SetSizer(panel17.sizer)
145         panel17.Fit()
146
147         button7 = wx.Button(self,label="Hoist Deceleration Measurement",pos=(490,208),size=(185,40))
148         button8 = wx.Button(self,label="Plot Data",pos=(690,208),size=(185,40))
149         self.Bind(wx.EVT_BUTTON, self.dec_measure, button7)
150         self.Bind(wx.EVT_BUTTON, self.dec_data, button8)
151
152         button9 = wx.Button(self,label="Gearbox Temperature Measurement",pos=(490,455),size=(185,40))
153         button10 = wx.Button(self,label="Plot Data",pos=(690,455),size=(185,40))
154         self.Bind(wx.EVT_BUTTON, self.temp_measure, button9)
155         self.Bind(wx.EVT_BUTTON, self.temp_data, button10)
156
```

```
157     def motor_tables(self,event):
158         #Open standard hoist motor tables
159         wb.open('http://www.sew-eurodrive.com/download/pdf/16936426.pdf')
160 
161     def hoist_tables(self,event):
162         #Open standards hoist data tables
163         wb.open('http://www.gude.cz/upload/File/el-lanove-navijaky.pdf')
164 
165     def dyn_torque(self,event):
166         decel = self.decel_calc()
167         Jeq = self.dyn_torque_calc()
168 
169         Rope_Dia = float(self.text4.GetValue())
170         Load_Slip = float(self.text21.GetValue())
171         Gear_Ratio = float(self.text22.GetValue())
172 
173         ximech = ((0.5*Rope_Dia*9.81*Load_Slip)*Gear_Ratio) #Mechanical loss torque [N.m]
174         Bt = zeros(len(decel))
175 
176         for k in range(0,len(decel)):
177 
178             Bt[k] = multiply(Jeq, decel[k]) - ximech
179 
180         print Bt
181 
182         close("all")
183         figure()
184         subplot(111)
185         plot(Bt,'b'),grid(), xlabel('Samples at Deceleration'), ylabel('Braking Torque [N.m]'), suptitle('Dynamic Braking Torque
186         show()
187 
188     def stat_torque(self,event):
189         decel = self.decel_calc()
190         Jeq = self.dyn_torque_calc()
191 
192         Rope_Dia = float(self.text4.GetValue())
193         Load_Slip = float(self.text21.GetValue())
194         Gear_Ratio = float(self.text22.GetValue())
195 
```


```
157     def motor_tables(self,event):
158         #Open standard hoist motor tables
159         wb.open('http://www.sew-eurodrive.com/download/pdf/16936426.pdf')
160 
161     def hoist_tables(self,event):
162         #Open standards hoist data tables
163         wb.open('http://www.gude.cz/upload/File/el-lanove-navijaky.pdf')
164 
165     def dyn_torque(self,event):
166         decel = self.decel_calc()
167         Jeq = self.dyn_torque_calc()
168 
169         Rope_Dia = float(self.text4.GetValue())
170         Load_Slip = float(self.text21.GetValue())
171         Gear_Ratio = float(self.text22.GetValue())
172 
173         ximech = ((0.5*Rope_Dia*9.81*Load_Slip)*Gear_Ratio) #Mechanical loss torque [N.m]
174         Bt = zeros(len(decel))
175 
176         for k in range(0,len(decel)):
177 
178             Bt[k] = multiply(Jeq, decel[k]) - ximech
179 
180         print Bt
181 
182         close("all")
183         figure()
184         subplot(111)
185         plot(Bt,'b'),grid(), xlabel('Samples at Deceleration'), ylabel('Braking Torque [N.m]'), suptitle('Dynamic Braking Torque
186         show()
187 
188     def stat_torque(self,event):
189         decel = self.decel_calc()
190         Jeq = self.dyn_torque_calc()
191 
192         Rope_Dia = float(self.text4.GetValue())
193         Load_Slip = float(self.text21.GetValue())
194         Gear_Ratio = float(self.text22.GetValue())
195 
```

```
232
233     def stat_diagnosis(self,event):
234         decel = self.decel_calc()
235         Jeq = self.dyn_torque_calc()
236
237         Rope_Dia = float(self.text4.GetValue())
238         Load_max = float(self.text3.GetValue())
239         Load_Slip = float(self.text21.GetValue())
240         Gear_Ratio = float(self.text22.GetValue())
241
242         ximech = ((0.5*Rope_Dia*9.81*Load_Slip)*Gear_Ratio) #Mechanical loss torque [N.m], must eq. to rotor
243         Bt_diag_st = zeros(len(decel))
244
245         for k in range(0,len(decel)):
246
247             Bt_stat = (1.6/1.2)*(multiply(Jeq, decel[k]) - ximech)
248             Bt_diag_st[k] = Bt_stat - ((1.6*9.81*Load_max*Rope_Dia*0.5)*Gear_Ratio)   #must divide by gear ratio
249
250             if Bt_diag_st[k] > 0:
251                 print Bt_diag_st[k],"OK"
252             elif Bt_diag_st[k] < 0:
253                 print Bt_diag_st[k],"NOT OK"
254
255
256 ## Calculation function for dynamic braking torque
257
258     def decel_calc(self):
259
260         rad = loadtxt('rad.txt')
261         time = loadtxt('time(1).txt')
262         rev = float(self.text1.GetValue())
263
264         N = len(rad)
265         speed = zeros(N)
266         decel = zeros(N)
267
268         for k in range(1,N):
269             speed[0]=rev*2*pi
270             speed[k]=(rad[k]-rad[k-1])/(time[k]-time[k-1])
```

```
271             decel[k]=(speed[k-1]-speed[k])/(time[k]-time[k-1])
272
273 ##        print speed, decel
274
275         return decel
276
277     def dyn_torque_calc(self):
278
279         Nmotor = float(self.text1.GetValue())
280         m = float(self.text5.GetValue())
281         Vh = float(self.text2.GetValue())
282
283         ## 1st Shaft - Rotational
284         J11 = float(self.text6.GetValue())
285         J21 = float(self.text7.GetValue())
286         J31 = float(self.text8.GetValue())
287         rev1 = float(self.text9.GetValue())
288
289         Jeq1_rot1 = J11*((rev1/Nmotor)**2)
290         Jeq2_rot1 = J21*((rev1/Nmotor)**2)
291         Jeq3_rot1 = J31*((rev1/Nmotor)**2)
292
293         Jeq1 = Jeq1_rot1 + Jeq2_rot1 + Jeq3_rot1
294
295
296         ## 2nd Shaft - Rotational
297         J12 = float(self.text10.GetValue())
298         J22 = float(self.text11.GetValue())
299         rev2 = float(self.text12.GetValue())
300
301         Jeq1_rot2 = J12*((rev2/Nmotor)**2)
302         Jeq2_rot2 = J22*((rev2/Nmotor)**2)
303
304         Jeq2 = Jeq1_rot2 + Jeq2_rot2
305
306         ## 3rd Shaft
307         J13 = float(self.text13.GetValue())
308         J23 = float(self.text14.GetValue())
309         J33 = float(self.text15.GetValue())
310         rev3 = float(self.text16.GetValue())
```

```
311
312         Jeq1_rot3 = J13*((rev3/Nmotor)**2)
313         Jeq2_rot3 = J23*((rev3/Nmotor)**2)
314         Jeq3_rot3 = J33*((rev3/Nmotor)**2)
315
316         Jeq3 = Jeq1_rot3 + Jeq2_rot3 + Jeq3_rot3
317
318         ## 4th Shaft
319         J14 = float(self.text17.GetValue())
320         J24 = float(self.text18.GetValue())
321         J34 = float(self.text19.GetValue())
322         rev4 = float(self.text20.GetValue())
323
324         Jeq1_rot4 = J14*((rev4/Nmotor)**2)
325         Jeq2_rot4 = J24*((rev4/Nmotor)**2)
326         Jeq3_rot4 = J34*((rev4/Nmotor)**2)
327
328         Jeq4 = Jeq1_rot4 + Jeq2_rot4 + Jeq3_rot4
329
330         Jeq_rot = Jeq1 + Jeq2 + Jeq3 + Jeq4
331
332         Jeq_lin = m*((Vh/(2*pi*Nmotor))**2)
333
334         Jeq = sum(Jeq_rot, Jeq_lin)
335
336         return Jeq
337
338     def temp_measure(self,event):
339         ##pi@raspberrypi ~ $ cd /sys/bus/w1/devices/
340         ##pi@raspberrypi /sys/bus/w1/devices $ sudo modprobe w1-gpio
341         ##pi@raspberrypi /sys/bus/w1/devices $ sudo modprobe w1-therm
342         ##pi@raspberrypi /sys/bus/w1/devices $ cd /sys/bus/w1/devices/
343         ##pi@raspberrypi /sys/bus/w1/devices $ ls
344
345         ##====Enabling Bus Interface for Sensor=========
346
347         os.system('sudo modprobe w1-gpio')
348         os.system('sudo modprobe w1-therm')
349
```

```
350         path = glob.glob('/sys/bus/w1/devices/28-000004a7fa45')[0]
351         sensor_file = path + '/w1_slave'
352
353         ##====Measurement with DS18B20 Temp. Sensor=========
354
355         def measure():
356             f = open(sensor_file, 'r')
357             data = f.readlines()
358             f.close()
359             return data
360
361         def temp_calc():
362             data = measure()
363             temp = data[1][-6:-1]
364             print temp
365             return temp
366
367         ##f = open('temp.txt', 'w')
368
369         N=10
370         temp = zeros(N)
371         t = zeros(N)
372         for k in range(N):
373             try:
374                 temp[k]=float(temp_calc())
375                 t[k] = datetime.datetime.now()
376                 ##Saving Measurements to File
377         ##        temp,t = (str(temp_calc()),str(datetime.datetime.now()))
378         ##        f.write('{} {}\n'.format(temp,t))
379                 time.sleep(1)
380
381             except:
382                 KeyboardInterrupt()
383         ##        f.close()
384
385         savetxt('temp.txt',temp)
386         savetxt('t.txt',t)
387
388
```

```
    def temp_data(self,event):
#         close("all")
        panel17 = wx.Panel(self, -1, pos=(415,252),size=(275,250))
        panel17.figure = figure()
        panel17.figure.set_size_inches((7.0,2.1))
        panel17.figure.set_dpi(80)
        panel17.axes = panel17.figure.add_subplot(111)
        panel17.figure.subplots_adjust(top=0.86, bottom=0.22)
#         t=loadtxt('t.txt')
        temp=loadtxt('temp.txt')
##          panel17.y_max=max(temp)
        panel17.axes.plot(temp),grid(),xlabel('Samples'),ylabel('t [Celcius]'),suptitle('Gearbox Temperature')
        panel17.canvas = FigureCanvas(panel17, -1, panel17.figure)
        panel17.toolbar = NavigationToolbar(panel17.canvas)
        panel17.toolbar.Realize()
        panel17.sizer=wx.BoxSizer(wx.VERTICAL)
        panel17.sizer.Add(panel17.canvas, 1, wx.LEFT | wx.TOP | wx.GROW)
        panel17.sizer.Add(panel17.toolbar, 0, wx.LEFT | wx.EXPAND)
        panel17.SetSizer(panel17.sizer)
        panel17.Fit()

    def dec_measure(self,event):
        # Use BCM GPIO references
        GPIO.setmode(GPIO.BCM)

        GPIO_In = 23
        ## GPIO 4 = Pin 16

        # Set pins as input
        GPIO.setup(GPIO_In,GPIO.IN)
        GPIO.add_event_detect(GPIO_In, GPIO.BOTH)

        rad=0
        t0=datetime.datetime.now()
        print rad,t0
        f = open('rad.txt','w')
        f.write('{} {}\n'.format(rad,t0))

        try:
            while True:
                if GPIO.event_detected(GPIO_In):
                    rad += (6*pi)/180 #6 degrees per divsion of coded wheel/ring
                    time=datetime.datetime.now()
                    f.write('{} {}\n'.format(rad,time))
                    print rad,time

        except KeyboardInterrupt:
            GPIO.cleanup()
            f.close()

    def dec_data(self,event):
#         close("all")
        panel16 = wx.Panel(self, -1, pos=(415,5),size=(275,250))
        panel16.figure = figure()
        panel16.figure.set_size_inches((7.0,2.1))
        panel16.figure.set_dpi(80)
        panel16.axes = panel16.figure.add_subplot(111)
        panel16.figure.subplots_adjust(top=0.86, bottom=0.22)
        rad = loadtxt('rad.txt')
        t = loadtxt('time(1).txt') #With individual time file
##          rad,t =loadtxt('rad.txt',unpack=True) #Unpacking of variables from measurement file
        panel16.y_max=max(rad)
        panel16.axes.plot(t,rad),grid(),xlabel('time[s]'),ylabel('angle [rad]'),suptitle('Angular Displacement Characteristic')
        panel16.canvas = FigureCanvas(panel16, -1, panel16.figure)
        panel16.toolbar = NavigationToolbar(panel16.canvas)
        panel16.toolbar.Realize()
        panel16.sizer=wx.BoxSizer(wx.VERTICAL)
        panel16.sizer.Add(panel16.canvas, 1, wx.LEFT | wx.TOP | wx.GROW)
        panel16.sizer.Add(panel16.toolbar, 0, wx.LEFT | wx.EXPAND)
        panel16.SetSizer(panel16.sizer)
        panel16.Fit()


app = wx.App()
frame = console(None,-1,"Braking Torque Analysis Application")
frame.Show()
app.MainLoop()
```

PIC Microcontroller Program (using MPLABX) for Optical Reflective Sensor

```
#if defined(__XC)
    #include <xc.h>         /* XC8 General Include File */
#elif defined(HI_TECH_C)
    #include <htc.h>        /* HiTech General Include File */
#endif

#include <stdint.h>         /* For uint8_t definition */
#include <stdbool.h>        /* For true/false definition */

/******************************************************************************/
/* Interrupt Routines                                                         */
/******************************************************************************/

/* Baseline devices don't have interrupts. Note that some PIC16's
 * are baseline devices.  Unfortunately the baseline detection macro is
 * _PIC12 */
#ifndef _PIC12

extern uint8_t counter;

void interrupt isr(void)
{
    /* This code stub shows general interrupt handling.  Note that these
    conditional statements are not handled within 3 seperate if blocks.
    Do not use a seperate if block for each interrupt flag to avoid run
    time errors. */


    /* TODO Add interrupt routine code here. */

    /* Determine which flag generated the interrupt */

    if(INTCONbits.INTF)
    {
        INTCONbits.INTF = 0;                //Enable Global Interrupt
        counter++;

         if (counter >= 1) {
             PORTAbits.RA0 = ~PORTAbits.RA0;
             counter = 0;
         }


    }
    else if(TMR1IF)
    {
        TMR1IF = 0 ;                    //Enable Global Interrupt
        PORTAbits.RA1 = ~PORTAbits.RA1;


    }
    else
    {
        /* Unhandled interrupts */
    }


}
#endif
```