



**UNIVERSITÀ
DEGLI STUDI
DI PADOVA**



**DIPARTIMENTO
DI INGEGNERIA
DELL'INFORMAZIONE**

DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

**CORSO DI LAUREA MAGISTRALE IN
ICT FOR INTERNET AND MULTIMEDIA**

“CREATING A DIGITAL TWIN OF RCCB CALIBRATION PROCESS ”

Relatore: Prof. / Dott Federica Battisti

Laureando: Wishnuputra Dhanu

ANNO ACCADEMICO 2020 – 2022

Data di laurea 13 dicembre 2022

Abstract

Residual Current Circuit Breaker (RCCB) is a safety device that will protect human or electrical wiring from electrical fault. One of the most important task in manufacturing the RCCB is the calibration process. In order to optimize the calibration algorithm, several parameters need to be adjusted to achieve faster and more efficient calibration process. However, at the moment adjusting the calibration parameters is a risky task. This is due to the fact that if the calibration parameters that is adjusted does not bring improvement or even worse makes the calibration process even slower. It will impact directly on the production line that are manufacturing thousands of RCCB devices everyday. Therefore it is a very risky task and we need to be very careful when adjusting these parameters.

In order to solve this problems, we will create a digital twin application that will simulate the calibration process of RCCB device. A digital twin is a digital copy of the physical object, in this case is the RCCB calibration process. With this digital twin application the engineers can safely calibrate the calibration parameters without any risk in the real production process. The other benefit is that there will be greater freedom for the engineers to experiment various calibration parameters, because they can simulate the result first before implementing it in the real production process.

One of the most important task in the calibration process is to find the demagnetization voltage that will calibrate the RCCB device into the correct residual current. Currently, this is achieved by calculating using a static mathematical expression. However the limitations of this approach is the mathematical formula is fixed unable to act according to the real condition in the calibration process. In order to overcome this limitations, this project will build a artificial neural network for predicting the demagnetization that will be used in the calibration process.

Contents

ABSTRACT	v
LIST OF FIGURES	ix
LIST OF TABLES	xi
LISTING OF ACRONYMS	xiii
1 INTRODUCTION	1
1.1 Background	1
1.2 Current Situation	2
1.3 Research Goals	2
1.4 Outline of the Thesis	2
2 LITERATURE REVIEW	3
2.1 RCCB	3
2.2 Operation Principle of RCCB	6
2.2.1 Differential Current Sensor	7
2.2.2 Differential Transformer	7
2.2.3 Filter Circuit	9
2.2.4 Electromechanical Relay	9
2.3 RCCB Calibration System	10
2.4 Machine Learning	11
2.5 Deep Learning	12
2.6 RCCB Calibration Process	13
2.6.1 Working Principle of Electromechanical Relay	13
2.6.2 Calibration Algorithm	14
3 DEVICE BEHAVIORS	17
3.1 Observing the Calibration Dataset	17
3.2 Inaccuracy of Current Measurement	27
4 DIGITAL TWIN MODEL	29
4.1 Digital Twin Implementation	29
4.1.1 Digital Twin Design	29
4.1.2 Calibration Dataset	31

4.1.3	Prediction Model	32
4.2	Digital Twin Exponential Model	33
4.2.1	Finding parameter for the exponential model	34
4.2.2	Evaluation Result of Exponential Model	35
4.3	Digital Twin Polynomial Model	38
4.3.1	Polynomial Model Degree 5	39
4.3.2	Polynomial degree 8	41
4.3.3	Evaluation Result of Polynomial Degree 8 Model	42
4.4	Digital Twin ML.NET Model	45
4.4.1	Creating the ML Model	46
5	NEURAL NETWORK MODEL FOR PREDICTING DEMAGNETIZATION VOLTAGE	55
5.1	Model 0	56
5.1.1	Dataset	57
5.1.2	Normalization of the data	58
5.1.3	Design of Model 0	59
5.1.4	Training Process	60
5.2	Model 1	62
5.2.1	Dataset	63
5.2.2	Normalization of the data	63
5.2.3	Design of Model 1	64
5.2.4	Training Process	64
5.3	Model 2	67
5.3.1	Dataset	67
5.3.2	Normalization of the data	68
5.3.3	Design of Model 2	68
5.3.4	Training Process	70
6	CONCLUSION	73
	REFERENCES	79
	ACKNOWLEDGMENTS	81

Listing of figures

2.1	Direct Contact	5
2.2	ABB F200 Residual Current Circuit Breaker	6
2.3	RCCB Schema	7
2.4	Differential Transformer Winding	8
2.5	Differential Transformer	8
2.6	Filter Circuit	9
2.7	Electromechanical Relay	10
2.8	RCCB Calibration System	11
2.9	Artificial Intelligence	12
2.10	Electromechanical Relay	14
2.11	RCCB Calibration Algorithm	15
3.1	Calibration Graph of Device 001RMG9S	18
3.2	Calibration Graph of Device 001RMG5C	19
3.3	Calibration Graph of Device 001RMG1P	20
3.4	Calibration Graph of Device 001RMG6Q	20
3.5	Calibration Graph of Device 001RMG60	21
3.6	Calibration Process of 001RMFV7 and 001RMG36	22
3.7	Calibration Process of Several Devices	23
3.8	Devices with similar initial current	24
3.9	Devices with repeated current measurement	25
3.10	Calibration process of multiple devices with service code 405030017303	26
3.11	Ramp current	27
4.1	Calibration process in the real system	30
4.2	Calibration process in the digital twin	31
4.3	Calibration process of several devices with different serial number	32
4.4	Model for predicting the residual current	33
4.5	Curve fitting of the exponential model	34
4.6	Plot of the digital twin result using exponential model	36
4.7	Plot of the real calibration data of service code 405030017303	37
4.8	Curve Fitting of Polynomial Degree 5	40
4.9	Curve Fitting of Polynomial Degree 8	41
4.10	Plot of the digital twin result using polynomial 8 model	43
4.11	Plot of the calibration data of service code: 405030017303	44

4.12	Graph of ML.NET Model after training	50
4.13	Sampling the result of ML.NET Model after training	51
4.14	Digital Twin simulation using ML.NET model	52
4.15	Plot of real calibration data of service code 405030017303	52
5.1	Graph of calibration dataset with service code 405030017303	56
5.2	Design of Model 0	57
5.3	Split the dataset into train and test set	58
5.4	Training loss and validation loss of Model 0	60
5.5	Test set for evaluating Model 0	61
5.6	Prediction error of Model 0	62
5.7	Design of Model 1	62
5.8	Design of Neural Network Model 1	64
5.9	Training loss and validation loss of Model 1	65
5.10	Accuracy of Model 1	66
5.11	Prediction Error of Model 1	66
5.12	Design of Model 2	67
5.13	Design of Neural Network Model 2	69
5.14	Model 2 Training and Validation Loss	70
5.15	Accuracy of Model 2	71
5.16	Prediction Error of Model 2	71
6.1	Evaluation Result	74
6.2	Evaluation Result	76

Listing of tables

3.1	Calibration Information of Device 001RMG9S	17
3.2	Calibration Information of Device 001RMG5C	18
3.3	Calibration Information of Device 001RMG1P	19
3.4	Calibration Information of Device 001RMG6Q	20
3.5	Calibration Information of Device 001RMG6o	21
3.6	Calibration Information of Device 001RMFV	22
4.1	Calibration Information of Device 001RMG9S	31
4.2	Calibration Information of Device 001RMG9S	39
4.3	Calibration Information of Device 001RMG9S	46
4.4	Calibration Information of Device 001RMG9S	53
5.1	Calibration Information of Device 001RMG9S	56

Listing of acronyms

RCCB Residual Current Circuit Breaker

RCD Residual Current Device

ML Machine Learning

1

Introduction

1.1 BACKGROUND

Residual Current Circuit Breaker (RCCB) is a safety device designed to break electric circuit when there is earth fault. It has a critical function to prevent death risk to someone and protect electrical devices.

One of the most important process in the manufacturing of the RCCB devices is the calibration process. Before calibration, the RCCB has residual current higher than the desired operating residual current. Therefore it has to be calibrated until it reaches the desired operating residual current.

In this digital era, many industries are eager to create digital twin of their production process. A digital twin is a virtual model designed to accurately reflect a physical object.[1]

The main objective of creating a digital twin is to simulate the process of a certain object digitally without running the physical object. The benefit of using a digital twin we can make some adjustment to the digital object and simulate the results without risking the real object.

In this project we will build a digital twin that will simulate the calibration process of a RCCB (Residual Current Circuit Breaker)

1.2 CURRENT SITUATION

The calibration process of RCCB could take longer than usual. Therefore calibration parameters need to be adjusted to reduce the calibration time.

Currently the calibration parameters need to be adjusted directly into the production line. Therefore there is a risk that if the calibration parameter is not correctly adjusted, the calibration result will get even worse. This means it took longer time to calibrate the RCCB devices.

1.3 RESEARCH GOALS

There are three goals in this project. The first goal is to identify the behaviors of the device during the calibration process. The second goal is to design and create a digital twin application that simulate the training process of the RCCB. The third goal is to implement artificial neural network model to predict the value of demagnetizing voltage (V_x) based on the input residual current I_x .

1.4 OUTLINE OF THE THESIS

This document consists of nine chapters. Chapter one is this chapter which is the introduction. Chapter two is Literature Review which consist of the theoretical aspect of the thesis project. Chapter three explains about several behaviors of the device during the calibration process. Chapter four explains about the design and implementation of the digital twin application. Chapter five, six, and seven explains several models that are used to simulate the calibration process of the RCCB device. Chapter eight explains artificial neural network model to predict the demagnetizing voltage (V_x). The last chapter is conclusion which explain the summary and future development of the project.

2

Literature Review

2.1 RCCB

A Residual Current Circuit Breaker is a safety measure device in the protection of electrical circuit. It functions as a current sensing device, which automatically measure and disconnect the electrical circuit whenever a fault is detected[2].

Function of RCCB:

1. Provide safety and protection against earth fault.
2. Provide protection against leakage current
3. Automatically disconnects the electrical circuit when the current exceeds the rated sensitivity.

Terms related with RCCB according to IEC 62873 [3]:

Earth fault current is a current flowing through the electrical equipment and eventually to the earth due to insulation fault of the electrical equipment.

Earth leakage current is current that is flowing from the live wire of the installation system to the earth without flowing through the electrical equipment.

Circuit breaker is a mechanical switching device, capable of breaking currents under normal circuit conditions and also carrying for a specified duration and breaking currents under specified abnormal circuit conditions such as those of short-circuit.

Residual current device (RCD) is a mechanical switching device or association of device designed to make, carry and break currents under normal service conditions and to cause the opening of the contacts when the residual current attains a given value under specified conditions.

Residual current operated circuit breaker without integral overcurrent protection (RCCB) is a residual current operated circuit breaker designed to perform the functions of protection against overloads and/or short circuits.

Break time (of an RCD) is a time which elapses between the instant when the residual operating current is suddenly attained and the instant of arc extinction in all poles.

RCCB provides accurate protection of people and electrical equipment against leakage current. RCCB detects an imbalance of the electrical flow and trips when the current flowing exceeds the rated sensitivity. This safety mechanism reduces the risk of death or serious injury. It also prevents the risk of fire which is normally caused by faulty wiring. RCCB is a safety device to detect and trip against electrical leakage currents. It ensures protection against electric shock caused by indirect contacts.

The main purpose of RCCB is to protect people from direct and indirect contacts. Direct contact happens when a person accidentally comes into contact with a live phase which is under tension. This is normally due to a broken socket, stripped cable, etc. Indirect contact means when a person is in contact with a normally insulated metal mass (e.g. casing of a household appliance) which comes into contact with a phase due to a fault. In both cases there is current imbalance between the power supply line and the neutral line. This imbalance is detected by the sensing devices.

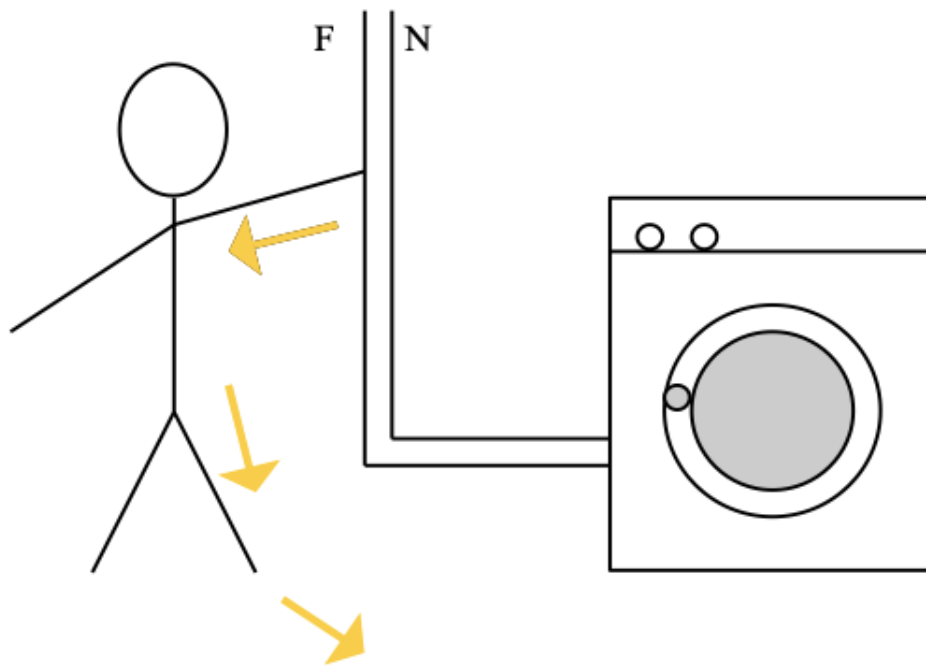


Figure 2.1: Direct Contact

In order for electrocution to occur, there is passage of current through the human body where the circuit constituted by the human body be closed which mean there is a way for the current to return. This path is usually the earth, which has zero potential, reached by feet resting on the ground or a hand touching a wall or a metal object connected to the ground. Humid environments and bare feet increase the dangers of direct contact while dry environments and rubber shoes will reduce the risk.

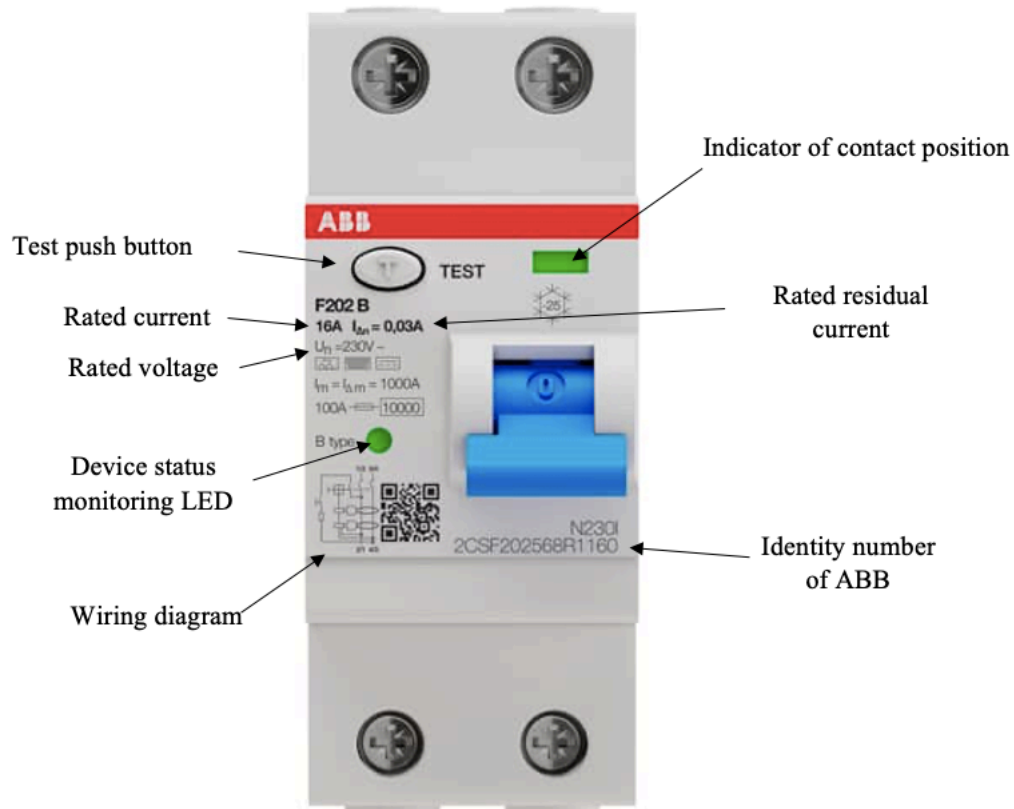


Figure 2.2: ABB F200 Residual Current Circuit Breaker

The figure 2.2 shows the RCCB of model F200 and its features from the internal ABB documentation [2]. Rated current is the desired residual current that will trip the RCCB. Indicator of contact position shows the status of the RCCB. Green colour indicates that the RCCB is in off position. Red colour indicates that the RCCB is in on position.

2.2 OPERATION PRINCIPLE OF RCCB

According to the principle of operation, RCCB can be represented as the following schematic [2]. Input terminals is connected to the electrical system that will be protected by the RCCB. Mechanical contact is the switch that is used to turn on or off the RCCB.

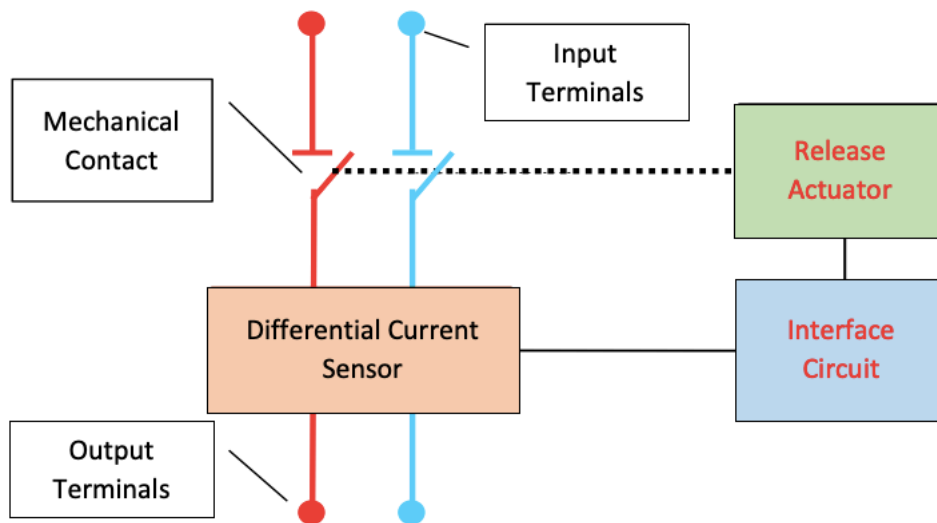


Figure 2.3: RCCB Schema

2.2.1 DIFFERENTIAL CURRENT SENSOR

The role of differential current sensor is to measure amount of the residual current that is flowing through the circuit. The sensor consist of a toroidal core made of magnetic material, a primary winding, and a secondary winding. The primary winding physically consists of several windings of wire throughout the magnetic core with the same number of turns and winding directions. When there is no earth leakage, the sum of the current in the conductors is zero. Therefore, the corresponding contributions to the magnetic flux in the magnetic core cancel out each other. In case there is earth leakage, there will be current flowing on the magnetic core, hence generating magnetic flux and induces a voltage on the secondary winding. The primary current is the earth leakage differential current where if it exceeds a certain value will trigger the RCCB to open.

2.2.2 DIFFERENTIAL TRANSFORMER

The residual current $I\Delta$ between the neutral wire and live wire generates a magnetic field. The variation of magnetic flux in the core induces a voltage in the secondary winding which produces current I_2 that will then flow into the filter circuit.

Figure 2.5 shows the core winding of the differential transformer. The core is made of high permeability materials and is characterized by a constant cross section. The parameters that

characterized the core are internal diameter (D_i), external diameter (D_e), and height (h).

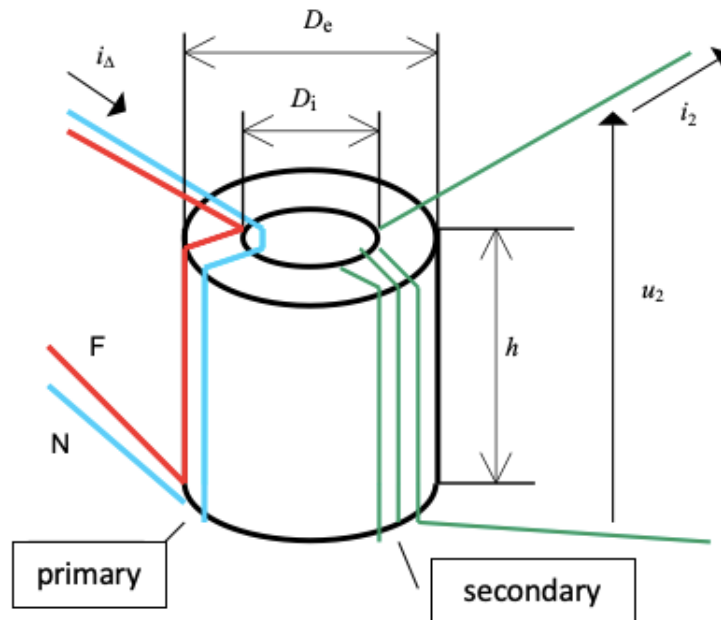


Figure 2.4: Differential Transformer Winding

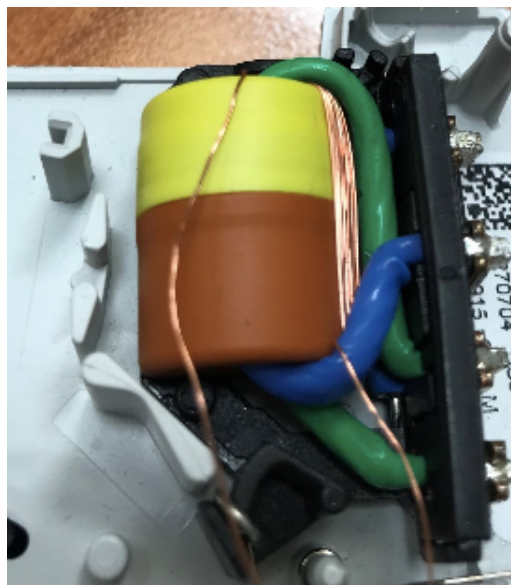


Figure 2.5: Differential Transformer

2.2.3 FILTER CIRCUIT

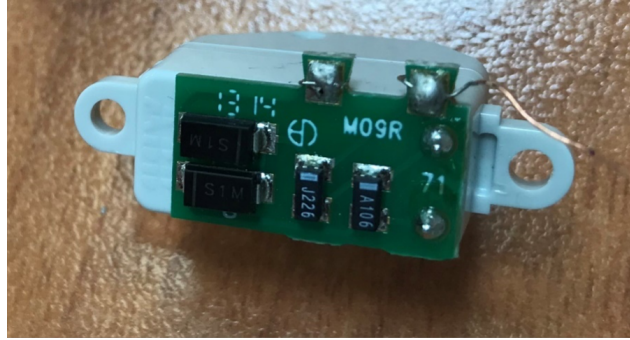


Figure 2.6: Filter Circuit

The filter circuit is an electronic board that is located between the differential transformer and the electromechanical relay. The filter circuit will act as a low-pass filter that prevents high frequency signal from the external components before entering the electromechanical relay[4]. The diodes D₁ and D₂ will have a role to protect the device against electric shocks.

2.2.4 ELECTROMECHANICAL RELAY

The figure 2.7 shows an electromechanical relay which is part of the RCCB[5]. The main function of the electromechanical relay is to break the circuit when the residual current exceeds a predetermined value. The electromechanical relay works when there is residual current detected by the differential transformer. Then the current from the secondary coil flows into filter circuit. Finally, the current will then flow into the electromechanical relay. The current will enter the magnetic coil. The magnetic coil will produce a force that will attract the floating anchor. If the force is larger than the magnetic force of the permanent magnet that attract the floating anchor on the other side, the release pin will be activated. This pin will force the RCCB switch to trip.

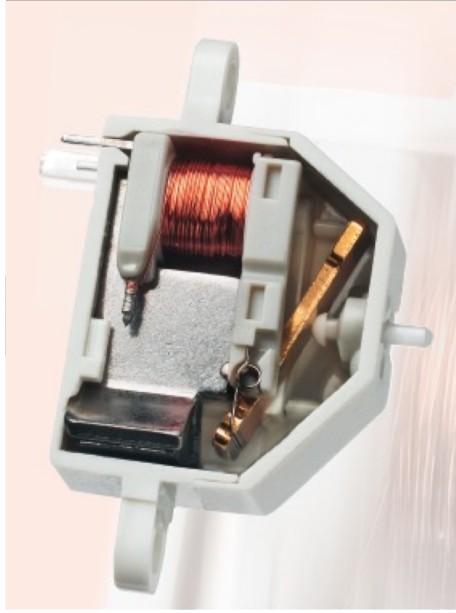


Figure 2.7: Electromechanical Relay

In normal working operation, the vectorial sum of all currents is zero. When there is an unbalanced condition due to a ground fault, a magnetic flux in the magnetic core is produced and electromotive force is present in the secondary winding [6].

The magnetic core is specifically fabricated with composition of 50%Ni and 50%Fe alloy called PERMENORM and it has a complicated asymmetric shape that has been designed to provide an appropriate magnetic field distribution. The magnetic core is made of two parts with a small air gap in between. The main role of the air gap is to provide two different paths to the magnetic flux. The permanent magnet that is used to produce the DC Flux is a SIMAGx-registered magnet made of AlNiCo 19/11 material [5].

2.3 RCCB CALIBRATION SYSTEM

The calibration system consists of Voltage Generator, Calibration Coil, Ramp Current Generator, the RCCB device, Current Measurement, Calibration Algorithm. Figure 2.8 shows the calibration system of the RCCB at ABB production plant in Santa Palomba.



Figure 2.8: RCCB Calibration System

2.4 MACHINE LEARNING

Machine learning is a field of inquiry devoted to understanding and building methods that 'learn', that is, methods that leverage data to improve performance on some set of tasks[7]. It is a branch of Artificial Intelligence (AI) and computer science that focus on using data and algorithms to imitate the way that humans learn and gradually improving its accuracy.

Machine learning is a field that devote to build a method that leverage data to improve the performance of some tasks. Machine learning build a model based on sample data, known as training data. Then it will use this data to make predictions or decision without being explicitly

programmed to do so.

Modern-day machine learning has two objectives. First is to classify data based on models which have been developed.[8] Second is to make predictions for future outcomes based on the models. For example, a machine learning algorithm for stock trading may inform the trader of future potential predictions.

In this project we will use machine learning for predictions purpose. We are going to use machine learning to predict the value of demagnetization voltage (V_x) based on the value of initial residual current (I_o) and residual current (I_x). We are going to train the machine learning model using the real calibration data from the production line.

2.5 DEEP LEARNING

Deep learning (also known as deep structured learning) is part of a broader family of machine learning methods based on artificial neural networks with representation learning. Learning can be supervised, semi-supervised or unsupervised.[9]

The figure 2.9 shows where Deep Learning is part of Machine Learning which is also part of Artificial Intelligence. Deep Learning is a class of machine learning algorithms that uses artificial neural network for classification and prediction purpose.

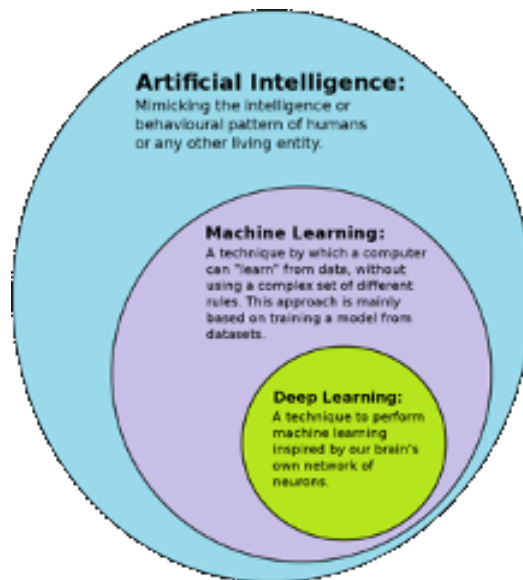


Figure 2.9: Artificial Intelligence

2.6 RCCB CALIBRATION PROCESS

Each RCCB devices will be calibrated before they are shipped to customer. The RCCB will be calibrated according to each model and its specification. This section will explain the steps on how the RCCB is calibrated.

2.6.1 WORKING PRINCIPLE OF ELECTROMECHANICAL RELAY

Figure 2.10 shows the main components of an electromechanical relay. Electromechanical relay has a role of converting electric force into mechanical force in order to start the opening of the circuit breaker [6].

In the figure 2.10, the permanent magnet C preloads the magnetic circuit A in order to maintain the floating anchor B closed by compensating the attractive force of the spring D. In normal conditions, the floating anchor is attached to the magnetic circuit due to the attractive force originated by the magnetic flux of the permanent magnet. During fault conditions, the coil E is energized. An alternated magnetic flux is superimposed to the DC flux of the magnet. When the mechanic force of the spring exceeds the magnetic force, the anchor moves the release F and causes the tripping of the circuit breaker [6].

At the beginning before the calibration process started, the permanent magnet has magnetization level higher than the desired magnetization level. The system calibrate the residual current device by applying damped alternating magnetic field to demagnetize the permanent magnet. As a result the magnetic field in the electromechanical relay is reduced. Therefore the magnetic force that keeps the relay closed is also reduced. Finally, this condition the residual current that is need to initiate the trip becomes lower. The main objective of the calibration process is to find a suitable demagnetization or magnetization voltage until we can find the desired residual current that will trip the residual current device [2].

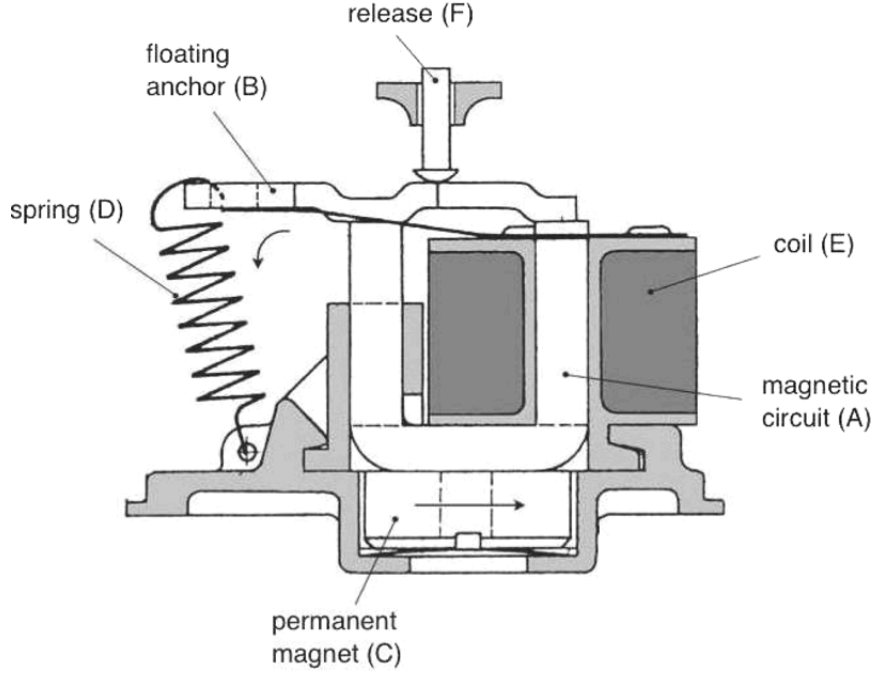


Figure 2.10: Electromechanical Relay

2.6.2 CALIBRATION ALGORITHM

In the previous section we have seen that in order to calibrate the RCCB we need to find the correct demagnetization or magnetization voltage. In order to find the right demagnetization or magnetization voltage, we need to know the relationship between the voltage V_x and residual current I . According to the ABB documentation, the equation that describe this relationship is described below [10]

$$I = I_0 - (e^{AV} - 1) K \quad (2.1)$$

K is a constant value, where it is determined based on the rated residual operating current. While the constant value A is determined according to the combination of the sensing elements such as the transformer, filter, electromechanical relay, and the calibration stage. I_0 is the initial residual current of the RCCB during the first step of the calibration. This mathematical expression is derived using data driven approach. By finding the expression that will best fit the calibration data. When we have defined that constant values and fixed the model, we can use this model to estimate the voltage V_x that will calibrate the RCCB to the desired tripping current.

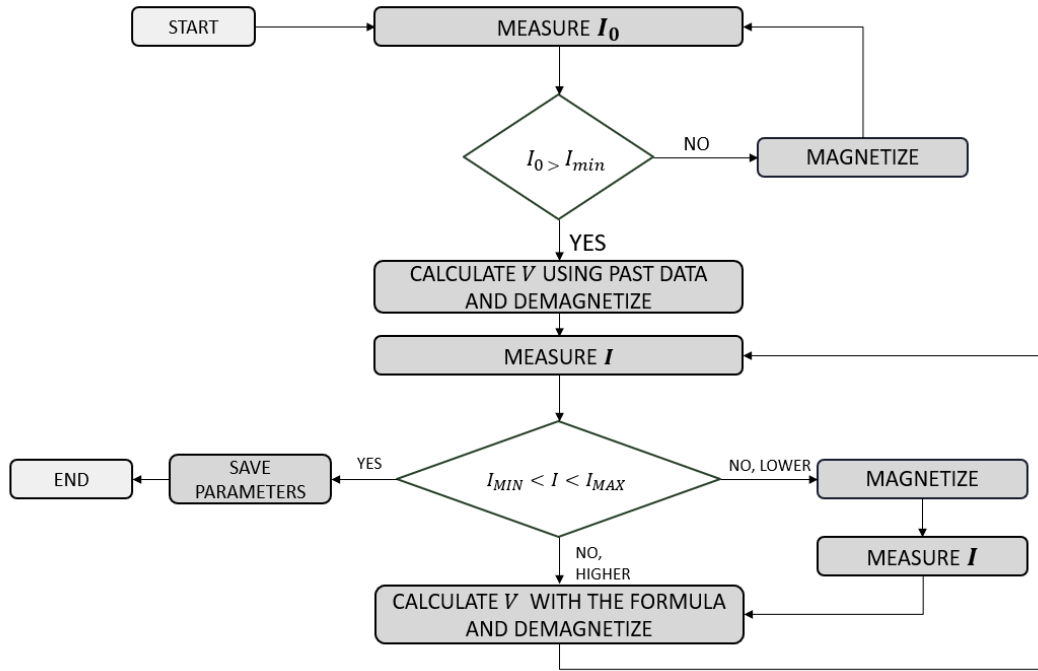


Figure 2.11: RCCB Calibration Algorithm

Figure 2.11 shows the calibration algorithm of the RCCB [11]. The first step of the calibration is to measure the initial residual current I_0 . Then based on the value of this initial residual current I_0 , the algorithm will perform magnetization shot if I_0 is less than the defined minimum current I_{min} . If I_0 is greater than I_{min} demagnetization shot is performed. The value of demagnetization voltage is computed using the mathematical expression 2.1. The constant parameter A and K are calculated according to the past calibration data with similar condition. Next, after demagnetization shot is performed, we measure again the residual current I . If the value residual current I is within the predefined minimum and maximum value. Then the system has found the correct tripping current and therefore the calibration process is completed. If the value of residual current I is higher than the predefined maximum value I_{max} , then the system will perform demagnetization shot again. This cycle will continue until the desired residual current is obtained. If the value of residual current I is lower than the predefined minimum value I_{min} , the the system will perform magnetization shot to increase the residual current I . Then the system will perform demagnetization shot until the desired residual current is obtained.

3

Device Behaviors

In the previous chapter we have seen the calibration process of the RCCB (Residual Current Circuit Breaker) and the technology that are used. In this chapter we are going to identify several behaviors of the RCCB device during the calibration process. Identifying the device behaviours will be useful to design the Digital Twin which will simulate the devices during the calibration.

The device behavior that we are going to identify is device F200 with service code 405030017303. We will identify the behavior based on the calibration data that is obtained from the database.

3.1 OBSERVING THE CALIBRATION DATASET

The following graph in figure 3.1 shows the relationship between residual current and demagnetization voltage of several devices with the same service code: 405030017303.

Service Code	405030017303
Machine Key	AM3116-1-5
Serial Number	001RMG9S
Duration	458990 ms
Result	Good

Table 3.1: Calibration Information of Device 001RMG9S

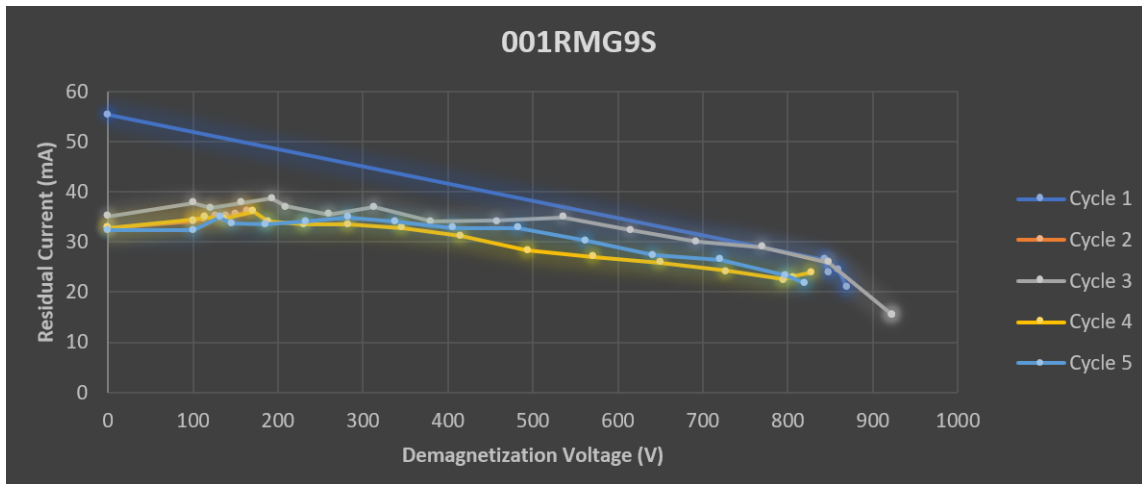


Figure 3.1: Calibration Graph of Device 001RMG9S

In figure 3.1 we can see the calibration process of F200 device with serial number 001RMG9S. This calibration process consists of 5 cycles. In the first cycle the initial current is usually larger than the initial current in the next cycle. In the first cycle the initial current is around 55.3 mA, while the initial current of the second until fifth cycle is around 34 mA.

Service Code	405030017303
Machine Key	AM3116-1-1
Serial Number	001RMG5C
Duration	226351 ms
Result	Good

Table 3.2: Calibration Information of Device 001RMG5C

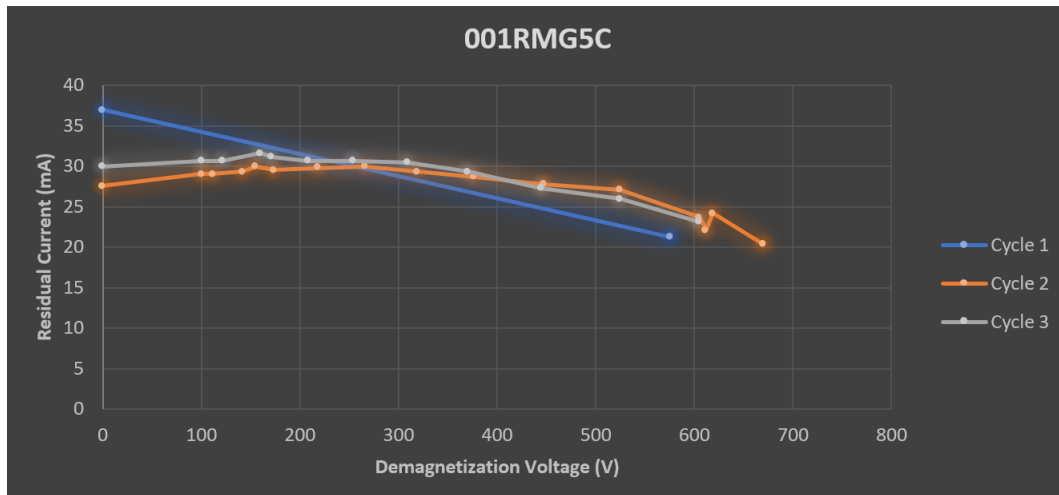


Figure 3.2: Calibration Graph of Device 001RMG5C

In figure 3.2, we can observe that if we apply low demagnetization voltage around 100 – 200 Volts, the residual current (I_x) will increase instead of decreasing.

Service Code	405030017303
Machine Key	AM3116-2-1
Serial Number	001RMG1P
Duration	292459 ms
Result	Good

Table 3.3: Calibration Information of Device 001RMG1P

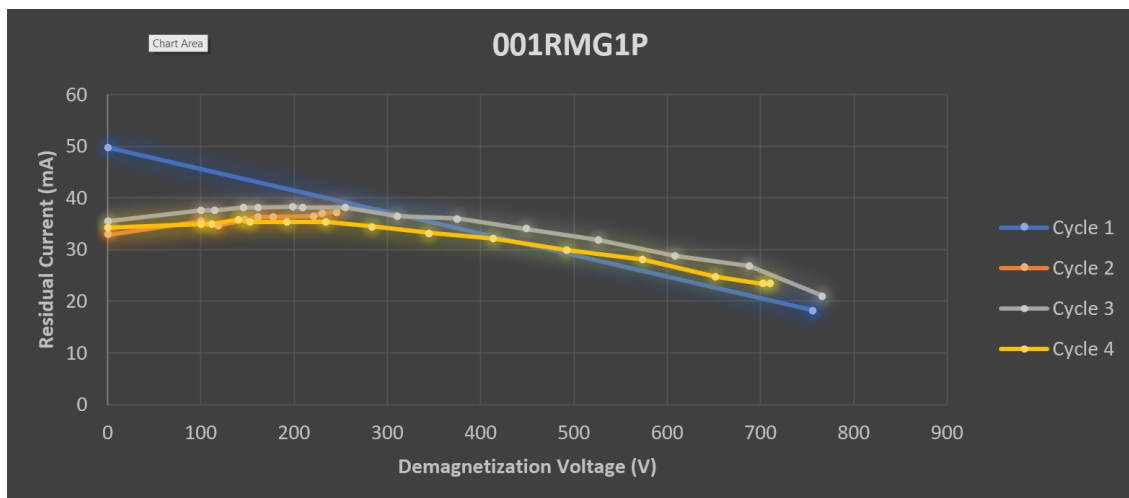


Figure 3.3: Calibration Graph of Device 001RMG1P

Service Code	405030017303
Machine Key	AM3116-2-6
Serial Number	001RMG6Q
Duration	238836 ms
Result	Good

Table 3.4: Calibration Information of Device 001RMG6Q

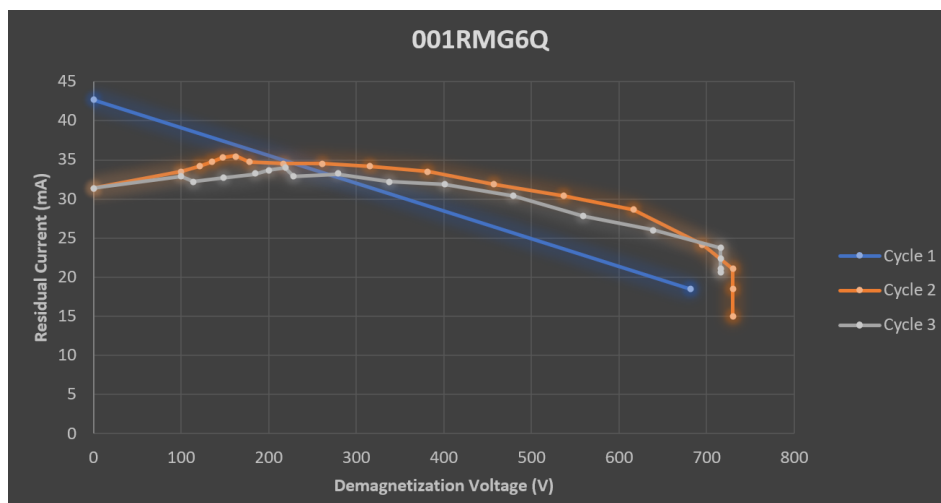


Figure 3.4: Calibration Graph of Device 001RMG6Q

During calibration process the algorithm will perform demagnetization until the device reach the correct residual current. If the calibration overshoots the good range, the algorithm must perform magnetization. This is done in order to bring the residual current of the device back to the initial residual current. However, the magnetization process takes longer time and after magnetization process the algorithm perform calibration slower than before.

In the above figure 3.4, we can observe that after the magnetization process, the calibration process takes a lot of steps to reach the correct residual current. Therefore, we should aim to avoid magnetization process.

Service Code	405030017303
Machine Key	AM3116-1-4
Serial Number	001RMG60
Duration	236108 ms
Result	Good

Table 3.5: Calibration Information of Device 001RMG60

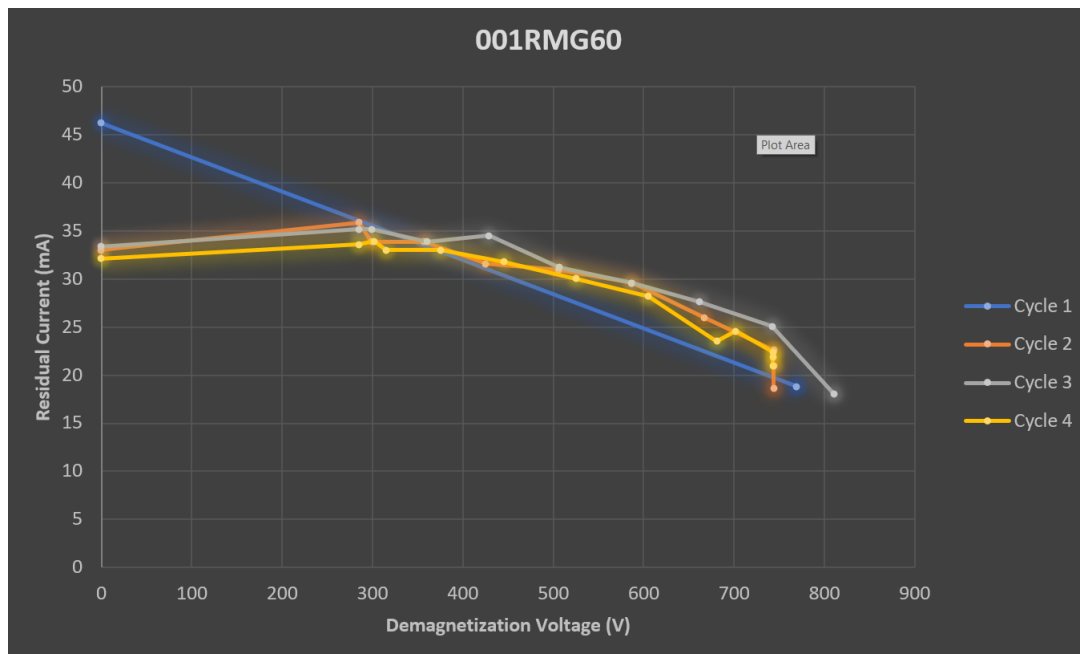


Figure 3.5: Calibration Graph of Device 001RMG60

Service Code	405030017303
Machine Key	AM3116-1-4
Serial Number	001RMFV7
Duration	113579 ms
Result	Good

Table 3.6: Calibration Information of Device 001RMFV

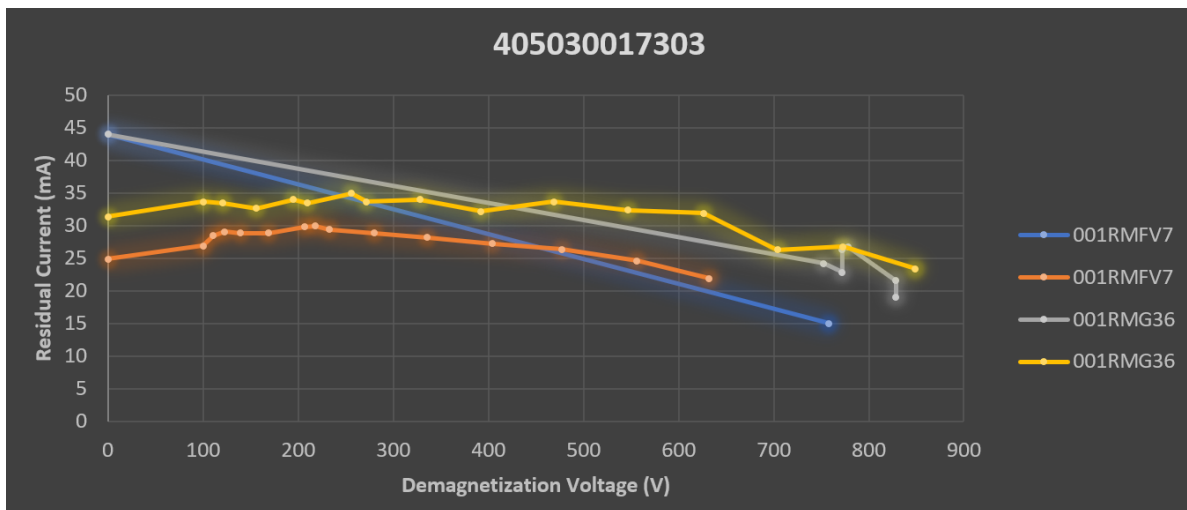


Figure 3.6: Calibration Process of 001RMFV7 and 001RMG36

The above figure 3.6 shows the calibration diagram of two devices with serial number 001RMFV7 and 001RMG36. Both devices have similar initial current during the first cycle of the calibration which is around 44 mA. However, the initial currents at the second cycle are different.

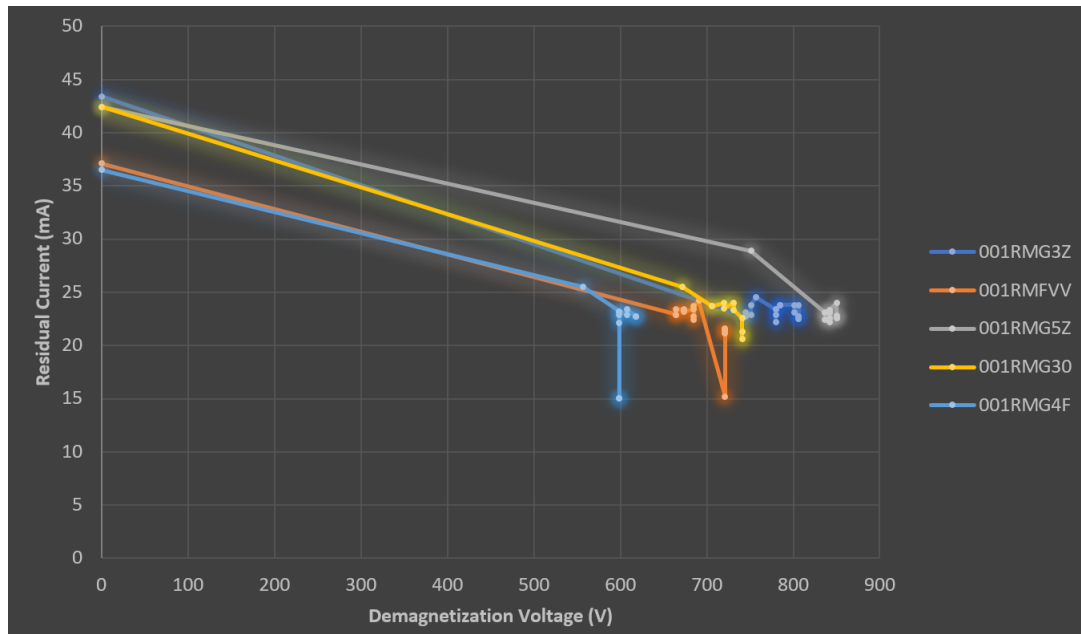


Figure 3.7: Calibration Process of Several Devices

In the above figure 3.7 we can observe that when we only slightly increase the demagnetization voltage, the residual current might increase instead of decreasing.

In the figure 3.7 above we can also observe that when residual current is close to the good range, it can be more challenging to calibrate. As we can see on the diagram above it requires many steps to complete the calibration even though the first shot is close to the good range. This is also due to the next demagnetization process may increase the residual current instead of decreasing it. As a result overall calibration time will increase if the repeated current measurement is unstable.

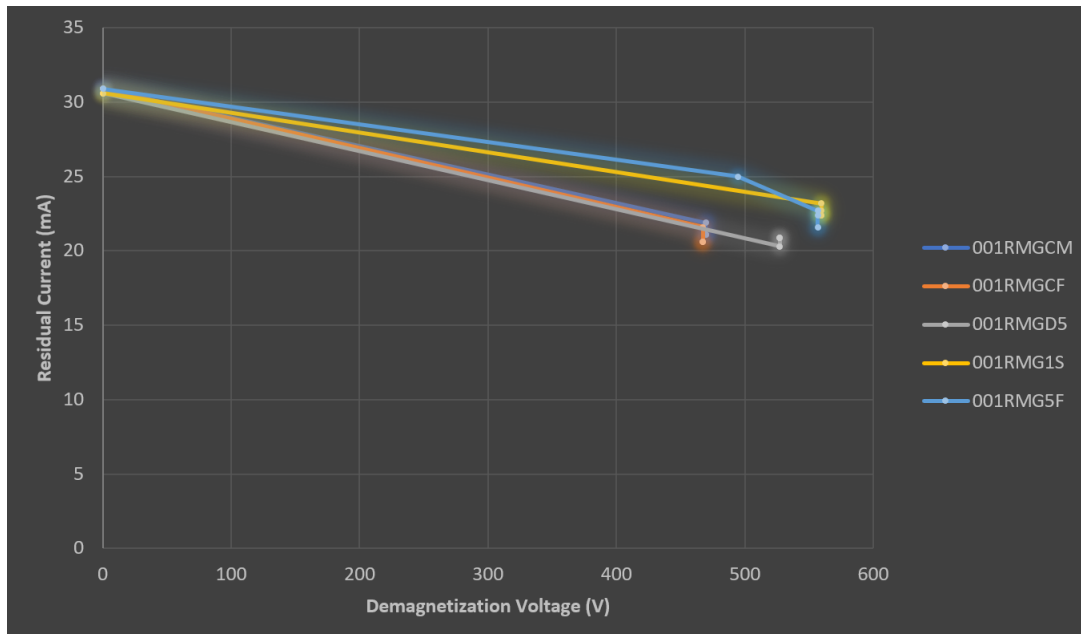


Figure 3.8: Devices with similar initial current

The above figure 3.8 shows the calibration process of several devices with the same service code 405030017303. These devices has similar initial current around 31 mA. Based on this diagram, we can observe that for the similar initial current value, the value of V demagnetization may not be the same.

The figure 3.9 below shows the repeated current measurement of several devices with the same service code 405030017303. Based on this diagram we can observe that the next repeated current measurement might increase instead of decreasing.

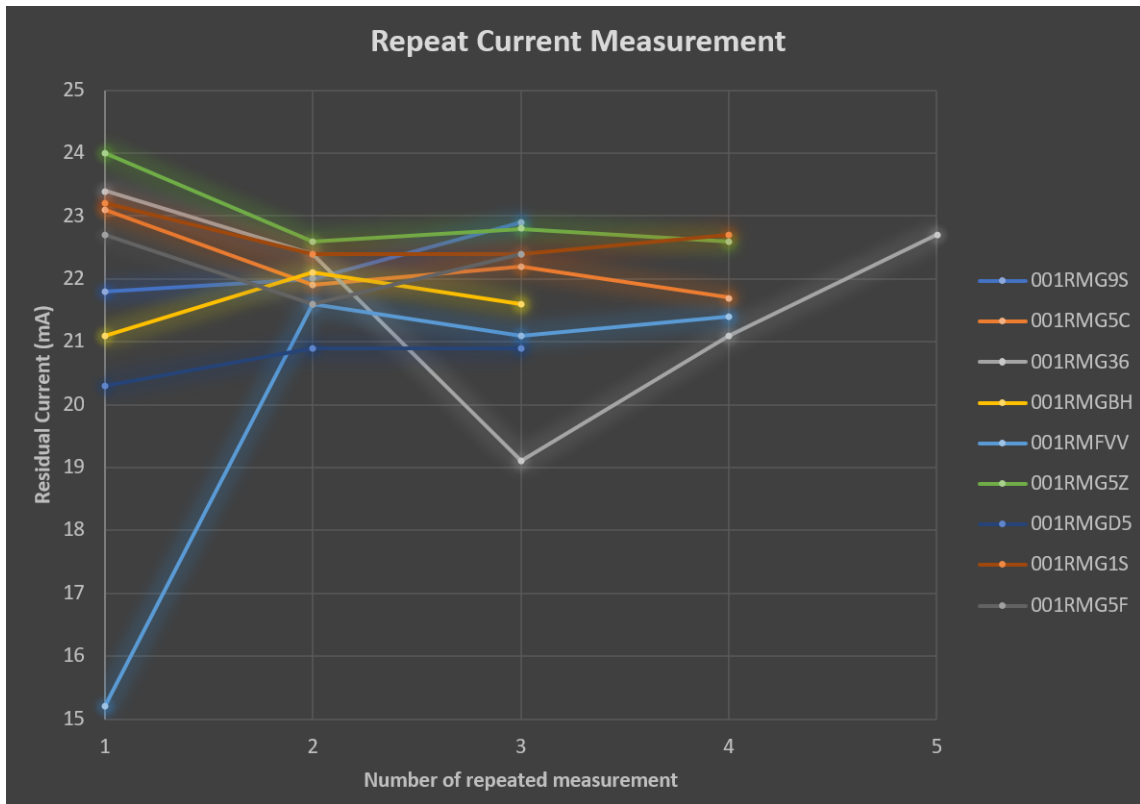


Figure 3.9: Devices with repeated current measurement

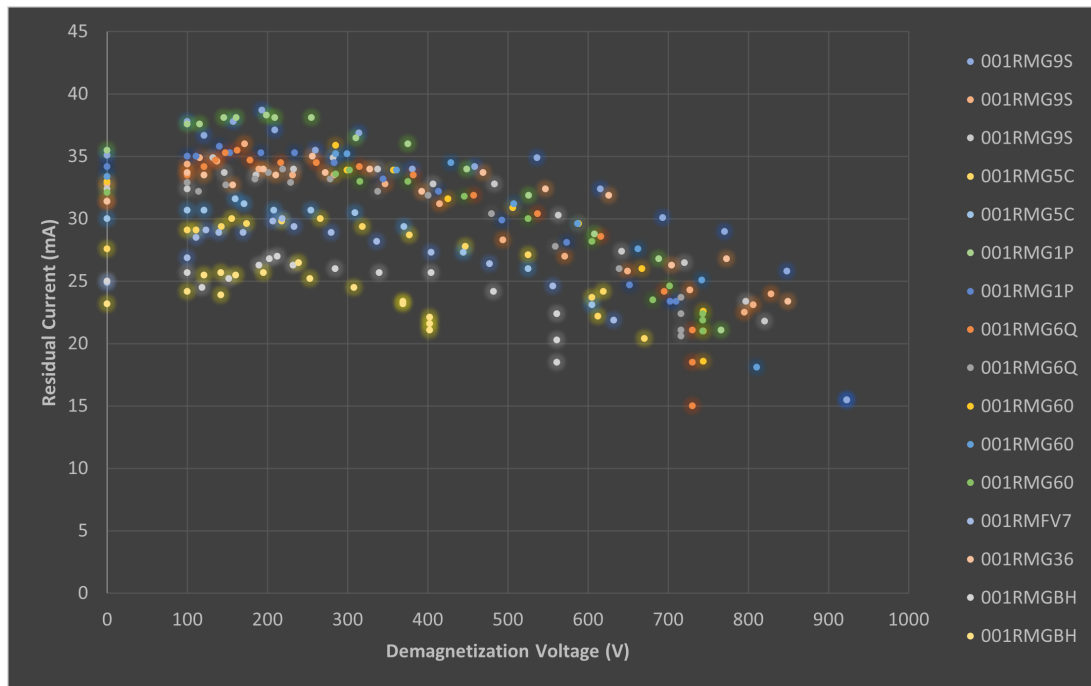


Figure 3.10: Calibration process of multiple devices with service code 405030017303

The figure 3.10 above shows the relationship between the residual current and demagnetization voltage of several devices with the same service code: 405030017303.

Based on several diagrams that we have seen above, we can conclude several behaviors of the device as the followings:

1. After the first magnetization, the new initial current is most likely lower than the original initial current
2. Applying V demagnetization between 0V – 200V will increase the residual current instead of decreasing it.
3. If the device is not rearmed, the current measurement will be equal to the minimum I ramp.
4. The algorithm become slow after it encounter magnetization. As a result, the total calibration time will increase.
5. When we only slightly increase the demagnetization voltage, the residual current might increase instead of decreasing.

6. The value of residual current normally will decrease after repeated current measurements. However, there is a percentage of probability that the value of residual current will increase after repeated current measurements.

3.2 INACCURACY OF CURRENT MEASUREMENT

Optimizing The I ramp I ramp is a current generated by the Basetta. The purpose of this current is to trip the device (RCCB). I ramp has a starting and ending current. These values are set by the Quality Control team. For example at the beginning the I ramp will start at 15mA and will end at 80mA. At the moment, the time taken from the starting current until it reaches the ending current is fixed to 5 seconds.

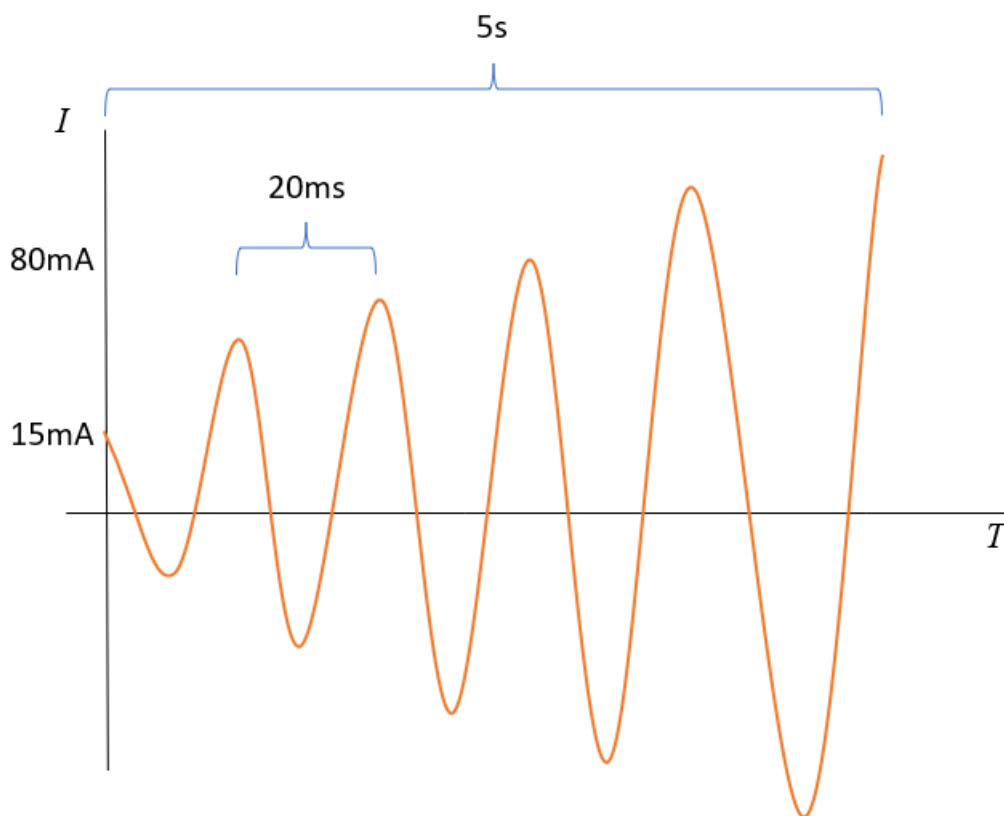


Figure 3.11: Ramp current

In the future we can dynamically adjust the time required from the starting current until it reaches the ending current. This is because, at the beginning of the calibration process, we do not need high accuracy to measure the initial current I_o , because the value of I_o is big. Therefore, we can increase the I ramp quickly. As the calibration process approach the good range of the $I\Delta$, then we can slow down the increment of the I ramp. Therefore, we will get a more accurate measurement of $I\Delta$.

The generated I ramp has frequency of 50Hz, which means it took 20ms to complete one cycle.

$$ratio = \frac{I_{end} - I_{start}}{time} = \frac{80mA - 15mA}{5s}$$

$$13 \frac{mA}{s} = 0.013 \frac{mA}{ms}$$

In one cycle

$$20ms \times 0.013 \frac{mA}{ms} = 0.26mA$$

The device requires at least three cycles to trip

$$3 \times 0.26mA = 0.78mA$$

4

Digital Twin Model

4.1 DIGITAL TWIN IMPLEMENTATION

The main purpose of the Digital Twin is to simulate the behavior of the RCCB device during the calibration process. Therefore, we can safely adjust the calibration parameters and predict the calibration results using the Digital Twin, without directly risking the production process. If we adjust the calibration parameters directly to the production line, it will be very risky if the result is not as we expected.

In this section we are going to take a look at the implementation of the Digital Twin Application. We will look at the design pattern of the Digital Twin. Then we will look at several models that we will use for predicting the residual current values.

4.1.1 DIGITAL TWIN DESIGN

Before we design the digital twin application, we will look at the calibration process in the real system. The figure 2.1 below shows the calibration process in the real systems. The first step is the SCADA will ask to the ABB.algorithm what is the action that needs to be carried out. Next the ABB.algorithm will give the action that should be carried out. The SCADA will then give instruction to the Machine based on the action that is given by the ABB.algorithm. These actions could be rearm, current measurement, demagnetization, and magnetization. Then the machine will carry this command to the Device. The machine will then give feedback to the

SCADA based on the command executed on the Device. The SCADA will then return the feedback to the ABB.algorithm to decide what is the next action. Then the loop will start again from the first step until the calibration is completed.

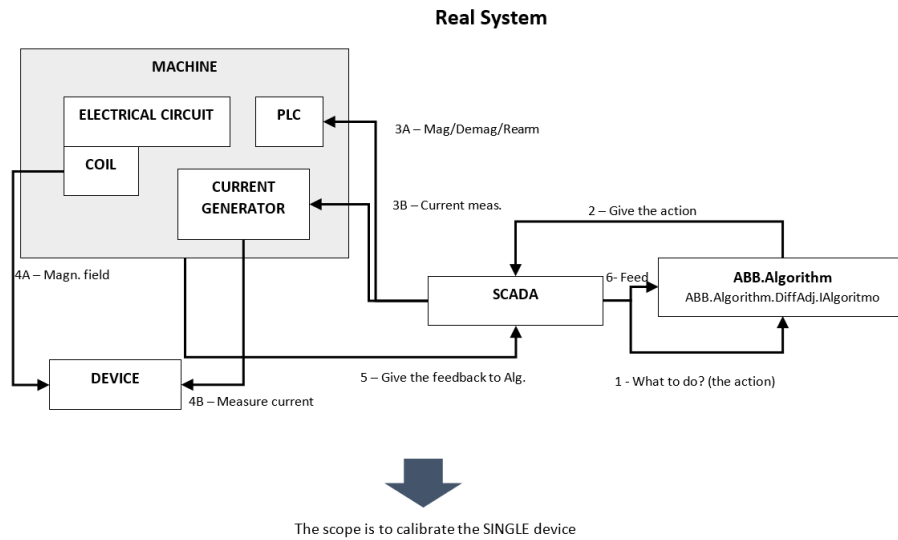


Figure 4.1: Calibration process in the real system

The real system can be represented as several objects in Digital Twin application. The figure 2.2 below shows the calibration process inside Digital Twin application. Basically, the process is similar with what is happening in the Real System. The difference is in the Digital Twin application, all the processes are happening inside a software.

The SCADA is represented by a Test Manager object. The Calibration Machine is represented with Machine Digital Twin object. The real RCCB Device is represented with Device Digital Twin object. The first step is the Test Manager will request the next action to the ABB.algorithm. The ABB.algorithm will then give the action to the Test Manager. Then the Test Manager will convey the action to the Machine Digital Twin. Next, the Machine Digital Twin will perform the action onto the Device Digital Twin. The Device Digital Twin will then give the feedback to the Machine Digital Twin and forward it to the Test Manager. Finally, The Test Manager will give the feedback to the ABB.Algorithm to decide the next action. This loop process will continue until the target calibration status is achieved.

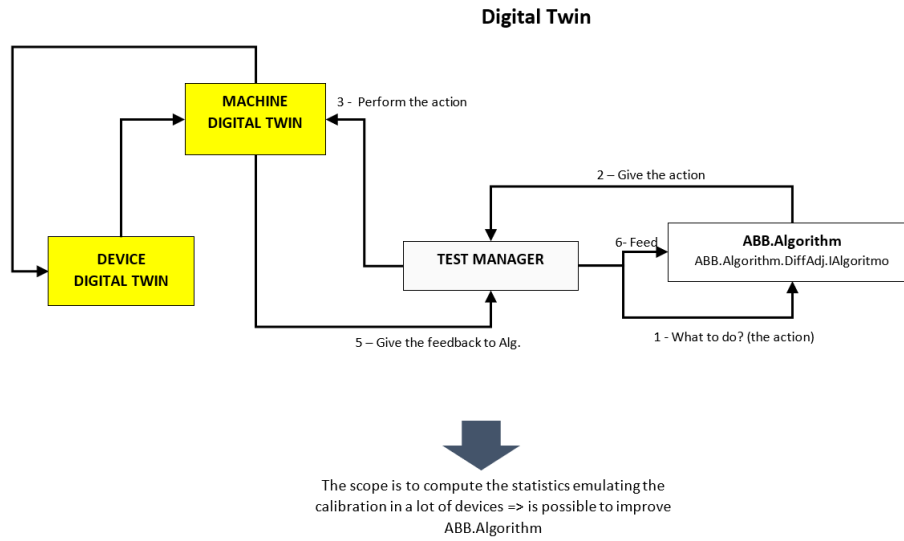


Figure 4.2: Calibration process in the digital twin

4.1.2 CALIBRATION DATASET

In the previous section we have define the design pattern of the Digital Twin. In this section we will look at the dataset which we will use to train a Model for predicting the calibration behavior.

The dataset that we will use is a calibration data that is extracted from the database with service code 405030017303. The table below shows the detail information about the calibration data. The calibration data was taken on 16 June 2022. There are around 362 pieces of devices. The average calibration time is around 45 seconds.

Service Code	405030017303
Number of pieces	362 pieces
Average Time	44906 ms

Table 4.1: Calibration Information of Device 001RMG9S

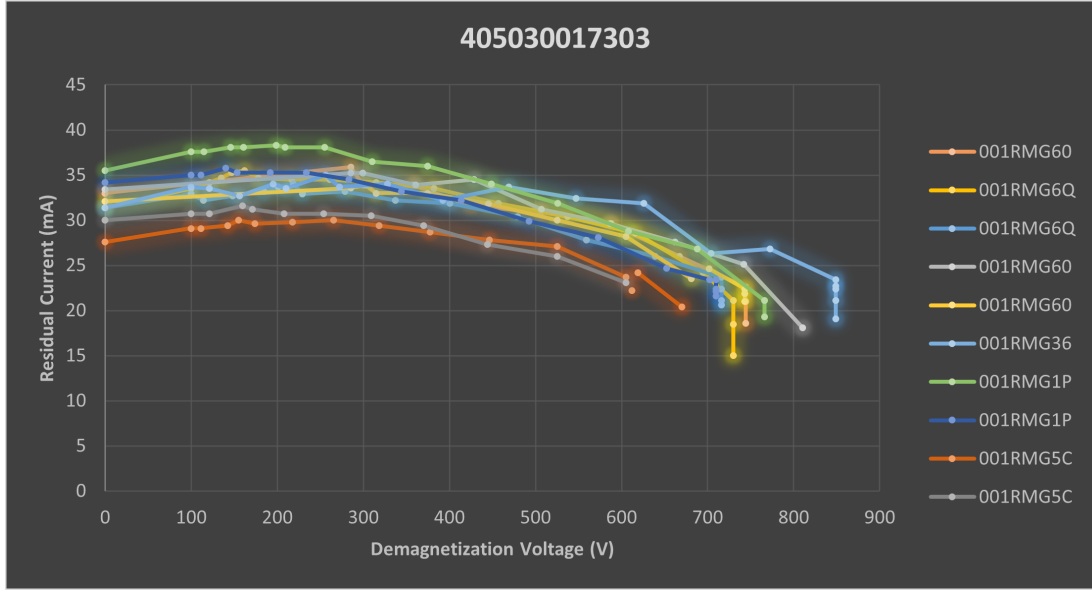


Figure 4.3: Calibration process of several devices with different serial number

The 4.3 above shows the plot of the calibration data from several devices with different serial number. We will then use this dataset to create several models for predicting the calibration behavior.

4.1.3 PREDICTION MODEL

In the previous sections we have seen the Digital Twin Implementation and the calibration dataset. In this section we will see how the Model that will be used to simulate the behavior of the RCCB device during calibration process.

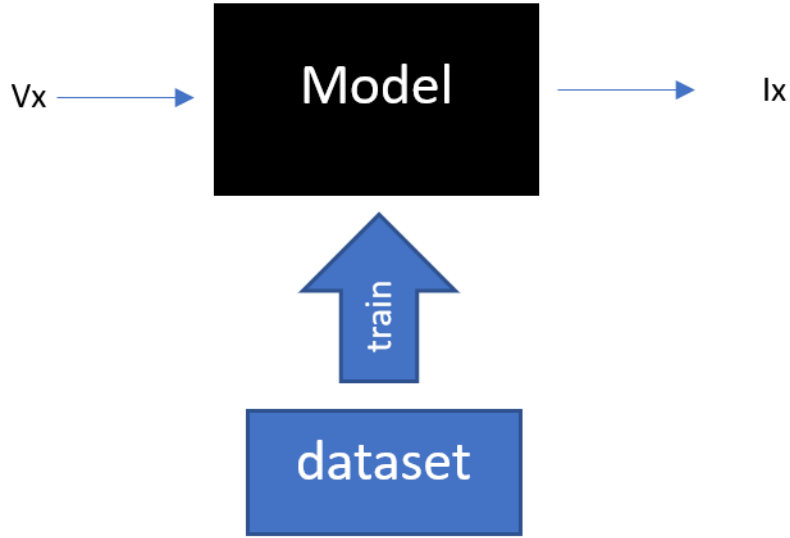


Figure 4.4: Model for predicting the residual current

We will create a Model for predicting the value of residual current (I_x) based on the value of demagnetization voltage (V_x). Therefore, the input to the Model will be V_x and the output is I_x . We will then use the dataset to train the Model.

There are several Model that we can use to predict the value of I_x . The Model that we will use in this project are Exponential Model, Polynomial Model, MLNET Model, and Neural Network Model. The implementation of these Models will be explained in the next following chapters.

4.2 DIGITAL TWIN EXPONENTIAL MODEL

In the section 4.1 we have seen how the Digital Twin program is design and implemented in .NET framework using C#. In next following sections we are going to see the model that will be used to predict the behavior of the RCCB calibration. In this chapter we will use the exponential model to simulate the behavior of the RCCB calibration. The formula that we will use is as follow.

$$I = I_0 - (e^{AV} - 1) K$$

4.2.1 FINDING PARAMETER FOR THE EXPONENTIAL MODEL

We will train the model using the calibration dataset that is mentioned in the previous chapter. We need to find the correct parameter that will best fit the calibration data.

In order to find the parameter for the exponential model, we are going to use MATLAB. Below is the result of curve fitting using MATLAB.

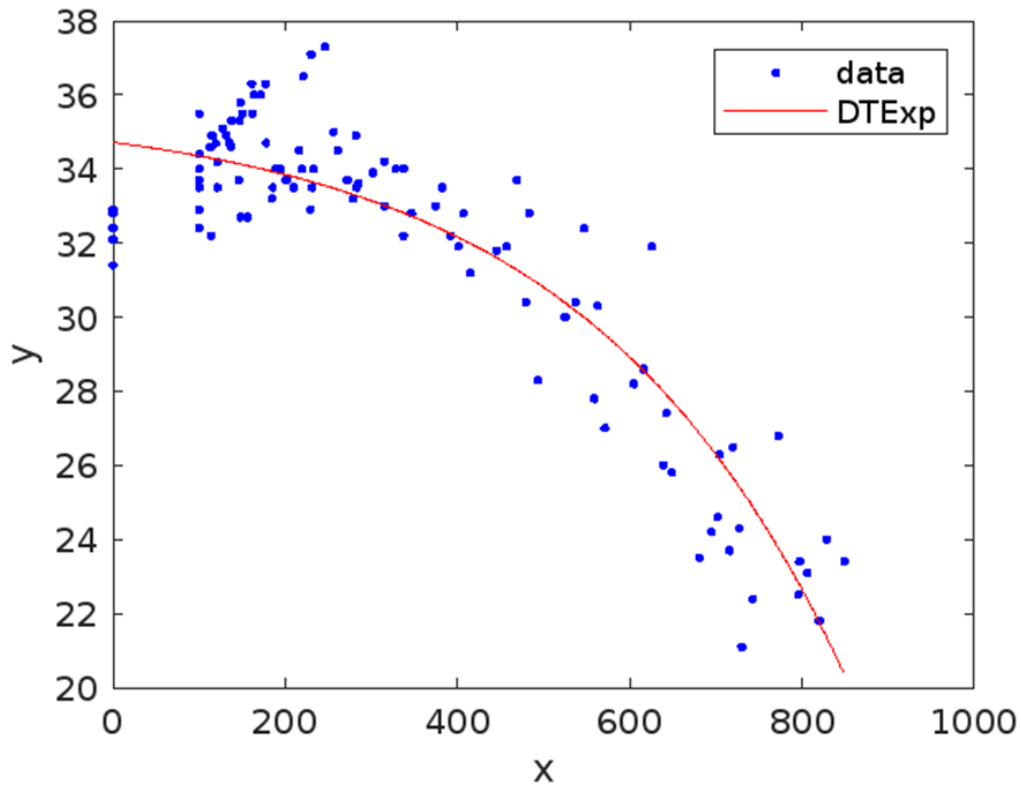


Figure 4.5: Curve fitting of the exponential model

Result of curve fitting using MATLAB

DTExp =

General model:

$DTExp(x) = (a - (\exp(b \cdot x) - 1) \cdot c)$

Coefficients (with 95% confidence bounds):

a =	34.72	(34.12, 35.31)
b =	0.003291	(0.002319, 0.004263)
c =	0.9352	(0.1529, 1.717)

There are three parameters that we use a, b, and c. According to the exponential formula, a corresponds to I_0 , b corresponds to A, and c correspond to K. Therefore $I_0 = 34.72$, $A = 0.003291$, $K = 0.9352$

4.2.2 EVALUATION RESULT OF EXPONENTIAL MODEL

Here we run the Digital Twin simulation based on the followings parameter to simulate device with service code: 405030017303.

Device parameter

- Probability of successful rearm is 98%
- Initial current = 34.72 mA (with 5% variance)
- a avg = 0.003291
- k avg = 0.9352

Ramp parameter

- Number of cycles to trip = 3 cycles
- Period of the ramp wave = 20 ms
- I start = 15 mA
- I end = 80 mA
- I ramp time = 5 s

Algorithm parameter

- I residual = 30 mA
- I buono max = 23.5 mA
- I buono min = 21.5 mA
- V magnetization = 950 mA

The diagram in figure 4.6 below simulates the behavior of the device based on the exponential model of the digital twin. We can compare the digital twin graph with the graph of real calibration data in figure 3.3. According to these graphs we can see that the graph of the real calibration data in figure 3.3, the residual current is increasing at the beginning of the calibration between 0V to 300V. While the graph of the digital twin simulation on figure 4.6, the residual current is generally decreasing as the demagnetization voltage increases.

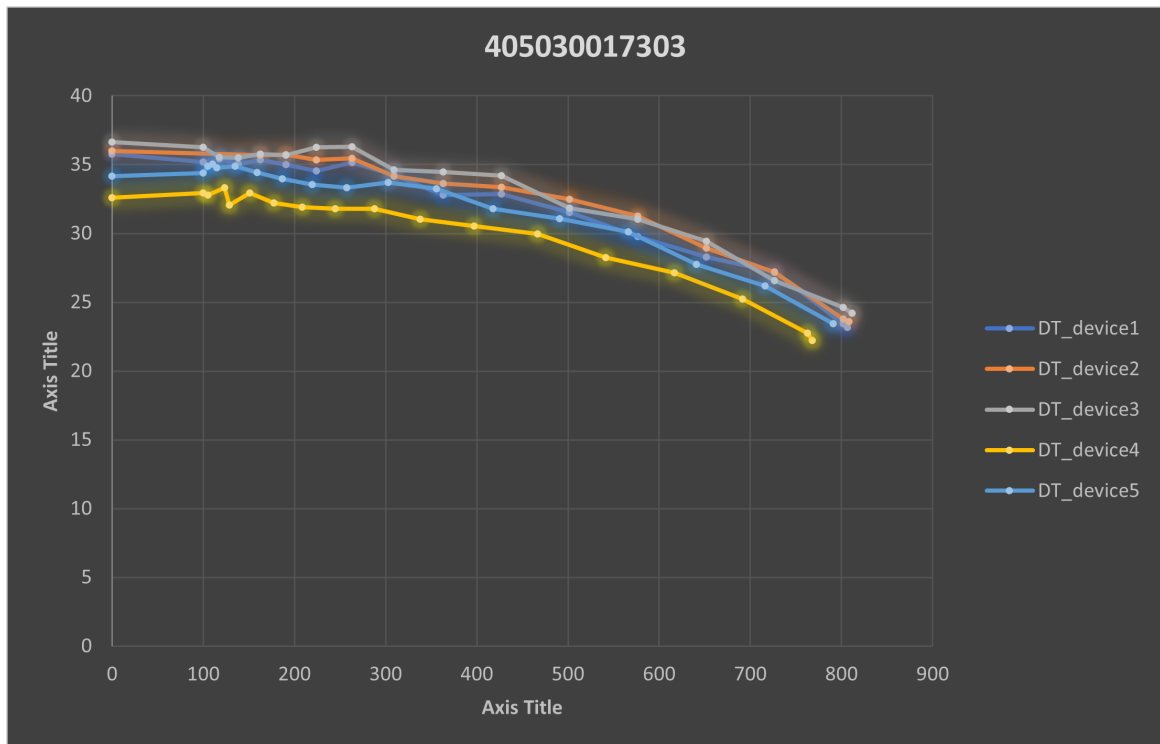


Figure 4.6: Plot of the digital twin result using exponential model

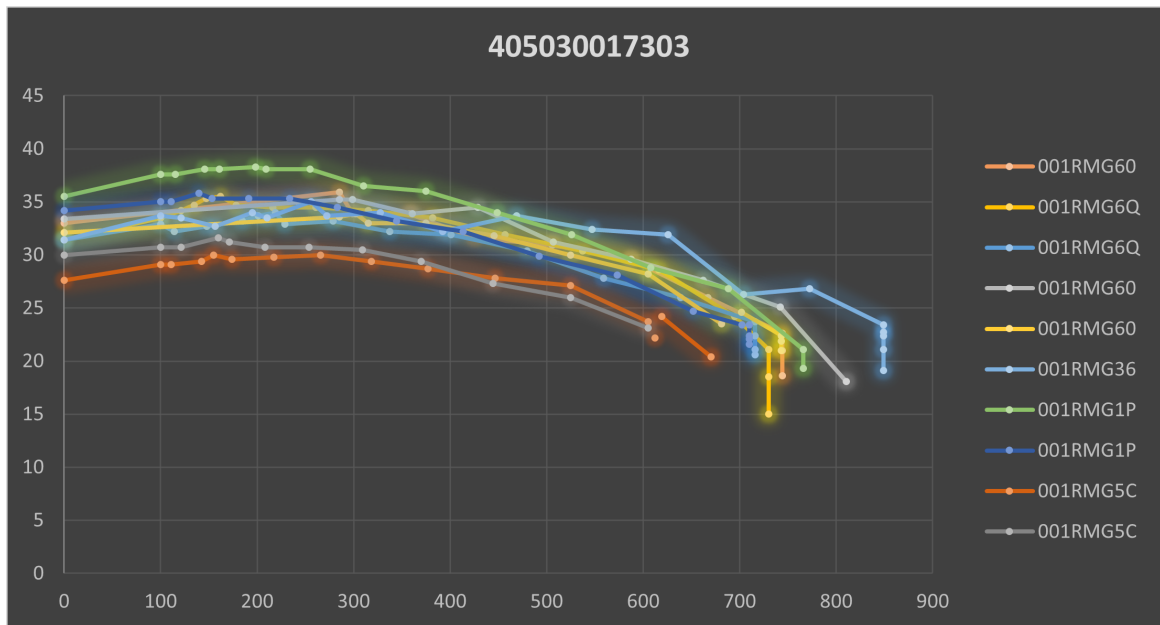


Figure 4.7: Plot of the real calibration data of service code 405030017303

Below data show the summary of the digital twin simulation. According to this data, we can see that the average calibration time is around 44,5 seconds.

First Trial

- Buoni:362
- Scarti:0
- Smagnetizza:1207
- Riarma:2980
- MisuraCorrente:2980
- Magnetizza:90
- Tempo di taratura stimato (sec):15.991,0
- Tempo di taratura medio (sec):44,2

Second Trial

- Buoni:362

- Scarti:0
- Smagnetizza:1213
- Riarma:2964
- MisuraCorrente:2964
- Magnetizza:84
- Tempo di taratura stimato (sec):15.915,0
- Tempo di taratura medio (sec):44,0

Third Trial

- Buoni:362
- Scarti:0
- Smagnetizza:1239
- Riarma:2976
- Magnetizza:101
- Tempo di taratura stimato (sec):16.126,0
- Tempo di taratura medio (sec):44,5

4.3 DIGITAL TWIN POLYNOMIAL MODEL

In the previous section we have seen the Exponential Model for predicting the RCCB calibration. In this section we will see the implementation of Polynomial Model for predicting the value of residual current I_x . In order to find the parameter for the polynomial model, we are going to use the real calibration data with the following information.

Service Code	405030017303
Number of pieces	362 pieces
Average Time	44906 ms
Result	Good

Table 4.2: Calibration Information of Device 001RMG9S

Then we performed curve fitting on these data using MATLAB to find the parameters for the polynomial models. We are going to compare several polynomial models and choose the suitable model for our Digital Twin application.

4.3.1 POLYNOMIAL MODEL DEGREE 5

In this section we will use polynomial with degree 5 as a Model to simulate the RCCB calibration process. Below figure 4.8 is the result of curve fitting using MATLAB and the calibration dataset.

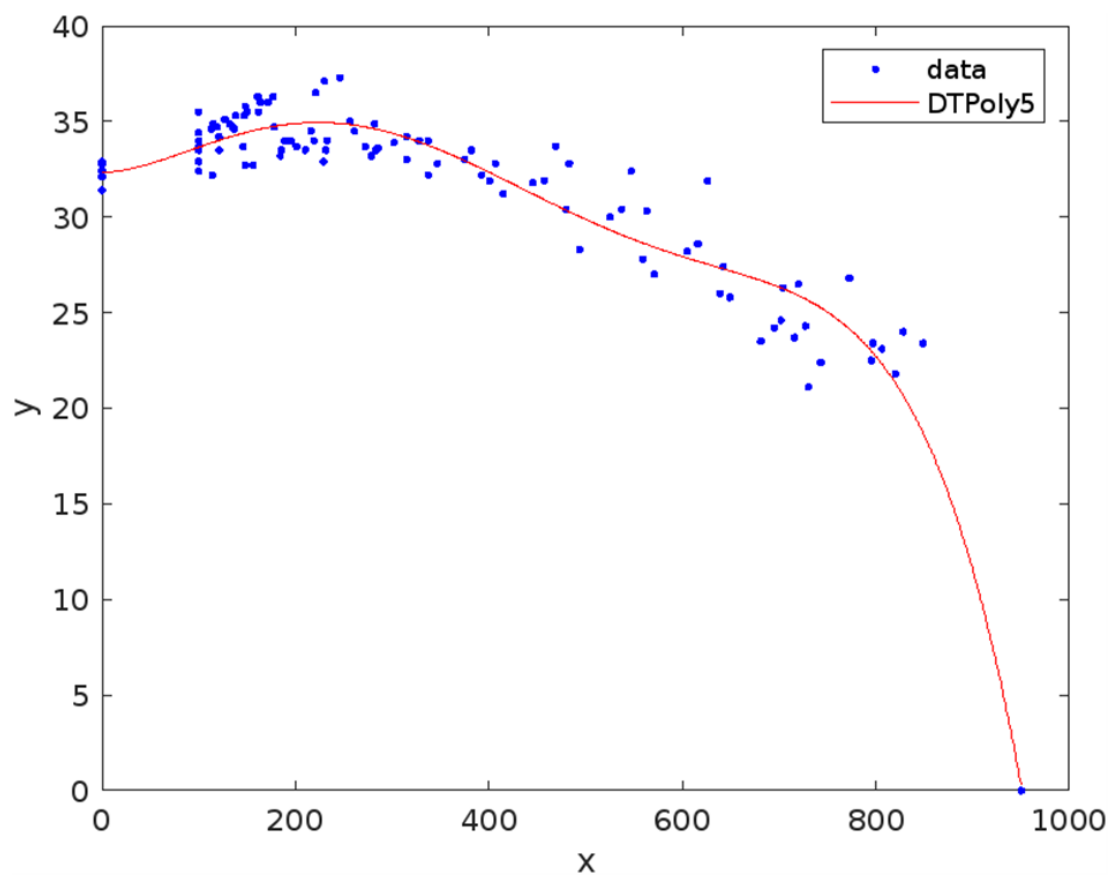


Figure 4.8: Curve Fitting of Polynomial Degree 5

Result of curve fitting using Matlab:

```
DTPoly5 =
Linear model Poly5:
DTPoly5(x) = p1*x^5 + p2*x^4 + p3*x^3 + p4*x^2 + p5*x + p6
Coefficients (with 95% confidence bounds):
p1 = -8.437e-13 (-1.114e-12, -5.739e-13)
p2 = 1.631e-09 (1.006e-09, 2.256e-09)
p3 = -1.046e-06 (-1.554e-06, -5.369e-07)
p4 = 0.0002083 (3.555e-05, 0.000381)
p5 = 0.00128 (-0.02134, 0.0239)
p6 = 32.33 (31.29, 33.36)

gof5 = struct with fields:
    sse: 241.6488
    rsquare: 0.9736
    dfe: 108
    adjrsquare: 0.9724
    rmse: 1.4958
```

4.3.2 POLYNOMIAL DEGREE 8

In this section we will use Polynomial degree 8 as the Model for predicting the value of the residual current I_x based on the value of the demagnetization voltage V_x .

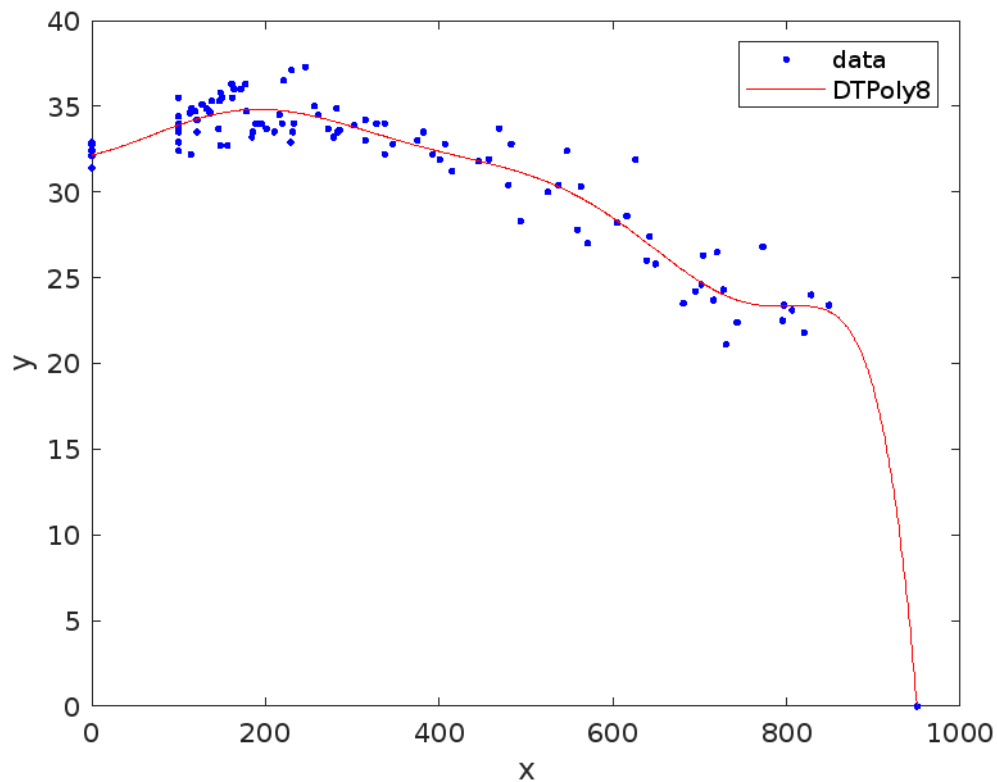


Figure 4.9: Curve Fitting of Polynomial Degree 8

The graph of Polynomial 8 in figure 4.9 fits better the dataset compare to the graph of Polynomial 5 in figure 4.8 The Polynomial degree 8 has lower error compare to the Polynomial degree 5.

Result of curve fitting using Matlab:

```

DTPoly8 =
Linear model Poly8:
DTPoly8(x) = p1*x^8 + p2*x^7 + p3*x^6 + p4*x^5 +
p5*x^4 + p6*x^3 + p7*x^2 + p8*x + p9
Coefficients (with 95% confidence bounds):
p1 = -1.86e-20 (-4.978e-20, 1.258e-20)
p2 = 5.58e-17 (-6.005e-17, 1.717e-16)
p3 = -6.434e-14 (-2.404e-13, 1.117e-13)
p4 = 3.506e-11 (-1.058e-10, 1.759e-10)
p5 = -8.554e-09 (-7.211e-08, 5.5e-08)
p6 = 4.805e-07 (-1.553e-05, 1.65e-05)
p7 = 5.307e-05 (-0.002012, 0.002118)
p8 = 0.0131 (-0.09187, 0.1181)
p9 = 32.14 (31.26, 33.03)

gof8 = struct with fields:
    sse: 166.2842
    rsquare: 0.9818
    dfe: 105
    adjrsquare: 0.9804
    rmse: 1.2584

```

4.3.3 EVALUATION RESULT OF POLYNOMIAL DEGREE 8 MODEL

Device Parameter:

- Probability of successful rearm is 96
- Probability that the next repeat current measurement will decrease = 92
- $p_1 = -1.86e-20$
- $p_2 = 5.58e-17$
- $p_3 = -6.434e-14$
- $p_4 = 3.506e-11$
- $p_5 = -8.554e-09$
- $p_6 = 4.805e-07$
- $p_7 = 5.307e-05$
- $p_8 = 0.0131$
- $p_9 = 32.14$

Ramp parameter

- Number of cycles to trip = 3 cycles

- Period of the ramp wave = 20 ms
- $I_{start} = 15 \text{ mA}$
- $I_{end} = 80 \text{ mA}$
- $I_{ramp \text{ time}} = 5 \text{ s}$

Algorithm parameter

- $I_{residual} = 30 \text{ mA}$
- $I_{buono \text{ max}} = 23.5 \text{ mA}$
- $I_{buono \text{ min}} = 21.5 \text{ mA}$
- $V_{magnetization} = 950 \text{ mA}$

Below graph in figure 4.10 shows the results of the digital twin simulation using the polynomial 8 model. Based on the graph, we can see that the residual current is increasing between demagnetization voltage 0V until 200V. This behavior represents the real behavior in the real calibration data in figure 4.11

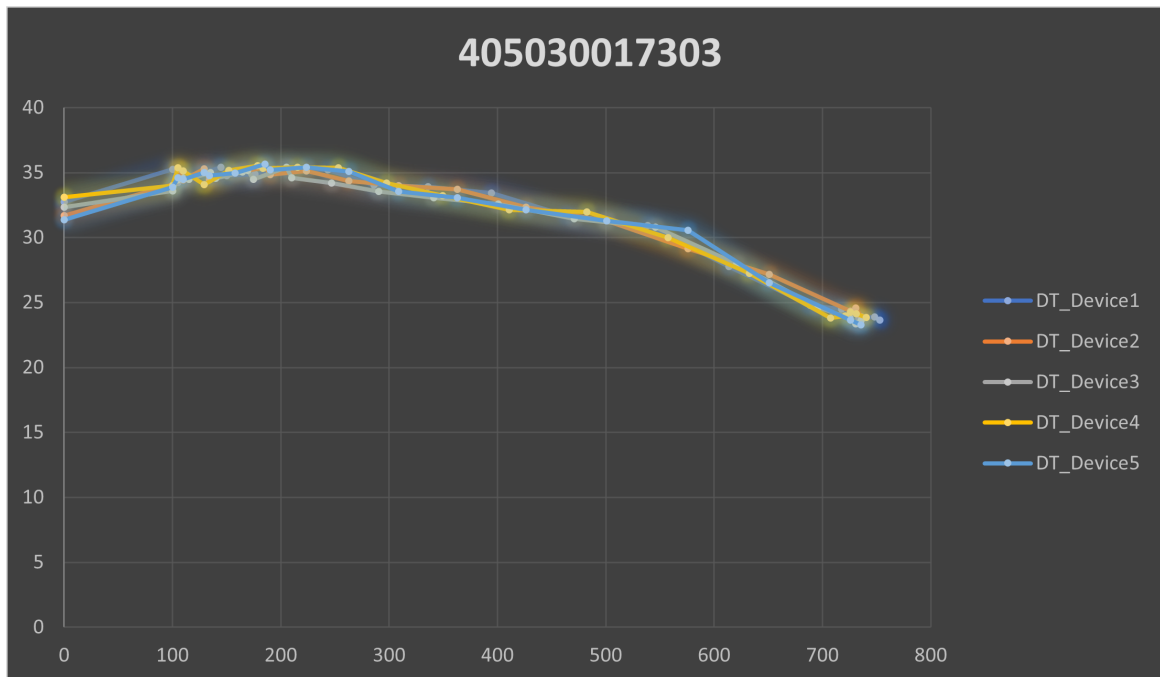


Figure 4.10: Plot of the digital twin result using polynomial 8 model

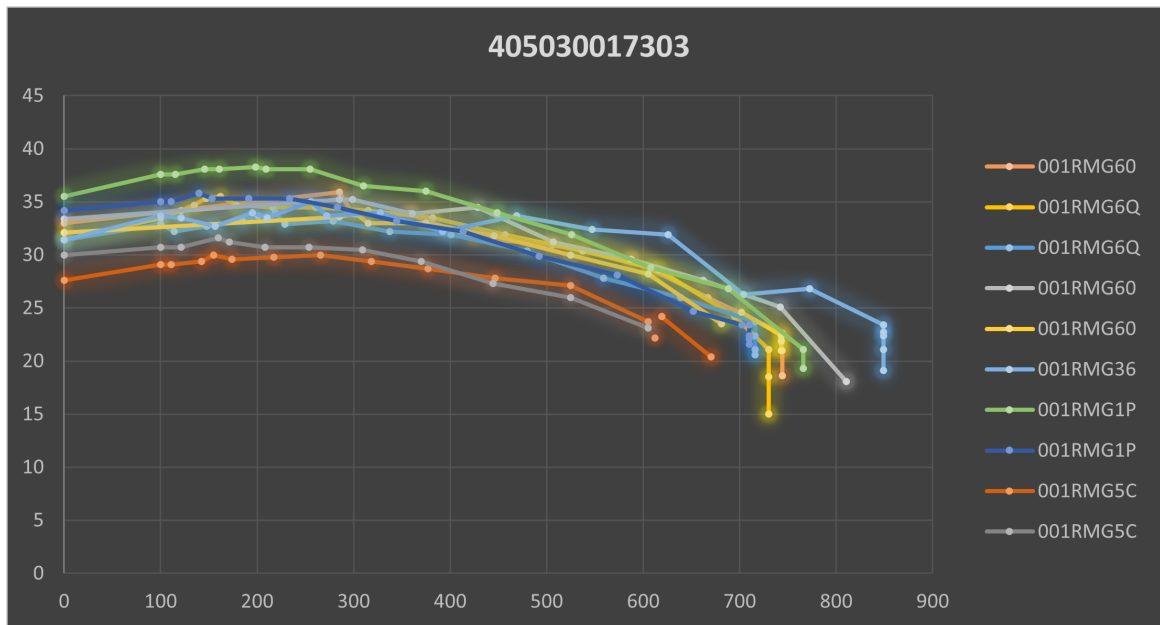


Figure 4.11: Plot of the calibration data of service code: 405030017303

The data below shows several calibration summary of the digital twin simulation. As we can see from the summary, the average calibration time is around 45 seconds.

First Trial:

- Buoni:362
- Scarti:0
- Smagnetizza:1351
- Riarma:3042
- MisuraCorrente:3042
- Magnetizza:54
- Tempo di taratura stimato (sec):16.491,0
- Tempo di taratura medio (sec):45,6

Second Trial:

- Buoni:362

- Smagnetizza:1269
- Riarma:2797
- MisuraCorrente:2797
- Magnetizza:63
- Tempo di taratura stimato (sec):15.310,0
- Tempo di taratura medio (sec):42,3

Third Trial

- Buoni:362
- Scarti:0
- Smagnetizza:1352
- Riarma:2985
- MisuraCorrente:2985
- Magnetizza:71
- Tempo di taratura stimato (sec):16.351,0
- Tempo di taratura medio (sec):45,2

4.4 DIGITAL TWIN ML.NET MODEL

In this section we will use ML.NET to create a Model for predicting the behavior of the RCCB calibration process.

ML.NET is an open source and cross-platform machine learning framework developed by Microsoft. In this project we will use the Model Builder to build the machine learning model.

4.4.1 CREATING THE ML MODEL

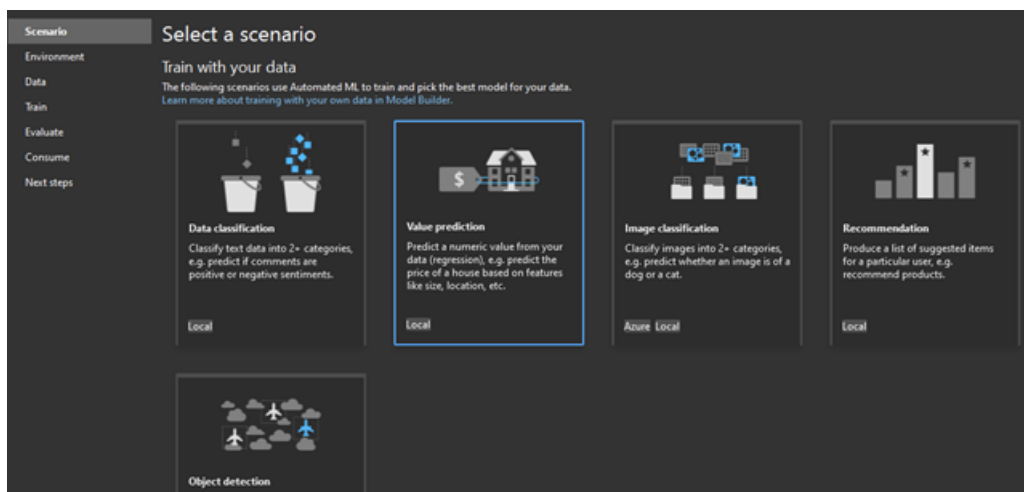
In this section we are going to analyze the performance of ML .NET for predicting the residual current. The machine learning model will be trained using real dataset with below information.

Service Code	405030017303
Number of pieces	362 pieces
Average Time	44906 ms

Table 4.3: Calibration Information of Device 001RMG9S

Below are the steps of using the Model Builder to create the machine learning model.

1. First, we choose value prediction for the scenario, because we are going to predict the value of the residual current I_x based on the value of demagnetization voltage V_x . Therefore V_x is our training feature and I_x is our training label.



2. Then we choose the local CPU for the environment.

Scenario

Environment

Data

Train

Evaluate

Consume

Next steps

Add data

In order to build a model, you must add data and choose your column to predict.
[How do I get sample datasets and learn more?](#)

Input

Data source type

☒ File (.csv, .tsv, .txt)

☐ SQL Server

C:\Users\ITWIDHA\OneDrive - ABB\Documents\ABB Project\Thesis\Wishnu Project Backup\DTModel3.csv [Browse...](#)

Column to predict (Label): [?](#)

Current

[Advanced data options...](#)

Data Preview

10 of 114 rows, and all 2 columns (including 0 columns that are ignored).

Voltage	Current
0	32,8
100	34
113	34,6
127	35,1
138	35,3
150	35,5
164	36
0	32,8
100	34,4
115	34,9

[Next step](#)

- Next, we upload the training dataset in csv format. The demagnetization voltage V_x will be the input to the machine learning model. The residual current I_x will be the output of the machine learning model.

Scenario
Environment
Data
Train
Evaluate
Consume
Next steps

Add data

In order to build a model, you must add data and choose your column to predict. [How do I get sample datasets and learn more?](#)

Input

Data source type

☒ File (.csv, .txt, .txt)
☐ SQL Server

C:\Users\ITWID\OneDrive - ABD\Documents\ABD Project\Thesis\Wishnu Project Backup\DTModel3.csv [Browse...](#)

Column to predict (Label):

[Advanced data options...](#)

Data Preview

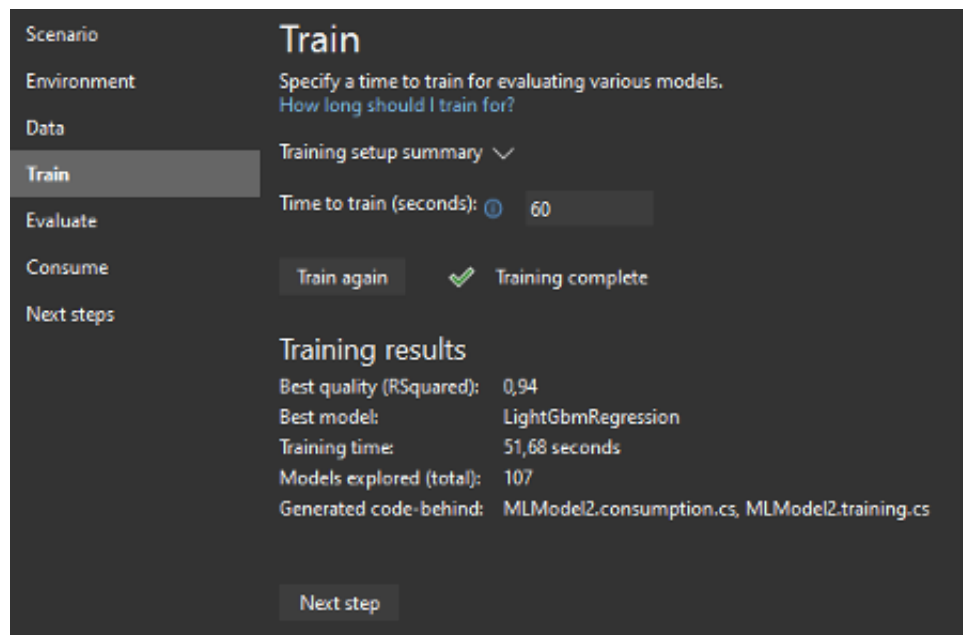
10 of 114 rows, and all 2 columns (including 0 columns that are ignored).

Voltage	Current
0	32,8
100	34
113	34,6
127	35,1
138	35,3
150	35,5
164	36
0	32,8
100	34,4
113	34,9

[Next step](#)

- Then, we train the data for 10 seconds. According to the guideline from Microsoft, the training time will be based on how big the data is. Below are the guideline table from Microsoft documentation.

Dataset size	Average time to train
0 - 10 MB	10 sec
10 - 100 MB	10 min
100 - 500 MB	30 min
500 - 1 GB	60 min
1 GB+	3+ hours



Below is the result of training the machine learning model. Based on the result, the best model is LightGbmRegression.

Experiment Results							
Summary							
ML Task: Regression							
Dataset: C:\Users\ITWIDHA\OneDrive - ABB\Documents\ABB Project\Thesis\Wishnu Project Backup\DTModel3.csv							
Label : Current							
Total experiment time : 51,68 Secs							
Total number of models explored: 107							
Top 5 models explored							
	Trainer	RSquared	Absolute-Loss	Squared-Loss	RMS-Loss	Duration	#Iteration
105	LightGbmRegression	0,9352	1,02	1,96	1,38	0,1	105
74	FastTreeTwedieRegression	0,9304	1,19	2,43	1,52	0,5	74
21	LightGbmRegression	0,9275	1,05	2,08	1,42	0,4	21
50	LightGbmRegression	0,9275	1,05	2,08	1,42	0,2	50
36	LightGbmRegression	0,9275	1,05	2,08	1,42	0,3	36
108	SdcaRegression	0,4888	3,80	36,24	5,48	8,4	108
Generate code behind files							

5. Evaluation After we have trained the model. Next, we can use the model to predict the value of the residual current based on the value of demagnetization voltage V_x that we input to the model.

Below we input 200V as a value of V demagnetization. Then we get 34,44 A as a result.

Evaluate

Results of training for your model can be found below.
How do I understand my model performance?

Best model:
RSquared: 0.93
Model: FastTreeWeedieRegression

Try your model

Sample data
 The following fields below are pre-filled by row 1 of your data.

Voltage

Results	
Current	34.44

If we plot the residual current results, we will get below graph.

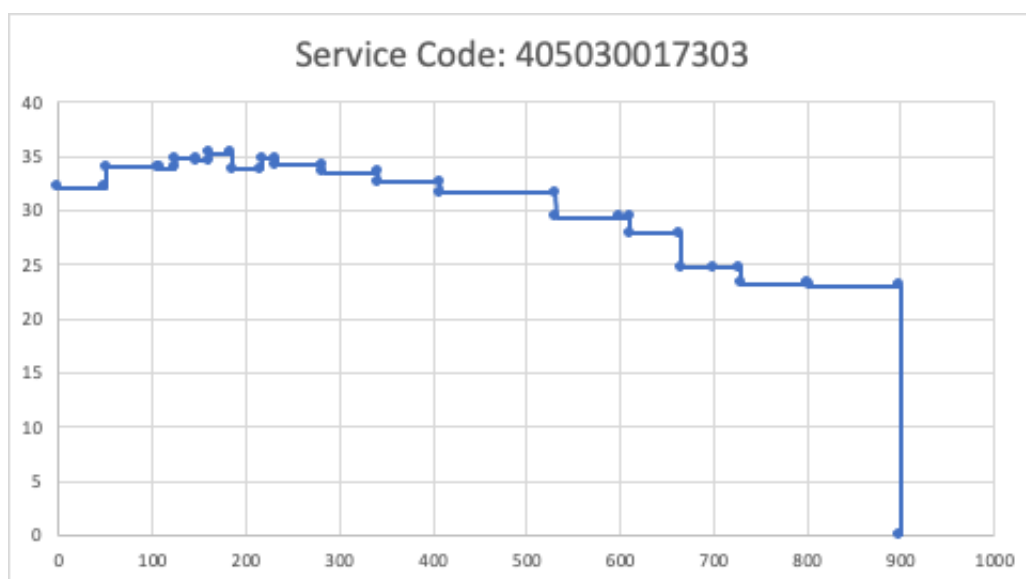


Figure 4.12: Graph of ML.NET Model after training

There are several behaviors that we can conclude based on the graph.

1. As we can see on the graph, the model produces stepped line graph.

2. The model will give constant value within a certain range of V demagnetization. For example, when demagnetization voltage is within 408V – 531V we get constant value of residual current which is 31,65A.
3. After 900V the model assumes that residual current is zero.

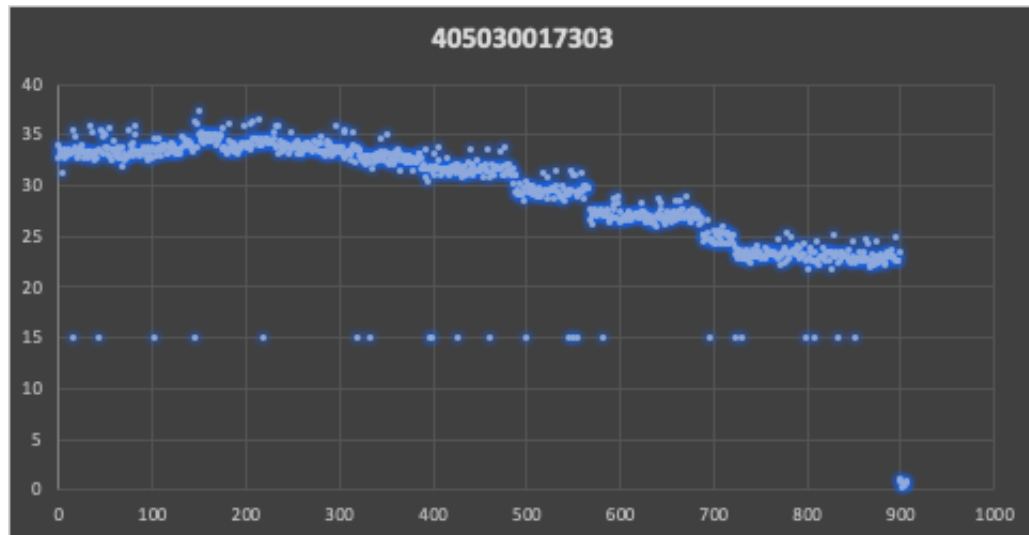
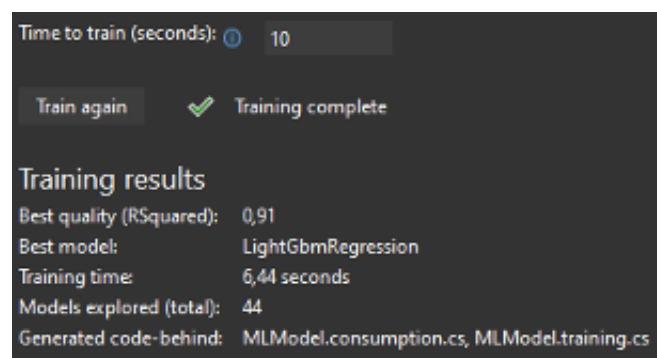


Figure 4.13: Sampling the result of ML.NET Model after training

The graph in figure 4.13 above is obtained by plotting the ML.NET model using Device-Sampling.cs program. As we can see from the graph the result of ML.NET model is like a step function. We also notice that after demagnetization voltage reach 900V, the residual current is always zero.

Implemented ML.NET Model in Digital Twin Application using 10 seconds training time.



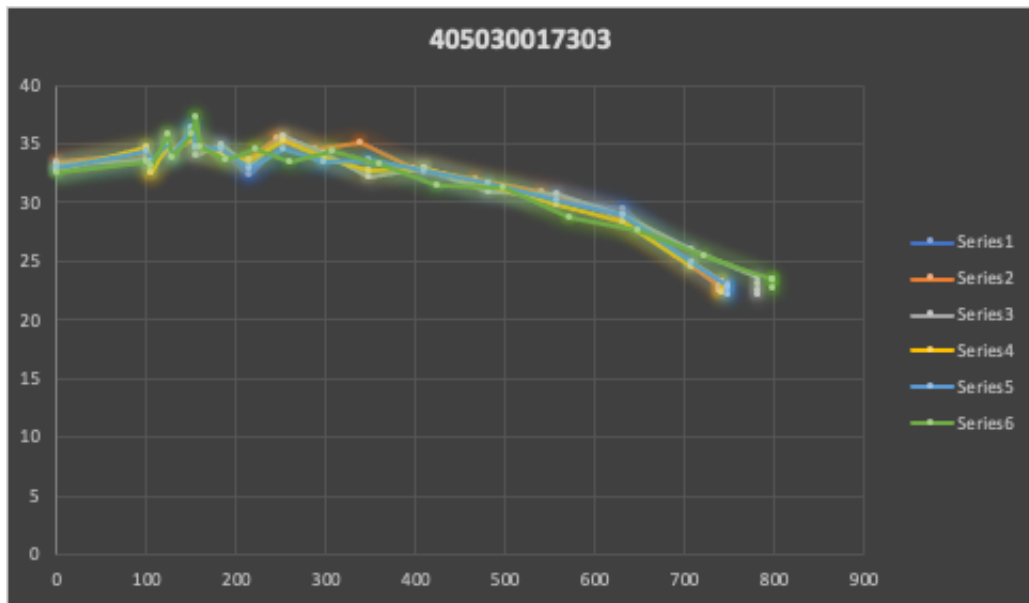


Figure 4.14: Digital Twin simulation using ML.NET model

The above figure 4.14 shows the simulation results of the Digital Twin using the ML.NET model. We can compare the graphs with below graph in figure 5.4 which is the calibration data of the real device.

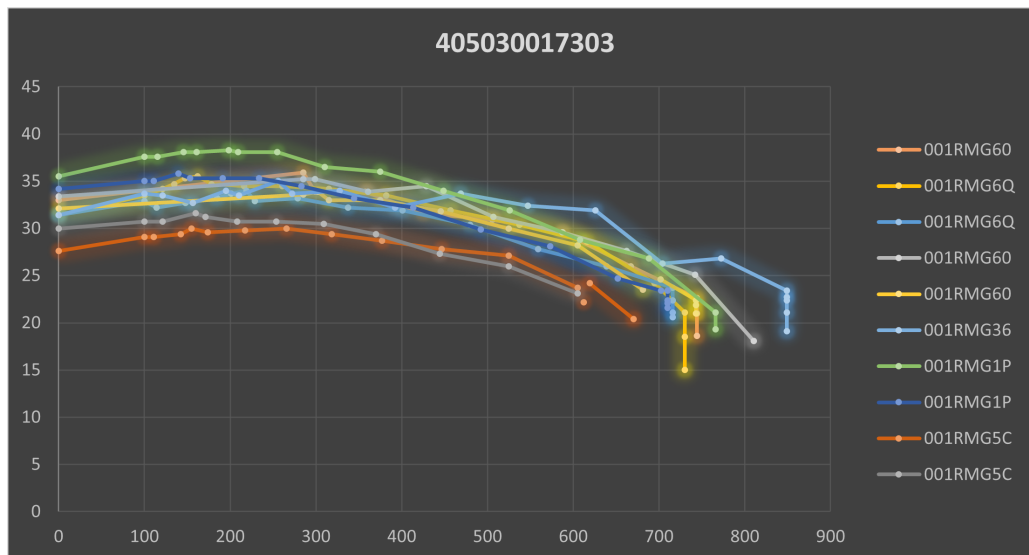


Figure 4.15: Plot of real calibration data of service code 405030017303

Service Code	405030017303
Number of pieces	362 pieces
Average Time	44906 ms

Table 4.4: Calibration Information of Device 001RMG9S

Then we run the calibration for several times, and we obtained below results. According to the data, the MLNET model is able to predict the average calibration time quite accurate which is around 45 seconds.

First Trial:

- Buoni:362
- Scarti:0
- Smagnetizza:1316
- Riarma:2990
- MisuraCorrente:2990
- Magnetizza:53
- Tempo di taratura stimato (sec):16.173,0
- Tempo di taratura medio (sec):44,7

Second Trial:

- Buoni:361
- Scarti:1
- Smagnetizza:1205
- Riarma:2840
- MisuraCorrente:2840
- Magnetizza:52
- Tempo di taratura stimato (sec):15.235,0

- Tempo di taratura medio (sec):42,1

Third Trial:

- Buoni:362
- Scarti:0
- Smagnetizza:1159
- Riarma:2869
- MisuraCorrente:2869
- Magnetizza:55
- Tempo di taratura stimato (sec):15.228,0
- Tempo di taratura medio (sec):42,1

Fourth Trial:

- Buoni:362
- Scarti:0
- Smagnetizza:1444
- Riarma:3180
- MisuraCorrente:3180
- Magnetizza:69
- Tempo di taratura stimato (sec):17.397,0
- Tempo di taratura medio (sec):48,1

5

Neural Network Model for Predicting Demagnetization Voltage

In the previous chapter we have seen the implementation of Digital Twin Application using several Model such as Exponential, Polynomial, ML.NET, and Neural Network. In this chapter we will design a Neural Network Model for predicting the value of demagnetization voltage V_x based on multiple inputs. One of the input is the residual current I_x . The other input will be based on the model design.

The neural network model will consist of three model. Model 0 will accept 2 input which are I_0 and I_x to predict the value of V_x . Model 1 will accept 4 input which are I_0 , I_1 , V_1 , and I_x to predict V_x . Model 2 will accept 6 inputs which are I_0 , I_1 , V_1 , I_2 , V_2 , and I_x to predict the value of V_x .

Model 0 will be used at the beginning of the calibration process after the initial residual current is measured which is the first shot. If after the first shot the residual current is still not within the good range, Model 1 will be used to predict the demagnetization voltage V_x that will be used in the second shot. Then if after the second shot, the residual current is still not within the good range, Model 2 will be used to predict the demagnetization voltage V_x that will be used in the third shot.

In order to train the Model we will use dataset from service code 405030017303. Below is the detail information about the dataset.

Service Code	405030017303
Number of pieces	362 pieces
Average Time	44906 ms

Table 5.1: Calibration Information of Device 001RMG9S

The graph in figure 6.1 below shows the preview of the dataset that we will use to train the neural network model.

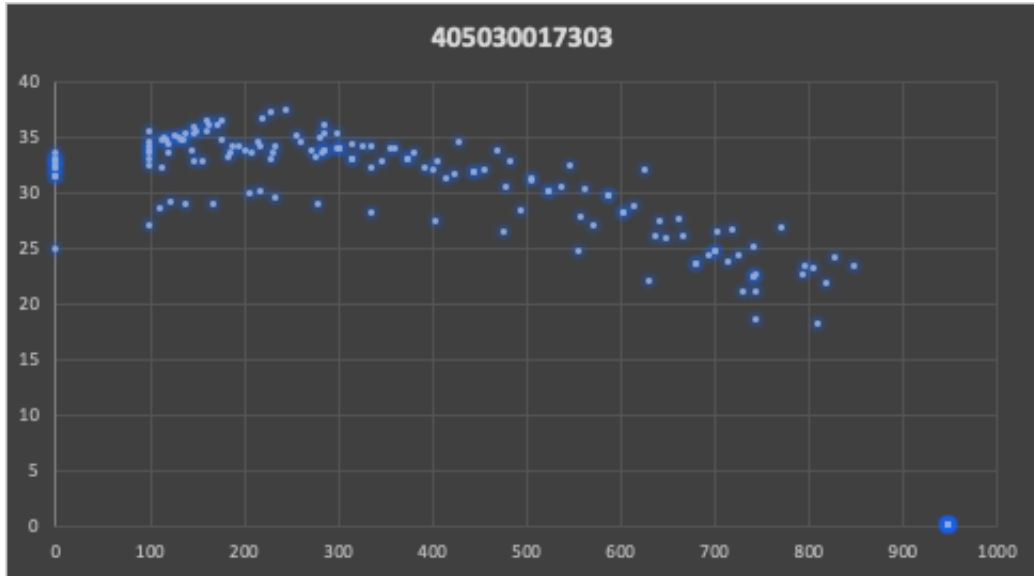


Figure 5.1: Graph of calibration dataset with service code 405030017303

5.1 MODEL O

The Model o will accept two inputs which are I_o and I_x . Our goal is to predict the value of V_x that will be used by the algorithm to find the target residual current of the device.

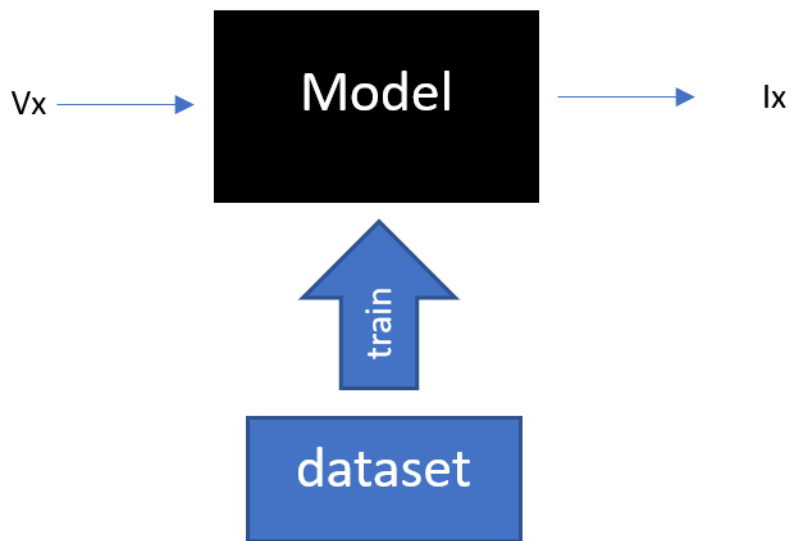


Figure 5.2: Design of Model 0

5.1.1 DATASET

Below is the preview of the dataset. Io and Ix are the input to the model and Vx is the output. The dataset consists of 165 rows.

	Io	Ix	Vx
0	32.8	32.8	0
1	32.8	34.0	100
2	32.8	34.6	113
3	32.8	35.1	127
4	32.8	35.3	138
...

The chart below shows the mean value and standard deviation of each data.

	mean	std
Io	31.948276	1.782394
Ix	23.494828	10.195414
Vx	577.896552	303.385383

Then we will split the data into training set and test set. First, we will shuffle the data randomly, then we will split the data. The training set will consist of 80% and the test set is 20%. We will use the training set to train the model. After the training is completed, we will use the test set to verify the accuracy of the model. Figure 7.3 below shows the distribution of the dataset after splitting them. There are 132 data that goes into training set and there are 33 data that goes into test set.

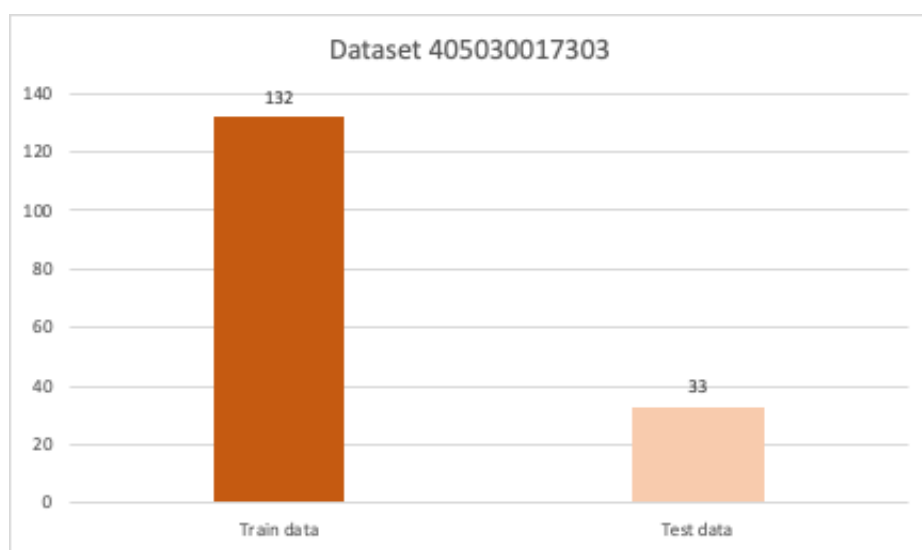


Figure 5.3: Split the dataset into train and test set

5.1.2 NORMALIZATION OF THE DATA

The next step is to normalize the data before we use them for training the model. The main purpose of normalization is to ensure that the training process will be stable. We will shift and scale the inputs into a distribution centered around 0 with standard deviation 1. Below is an example of the I_o and I_x after being normalized.


```

Before Normalization:
[[32.9 35.8]
 [32.9  0. ]
 [31.4 32.4]
 [32.8  0. ]
 [31.4 33.2]]
After Normalization:
[[ 0.59  0.78]
 [ 0.59 -3.01]
 [-0.09  0.42]
 [ 0.54 -3.01]
 [-0.09  0.51]]

```

5.1.3 DESIGN OF MODEL o

Model o will consist of 3 layers as shown in figure 6.4. The first layer is the input layer which receives two inputs I_o and I_x . The second layer is the hidden layer which consists of two hidden layers. The first and second hidden layers consist of 64 neurons. The third layer is the output layer which is the demagnetization voltage V_x .

Model: "sequential"

Layer (type)	Output Shape	Param #
normalization (Normalization)	(None, 2)	5
dense (Dense)	(None, 64)	192
dense_1 (Dense)	(None, 64)	4160
dense_2 (Dense)	(None, 1)	65
Total params: 4,422		
Trainable params: 4,417		
Non-trainable params: 5		

5.1.4 TRAINING PROCESS

Before we can use the model for predicting the demagnetization voltage V_x . We need to train the model using the dataset. Our goal is to minimize the error of the model as small as possible. Therefore we can increase the accuracy of the model. To calculate the error of the model we will use a loss function. The loss function that we will use to train this model is Mean Absolute Error.

$$MAE = \frac{\sum_{i=1}^n |y_i - x_i|}{n} = \frac{\sum_{i=1}^n |e_i|}{n}$$

Our goal is to find the lowest MAE. In order to find the lowest MAE we need to optimize the model multiple times until we found the Lowest MAE. The optimizer that we will use is Adam. Adam optimization is a stochastic gradient descent method. The learning rate that we will use in this Adam optimizer is 0.001.

The training will consist of 1000 epochs, which means that the model will undergo training cycle for 1000 times. We will use 20% of the training data for validation. Validation is useful to monitor when the model is overfitting. If the loss is much lower than the validation loss, then the model is overfitting.

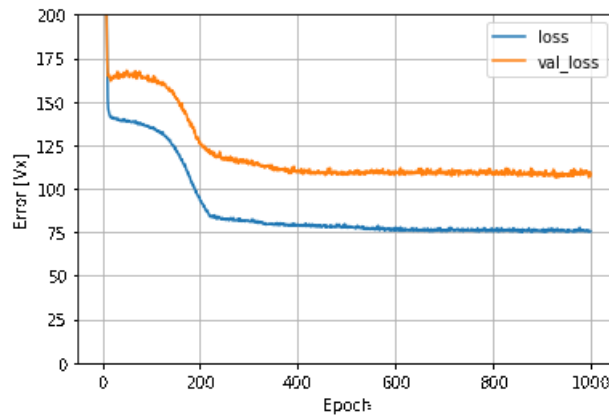


Figure 5.4: Training loss and validation loss of Model 0

After the training process is completed, the next step is to evaluate the model performance. We will use the test set to evaluate the performance.

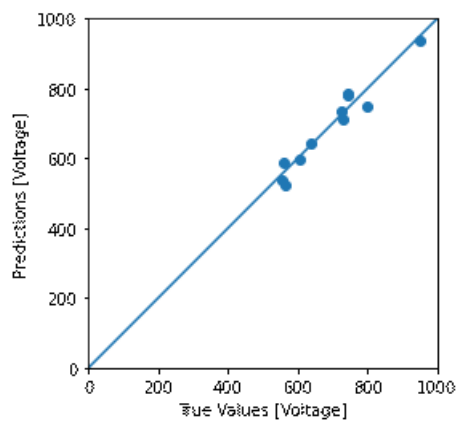
	I0	Ix	Vx
21	32.8	24.3	727
25	32.8	0.0	950
36	32.4	30.3	563
39	32.4	23.4	797
67	31.4	21.1	730
81	31.4	27.8	559
82	31.4	26.0	639
127	24.9	24.6	556
139	33.0	21.0	744
140	33.0	18.6	744
160	32.1	28.2	605

Figure 5.5: Test set for evaluating Model 0

The result of the evaluation is shown below. The average error of the model after we completed the training is around 23.870

Mean absolute error [Vx]	
dnn_model0	23.869524

The diagram below shows the accuracy of the model in predicting the value of Vx. It compares predicted Vx against the true Vx. The closer the predicted value to the line, the higher the accuracy.



Below diagram in figure 5.6 shows the prediction error of Model 0.

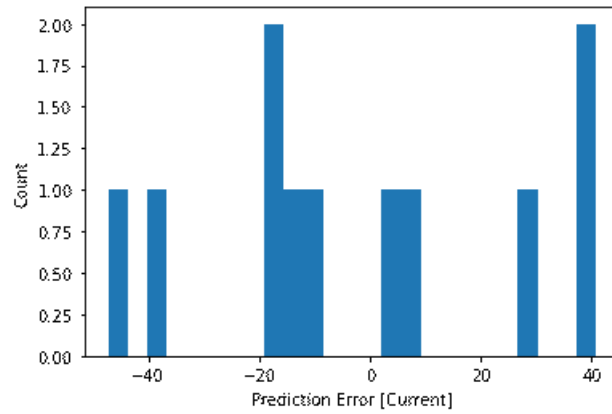


Figure 5.6: Prediction error of Model 0

5.2 MODEL 1

In the previous section we have seen the implementation of Model 0. In this section we will look at the implementation of Model 1. This Model 1 will accept four inputs which are I_0 , I_1 , V_1 , and I_x . Our goal is to predict the value of V_x (demagnetization voltage) that will be used by the algorithm to find the correct residual current of the device.

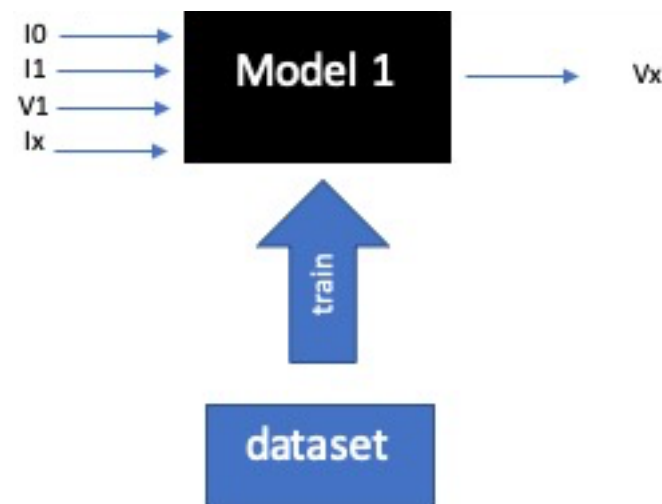


Figure 5.7: Design of Model 1

5.2.1 DATASET

Below is the preview of the dataset. I_0 , I_1 , V_1 , and I_x are the input to the model. V_x is the output. The dataset consists of 129 rows.

	I_0	I_1	V_1	I_x	V_x
0	32.8	34.0	100	34.6	113
1	32.8	34.6	113	35.1	127
2	32.8	35.1	127	35.3	138
3	32.8	35.3	138	35.5	150
4	32.8	35.5	150	36.0	164
...

The chart below shows the mean value and standard deviation of each input.

	mean	std
I_0	31.339806	2.429893
I_1	31.068932	3.842564
V_1	357.563107	218.735298
I_x	30.292233	4.652928
V_x	403.087379	232.814245

5.2.2 NORMALIZATION OF THE DATA

The next step is to normalize the data before we use them for training the model. Below is preview of the data before and after normalization.

```
Before Normalization:
[[ 31.4  33.5 100.   34.2]
 [ 32.4  34.   233.   34.9]
 [ 33.   30.9 506.   29.6]
 [ 31.4  30.4 537.   28.6]
 [ 32.8  36.   172.   34. ]]
```

```
After Normalization:
[[ 0.02  0.64 -1.18  0.84]
 [ 0.44  0.77 -0.57  1.   ]
 [ 0.69 -0.04  0.68 -0.15]
 [ 0.02 -0.17  0.82 -0.37]
 [ 0.6   1.29 -0.85  0.8 ]]
```

5.2.3 DESIGN OF MODEL 1

Model 1 consist of three layers will consist of two hidden layers. The first and second hidden layer will consist of 64 neurons.

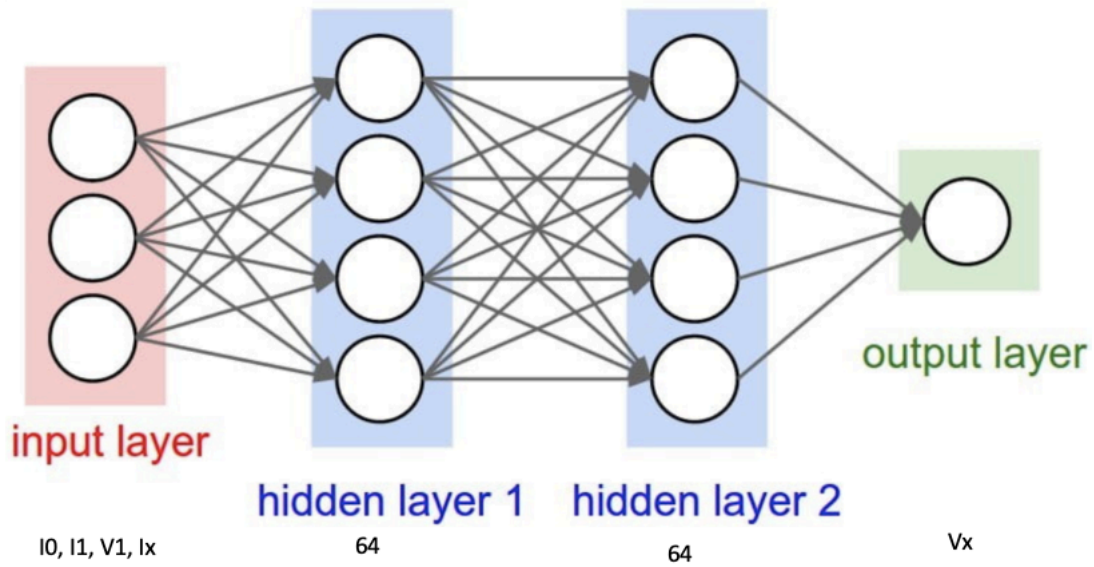


Figure 5.8: Design of Neural Network Model 1

Model: "sequential"

Layer (type)	Output Shape	Param #
normalization (Normalization)	(None, 4)	9
dense (Dense)	(None, 64)	320
dense_1 (Dense)	(None, 64)	4160
dense_2 (Dense)	(None, 1)	65

=====
Total params: 4,554
Trainable params: 4,545
Non-trainable params: 9

5.2.4 TRAINING PROCESS

The training process of Model 1 is similar with the training process of Model 0. The loss function that we will use in this model is Mean Absolute Error. The optimizer that we will use is

Adam with learning rate of 0.001.

The training will consist of 1000 epochs. We will use 20% of the training data for validation. Below graph in figure 7.11 shows the training loss and validation loss during the training process from the first epoch until the 1000th epoch.

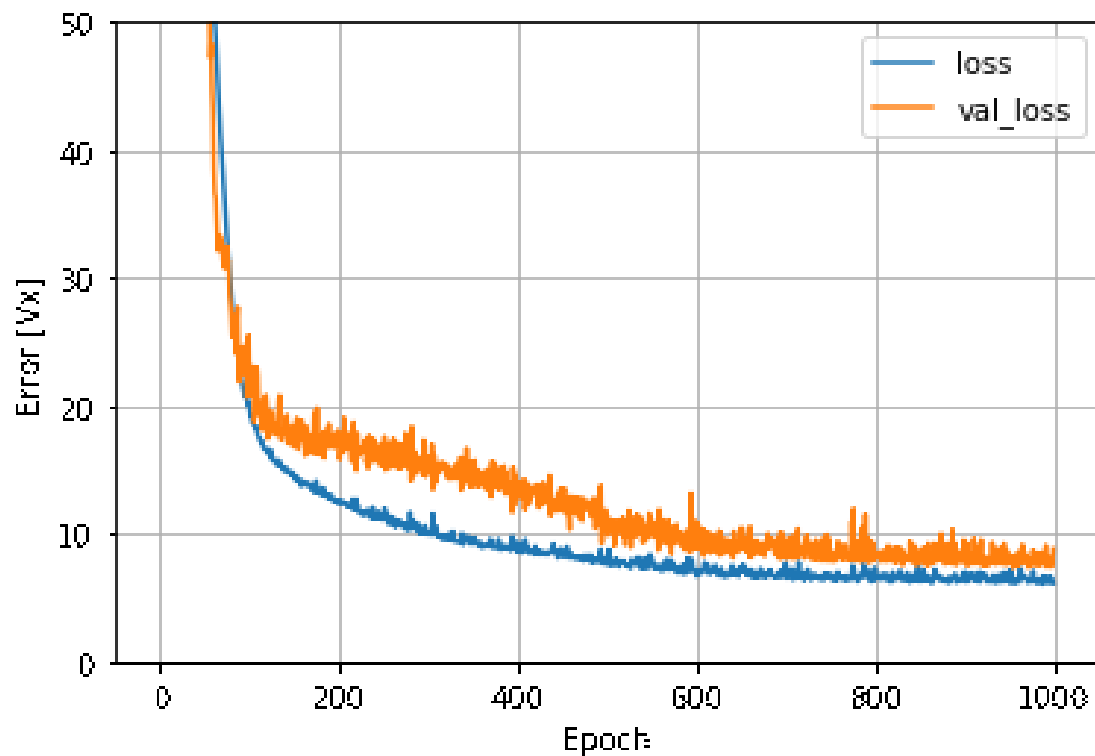


Figure 5.9: Training loss and validation loss of Model 1

After we have completed the training process, next step is to evaluate the model performance. We will use the test set to evaluate the performance.

```
Mean absolute error [Vx]
dnn_model1          9.729853
```

The diagram in 5.10 below shows the accuracy of the model in predicting the value of Vx. It compares predicted Vx against the true Vx. The closer the predicted value to the line, the higher the accuracy.

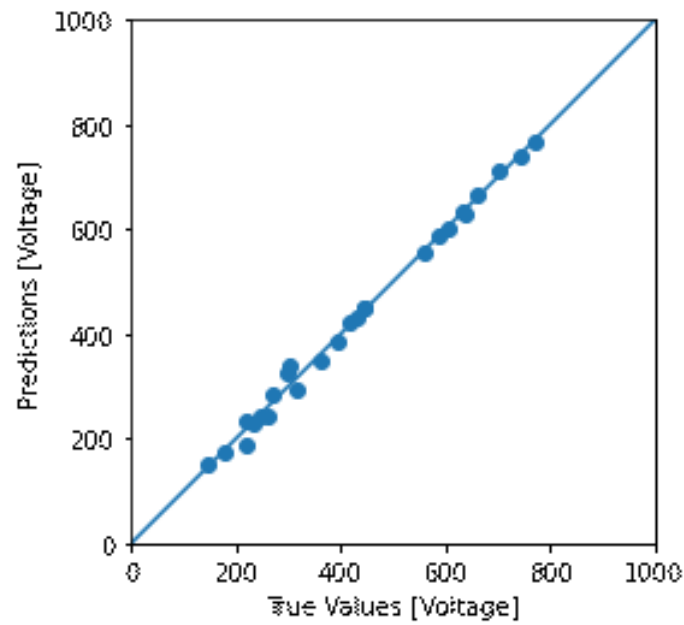


Figure 5.10: Accuracy of Model 1

Below diagram in figure 5.11 shows the error distribution of the predictions. As we can see on the diagram the distribution of the error is mostly around zero.

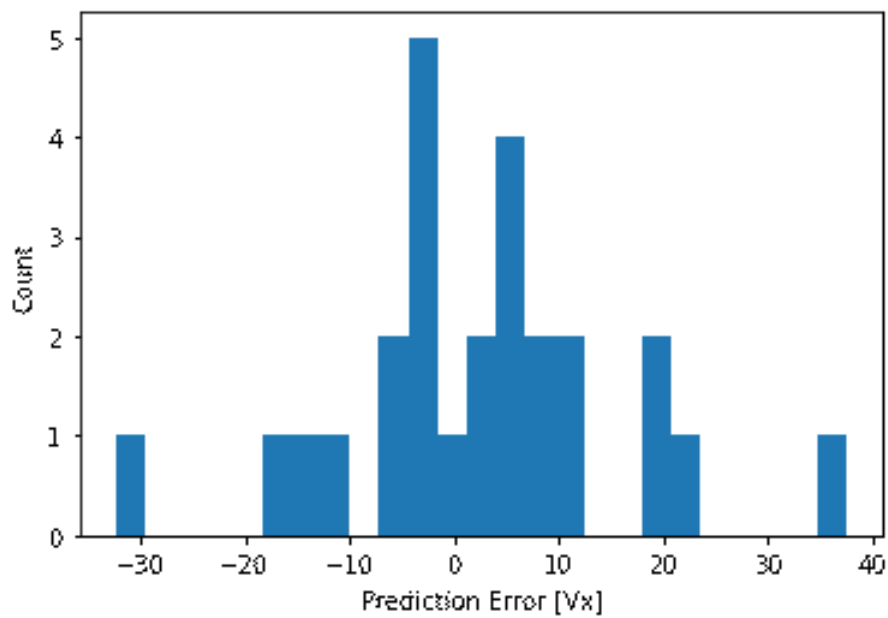


Figure 5.11: Prediction Error of Model 1

5.3 MODEL 2

The Model 2 will accept six inputs which are I_0 and I_x . Our goal is to predict the value of V_x which will be used by the ABB algorithm to find the target residual current of the device.

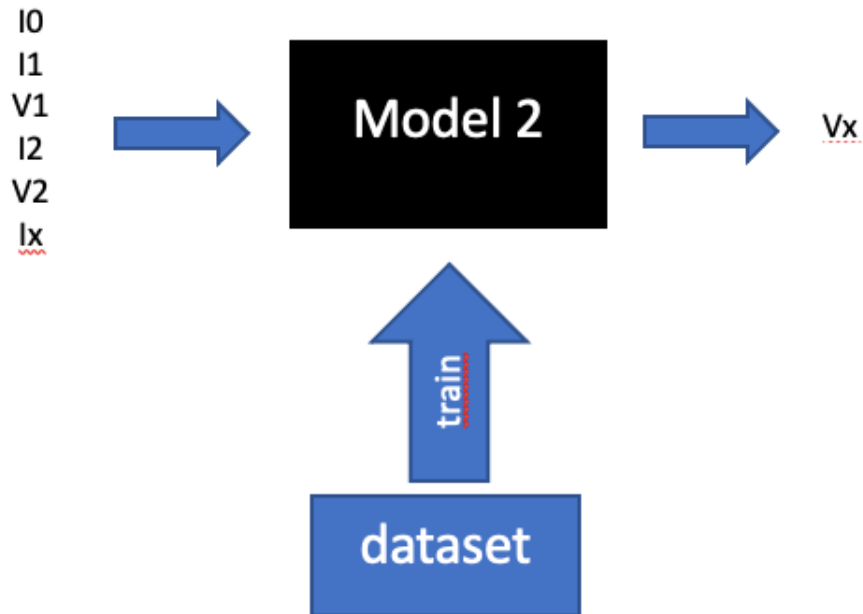


Figure 5.12: Design of Model 2

5.3.1 DATASET

Below is the preview of the dataset. There are six inputs to the Model which are I_0 , I_1 , V_1 , I_2 , V_2 , and I_x , which will be the features. The output is V_x which will be the label. There are total 118 rows in the dataset.

	I_0	I_1	V_1	I_2	V_2	I_x	V_x
0	32.8	34.0	100	34.6	113	35.1	127
1	32.8	34.6	113	35.1	127	35.3	138
2	32.8	35.1	127	35.3	138	35.5	150
3	32.8	35.3	138	35.5	150	36.0	164
4	32.8	35.5	150	36.0	164	34.9	115
...

The chart below shows the mean value and standard deviation of each data.

	mean	std
I0	31.506383	2.253113
I1	31.711702	3.443643
V1	339.361702	198.595274
I2	30.971277	4.077650
V2	386.191489	216.696124
Ix	29.990426	4.887840
Vx	434.000000	230.029545

Then we will split the data into training set and test set. The training set will consist of 80% and the test set is 20%. Therefore, the training set consists of 94 rows of data and the test set consists of 24 rows of data.

5.3.2 NORMALIZATION OF THE DATA

The next step is to normalize the data before we use them for training the model. Below is preview of the data before and after normalization.

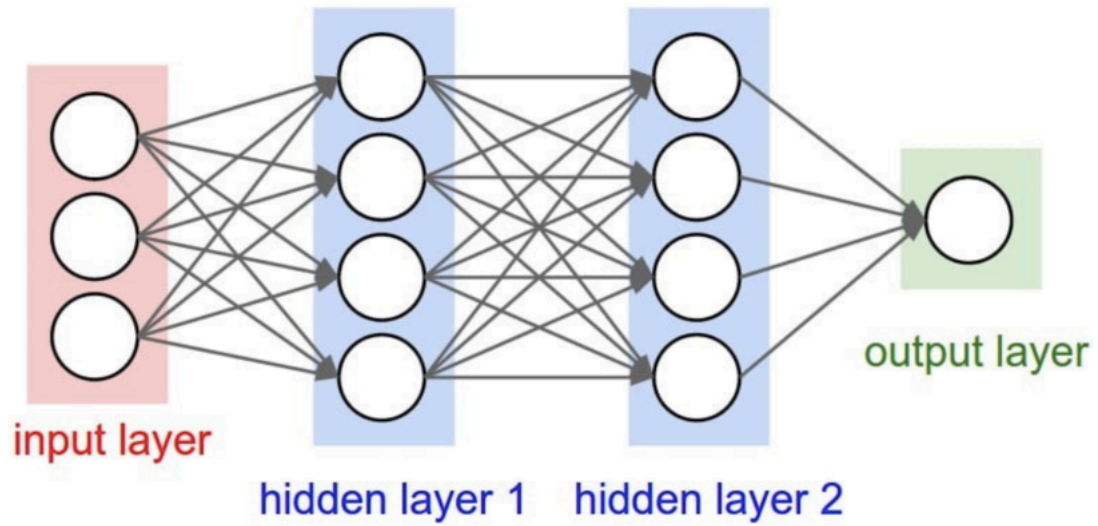
```
Before Normalization:
[[ 24.9  28.9 139.   28.9 169.   29.8]
 [ 24.9  29.1 123.   28.9 139.   28.9]
 [ 24.9  26.4 477.   24.6 556.   21.9]
 [ 32.8  36.  172.   34.  189.   33.5]
 [ 24.9  30.  218.   29.4 233.   28.9]]

After Normalization:
[[-2.95 -0.82 -1.01 -0.51 -1.01 -0.04]
 [-2.95 -0.76 -1.1  -0.51 -1.15 -0.22]
 [-2.95 -1.55  0.7  -1.57  0.79 -1.66]
 [ 0.58  1.25 -0.85  0.75 -0.91  0.72]
 [-2.95 -0.5  -0.61 -0.39 -0.71 -0.22]]
```

5.3.3 DESIGN OF MODEL 2

Model 2 consists of 3 layers as shown in figure 7.14. The first layer is the input layer which receives six inputs I₀, I₁, V₁, I₂, V₂, and I_x. The second layer is the hidden layer which consists

of two hidden layers. The first and second hidden layers consist of 64 neurons. The third layer which is the output layer which is the label V_x .



$I_0, I_1, V_1,$
 I_2, V_2, I_x

64

64

V_x

Figure 5.13: Design of Neural Network Model 2

Model: "sequential"

Layer (type)	Output Shape	Param #
normalization (Normalization)	(None, 6)	13
dense (Dense)	(None, 64)	448
dense_1 (Dense)	(None, 64)	4160
dense_2 (Dense)	(None, 1)	65

=====
Total params: 4,686
Trainable params: 4,673
Non-trainable params: 13

5.3.4 TRAINING PROCESS

The training process of Model 2 is similar with the previous training process of Model 0 and 1. The loss function that we will use in this model is Mean Absolute Error. The optimizer that we will use is Adam with learning rate of 0.001.

The training will consist of 1000 epochs. We will use 20% of the training data for validation. Below graph in figure

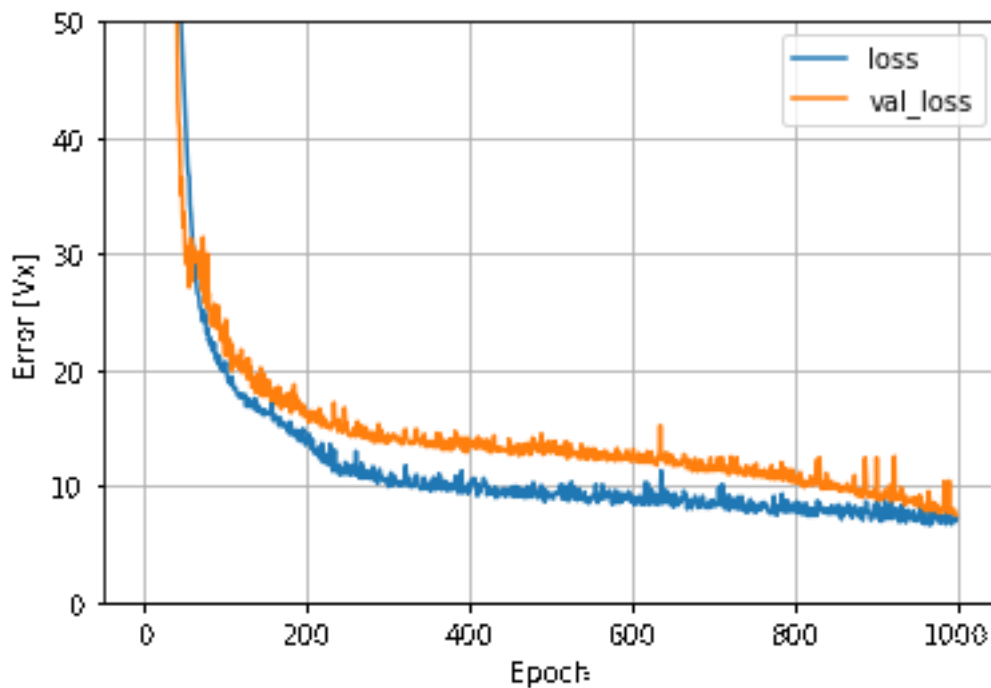


Figure 5.14: Model 2 Training and Validation Loss

After we have completed the training process, next step is to evaluate the model performance. We will use the test set to evaluate the performance.

```
Mean absolute error [Vx]
dnn_model2              6.569524
```

The diagram in 5.15 below shows the accuracy of the model in predicting the value of V_x . It compares the predicted value of V_x against the true value of V_x . The closer the predicted value to the line, the higher the accuracy.

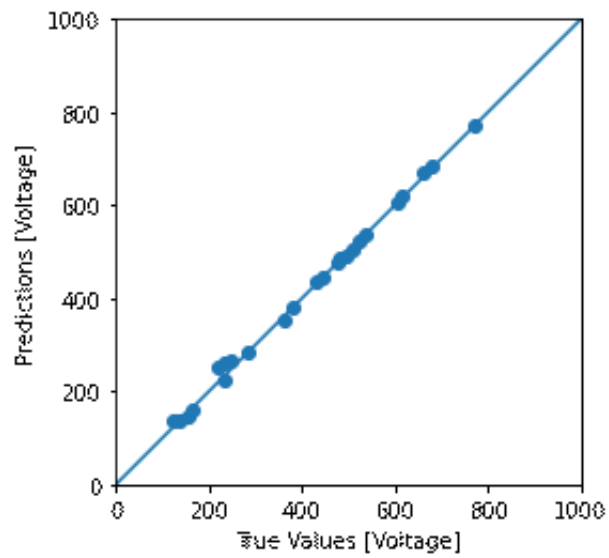


Figure 5.15: Accuracy of Model 2

Below diagram in figure 5.16 shows the error distribution of the predictions. As we can see on the diagram the distribution error is mostly scattered around zero.

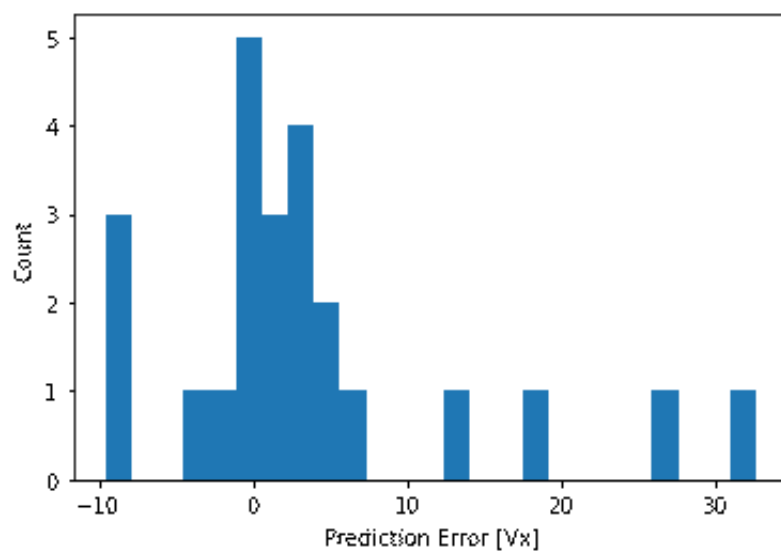


Figure 5.16: Prediction Error of Model 2

6

Conclusion

In the previous chapter we have seen the result of Model 0, 1, and 2. In the first shot of the calibration, the algorithm will use Model 0 to predict the demagnetization voltage V_x . Then, in the second shot of the calibration, the algorithm will use Model 1 to predict the demagnetization voltage V_x . Finally in the third shot of the calibration, the algorithm will use Model 2 to predict the demagnetization voltage V_x . After the third shot, if the algorithm still unable to calibrate the device into the good range of residual current, the algorithm will continuously use Model 2 to find the demagnetization voltage V_x .

Below table shows the summary of the performance of Model 0, 1, and 2 based on the evaluation that we have done in chapter 3. As we can see in the table below the error of each model continue to decrease as we have more inputs. The error of Model 0 is around 23.87. Then the error of Model 1 decrease significantly to around 9.73. Finally the error of Model 2 further decrease to 6.57.

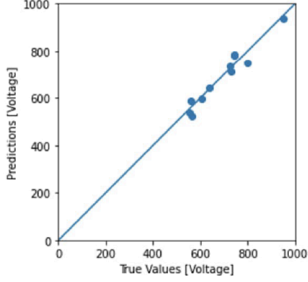
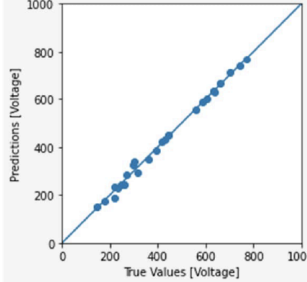
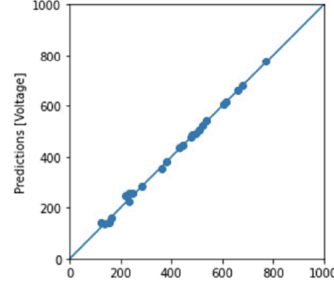
Model 0	Model 1	Model 2
Input Layer: I_0, I_x Hidden Layer: Layer 1: 64 neurons Layer 2: 64 neurons Output Layer: V_x	Input Layer: I_0, I_1, V_1, I_x Hidden Layer: Layer 1: 64 neurons Layer 2: 64 neurons Output Layer: V_x	Input Layer: $I_0, I_1, V_1, I_2, V_2, I_x$ Hidden Layer: Layer 1: 64 neurons Layer 2: 64 neurons Output Layer: V_x
Training Parameters Epochs: 1000 Batch size: 4 Validation split: 0.2	Training Parameters Epochs: 1000 Batch size: 8 Validation split: 0.2	Training Parameters Epochs: 1000 Batch size: 8 Validation split: 0.2
Average Error Mean absolute error [Vx] dnn_model0 23.869524	Average Error Mean absolute error [Vx] dnn_model1 9.729853	Average Error Mean absolute error [Vx] dnn_model2 6.569524
Prediction Accuracy 	Prediction Accuracy 	Prediction Accuracy 

Figure 6.1: Evaluation Result

In the table above, we can see that the prediction accuracy from Model 0 to Model 2 is improving steadily.

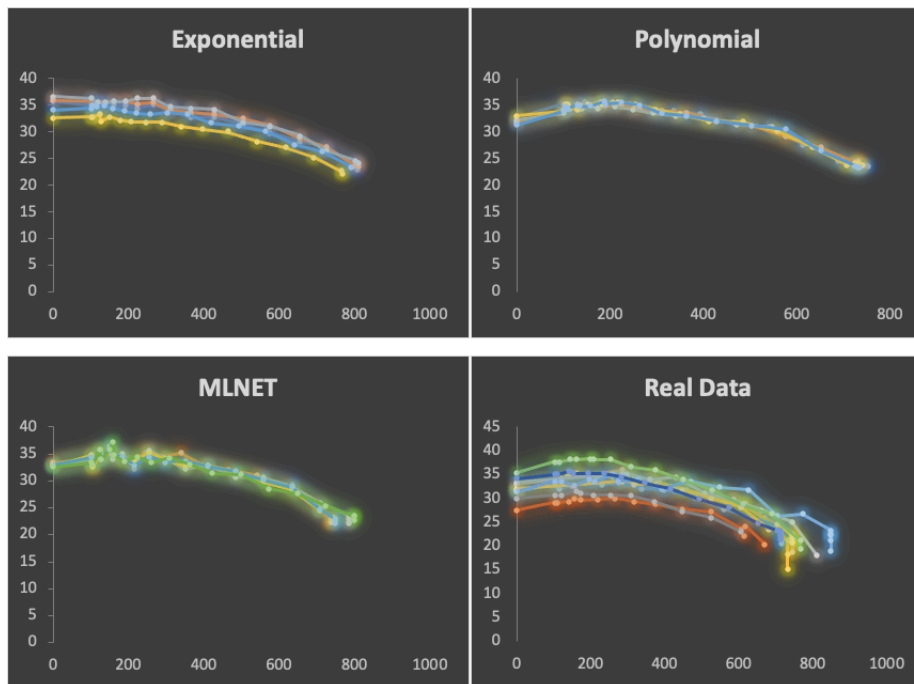
Therefore, based on the information above, we can conclude that we can use Artificial Neural Network (ANN) for predicting the demagnetization voltage during calibration. The accuracy of the ANN model will keep increasing as we provide more input to the ANN model.

CONCLUSION ON DIGITAL TWIN FOR SIMULATING RCCB CALIBRATION

In the Part 1, we have seen a list of behaviors of the RCCB Devices. Then based on these behaviors we create a Digital Twin application that could simulate the behavior of the device during the calibration process. The software design and implementation of the Digital Twin has been discussed in Part 2.

With the Digital Twin application that is developed in C#, we can apply several models to simulate the behavior of the RCCB calibration process. In Part 2 we have tried 3 model for this simulation that is Exponential Model, Polynomial Model, and ML.NET Model. All the models can predict the average calibration time that is close to the real calibration time. However, the shape of the calibration graph is different across each model.

Based on the graphical results of each model. We can conclude that the Polynomial model is the model that best represent the calibration behavior of the real data in terms of visual graph.



With this Digital Twin application, now we can safely optimize the calibration algorithm and simulate the results without risking the production process. Meanwhile before the Digital Twin is created, we must adjust the calibration algorithm and implemented it directly in the production to see the results. Which can be very risky for the production if the adjustment parameters make the calibration process even slower than before.

CONCLUSION ON NEURAL NETWORK MODEL FOR CALIBRATING THE RCCB

Chapter 6 has discussed about the use of neural network model for predicting the value of V_x that the ABB algorithm will use to calibrate the RCCB. Based on the results, we can conclude that we can use these models for predicting the value of V_x . The higher the model that we use the higher the accuracy.

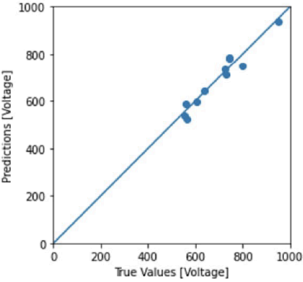
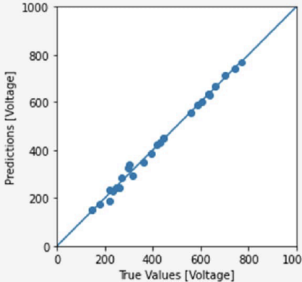
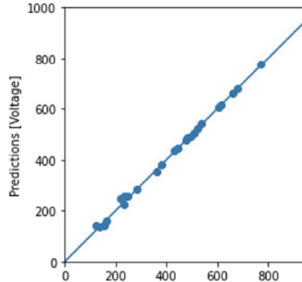
Model 0	Model 1	Model 2
Input Layer: I_0, I_x Hidden Layer: Layer 1: 64 neurons Layer 2: 64 neurons Output Layer: V_x	Input Layer: I_0, I_1, V_1, I_x Hidden Layer: Layer 1: 64 neurons Layer 2: 64 neurons Output Layer: V_x	Input Layer: $I_0, I_1, V_1, I_2, V_2, I_x$ Hidden Layer: Layer 1: 64 neurons Layer 2: 64 neurons Output Layer: V_x
Training Parameters Epochs: 1000 Batch size: 4 Validation split: 0.2	Training Parameters Epochs: 1000 Batch size: 8 Validation split: 0.2	Training Parameters Epochs: 1000 Batch size: 8 Validation split: 0.2
Average Error Mean absolute error [V_x] dnn_model0 23.869524	Average Error Mean absolute error [V_x] dnn_model1 9.729853	Average Error Mean absolute error [V_x] dnn_model2 6.569524
Prediction Accuracy 	Prediction Accuracy 	Prediction Accuracy 

Figure 6.2: Evaluation Result

FURTHER DEVELOPMENT

There are several ideas to further develop the current Digital Twin application.

The Digital Twin can be further developed by adding more devices with different service code. Therefore, we can simulate the calibration process of more devices.

Currently the Digital Twin application that is developed is a console-based application. Therefore, if we want to simulate different device model, we have to edit directly using Visual Studio. In the future, we can add user interface within the application. This user interface will enable the user of the Digital Twin to modify the simulation parameters directly on the application. Therefore, a user who doesn't have software development skills can use this Digital Twin Application.

The next development that we could add to the current Digital Twin Application is a graph visualization. Currently the output of the Digital Twin is the value of current and voltage which in .csv format. Therefore, if we want to visualize the result, we have to plot it manually using MS Excel. In the future, we can add feature in the Digital Twin application for visualizing the graph result of the calibration process simulation. Therefore, we can perform graph visualization within the Digital Twin application without the need of other software like MS Excel.

Further Development of the Neural Network Model for calibrating the RCCB

The Neural Network Model that we currently trained is for one RCCB device. In the future, we can add more RCCB device model to the Neural Network Model. Therefore, the model capable of predicting the V_x for many RCCB devices.

References

- [1] W. contributors, “Digital Twin,” in *Wikipedia, The Free Encyclopedia*, 2022.
- [2] ABB, “Protection against earth faults with residual current devices,” *Technical Guide*, 2015.
- [3] J. Mills, “Iec 62873-2: Residual current operated circuit-breakers for household and similar use - part 2: Residual current devices (rcds) - vocabulary,” *IEC*, 2016.
- [4] C. Amadori, “L’interruttore differenziale. un’introduzione concisa,” 2015.
- [5] P. K. D. Smugala, “Ma7 relay parameters calculations and multiphysics modeling,” *ABB, Tech. Rep.*, 2013.
- [6] F. Freschi, “High-frequency behavior of residual current devices,” *IEEE Transactions on Power Delivery*, vol. 27, no. 3, pp. 1629–1635, 2012.
- [7] T. Mitchell, “Machine learning,” *McGraw Hill*, 1997.
- [8] J. Mills, “Introduction to ai part 1,” *Edzion*, Dec. 2020.
- [9] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” *Nature*, vol. 521, pp. 436–444, 2015.
- [10] A. V. L. Ghezzi, “Analisi comparativa di alcune possibili soluzioni per una stazione di tarature per relé differenziali,” *ABB Sace LPD, Tech. Rep.*, 2016.
- [11] D. V. Froicu, “Modeling and learning based calibration of residual current devices,” *Master Thesis in Università di Bologna*, 2020.

Acknowledgments

In 2022 I was doing an internship at ABB Santa Palomba, a city about 20 km south of Roma. During this internship, I was doing a Thesis project to create a digital twin application.

I would like to say special thank you to my academic tutor Professor Federica Battisti for guiding me through this process in creating the thesis.

My deep gratitude to my project manager Nicola Risoli who gave me suggestions and guidance in developing the application and algorithm.

I would like to give special gratitude to my colleagues at ABB, Michelle di Mare and Giovanni Milanese who have supported me during the internship and thesis project.

And finally, I would like to dedicate this thesis work to my lovely parents who always supported me during my entire study in the University of Padua.