NOVA SCHOOL OF SCIENCE & TECHNOLOGY

DEPARTMENT OF
ELECTRICAL AND COMPUTER ENGINEERING

MIGUEL ALEXANDRE AMARAL TEIXEIRA

Degree in Sciences of Electrical and Computer Engineering

DATA-DRIVEN MODELLING OF AC LOSSES IN HIGH-TEMPERATURE SUPERCONDUCTING COILS

MASTER IN ELECTRICAL AND COMPUTER ENGINEERING

NOVA University Lisbon
march, 2022

DEPARTMENT OF
ELECTRICAL AND COMPUTER ENGINEERING

NOVA SCHOOL OF
SCIENCE & TECHNOLOGY

DATA-DRIVEN MODELLING OF AC LOSSES IN HIGH-TEMPERATURE SUPERCONDUCTING COILS

MIGUEL ALEXANDRE AMARAL TEIXEIRA

Degree in Sciences of Electrical and Computer Engineering

Adviser: Doutor João Miguel Murta Pina
Assistant Professor, NOVA SST

Examination Committee

Chair: Doutor Luís Filipe dos Santos Gomes
Associate Professor, NOVA SST

Rapporteur: Doutor Pedro Miguel Ribeiro Pereira
Auxiliar Professor, NOVA SST

Member: Doutor João Miguel Murta Pina
Assistant Professor, NOVA SST

MASTER IN ELECTRICAL AND COMPUTER ENGINEERING

NOVA University Lisbon
march, 2022

Data-driven modelling of AC losses in high-temperature superconducting coils

This document was created with the (pdf/Xe/Lua)LaTeX processor and the NOVAthesis template (v6.9.2) [1].

To my family.

Acknowledgements

First and foremost, I would like to express my gratitude to my advisor Professor João Pina for his support and the expertise he provided me with during the development of this project.

Secondly, I would like to thank Doutor Roberto Oliveira for his helpfulness and all the help he provided me with.

I would also like to thank NOVA School of Science and Technology for all the knowledge it has given me during these years.

Finally, I would like to thank my family for supporting me during these years and my entire life and my friends for accompanying me during this journey.

"You cannot teach a man anything; you can only help him discover it in himself." (Galileo)

Abstract

Predicting the loss in superconductive power devices is of utmost importance when designing such devices. This is because the cooling system needs to be designed accordingly. The current methods for predicting AC Loss are either inaccurate or very time consuming.

These conventional methods for predicting loss are of two types in which one is faster but inaccurate, while the other is very accurate but also very time consuming. While currently they are both employed in different stages of the design process, there is an interest in a faster, but still accurate, form of predicting AC Loss.

Studies have time and time again shown that Artificial Neural Networks are capable of taking on complex tasks and handling them faster than regular computing. Because of this, in this work, an Artificial Neural Network based approach is proposed as to predict AC Loss in various configurations of HTS coils. This approach aims to replicate the accuracy of standard numerical models while being much faster than said models.

This results in a final framework comprised of two distinct sequential Neural Networks that are capable of predicting the AC Loss for different configurations of HTS coils nearly instantaneously while still being very accurate and reliable in their predictions.

Keywords: Superconductor, HTS, Power Devices, AC Loss, Artificial Neural Network

Resumo

A capacidade de previsão de perdas em dispositivos de potência supercondutores é um assunto de alta importância aquando do desenho dos mesmos. Isto deve-se ao facto de o sistema de arrefecimento necissitar de ser desenhado de acordo com as mesmas. Os métodos atuais de previsão de perdas AC são ou pouco fiávies, ou bastante demorados.

Estes métodos atuais de previsão de perdas são de dois tipos em que um é mais rápido mas pouco preciso, enquanto o outro é bastante preciso mas, no entanto, muito demorado. Embora atualmente sejam ambos empregados em fases diferentes do processo de desenho, continua a existir interesse numa forma rápida e precisa de prever perdas AC.

Estudos têm vindo a provar que as Redes Neuronais são capazes de enfrentar tarefas complexas e lidar com elas de forma mais rápida que a computação tradicional. Dado isto, neste trabalho propõe-se uma abordagem baseada em Redes Neuronais para prever perdas AC em várias configurações de bobinas HTS. Esta abordagem visa a replicar a fiabilidade de métodos numéricos sendo, no entanto, bastante mais rápida.

Isto resulta numa *framework* final composta por duas Redes Neuronais distintas sequenciais que é capaz the prever perdas AC em diversas configurações de bobinas de forma quase instantânea sendo, no entanto, bastante correta e confiável nas suas previsões.

Palavras-chave: Supercondutor, HTS, Dispositivos de Potência, Perdas AC, Redes Neuronais

Contents

List of Figures

List of Tables

Acronyms

1G	First Generation HTS Tape
2G	Second Generation HTS Tape
AC	Alternating Current
ANN	Artificial Neural Network
BSCCO	High-Temperature Superconductor: $Bi_2Sr_2Ca_nCu_{n+1}O_{6+2n}$
DC	Direct Current
FEM	Finite Element Method
HTS	High-Temperature Superconductor
SMES	Superconducting Magnetic Energy Storage System
YBCO	High-Temperature Superconductor: $YBa_2Cu_3O_{7-\delta}$

Symbols

B_c	Critical Magnetic Flux Density (T)
J	Current Density
B	Magnetic Flux Density (T)
J_c	Critical Current Density
M	Magnetization (A/m)
MSE	Mean Squared Error
μ_0	The value of vacuum permeability: $4\pi * 10^{-7}$ H/m
R	Coefficient of Correlation
ρ	Electrical Resistivity ($\Omega * \mathrm{m}$)
R^2	Coefficient of Determination
T_c	Critical Temperature (K)
T	Temperature (K)

1 Introduction

1.1 Background and Motivation

A major challenge when implementing a new a power device with superconductive elements is to be able to design a proper cooling system capable of handling any excess heat present in the system. The biggest source of heat in said systems is the loss through Joule's Effect in the superconductive element itself. The challenge is being able to predict the amount of loss in a system given its configuration as to create a proper cooling solution.

Although there are ways to predict the loss in superconductive materials, most of these ways are either imprecise or time-consuming. Having models that predict loss precisely but are slow increases the duration of the design phase of a superconductive power device significantly.

Current modeling solutions for predicting loss in superconductors are based on the physical properties of these materials. This means that these models are mathematically very complex and, because of this, require an immense computing power in order to be calculated. This need for computing power leads to an increase in duration of the design phase, but also to an increase in cost as more computing power translates to more investment.

1.2 Research Question and Hypothesis

This works aims, therefore, to propose a more efficient way of accurately predict the AC loss in superconductive power devices. Having a more efficient model would lead to a decrease in time spent on the design stage and to a decrease in overall investment as less computing power would be necessary.

Research Question: Is there a more efficient way of predicting AC loss in superconductive power devices?

The solution to this question proposed in this work is to implement and Artificial Intelligence model to predict this loss. The aim is to make use of Deep Learning to convert a small amount of known data into an infinite prediction model that is able to,

given the configuration, predict the possible AC loss.

Hypothesis: An Artificial Intelligence based model would be able to accurately predict AC loss in superconducting power devices much faster that current mathematical models.

1.3 Objectives

This work is part of the project **tLOSS - Transformando o Cálculo de Perdas em Sistemas de Potência com Supercondutores de Alta Temperatura** [2]. One of the objectives of this project is to transform the methodologies of calculating loss in devices with superconductive components. As such, the main objective of this work is to implement and Artificial Intelligence based model that is capable of predicting AC loss in different configurations of High-Temperature Superconductor (HTS) coils. To achieve this, smaller objectives have to be met:

This first objective is to be able to characterize the loss in the tape that is used in the coils. This means finding parameters that are unique to that tape so that different tapes are viewed by the model as different inputs. Since this only has to be done for a small piece of HTS tape, and the AC loss in this small pieces of tape behaves more predictably, the aim is to find general equation to characterize the loss. This equation will have constants that are different for different tapes and these constants will be used to characterize the tape.

The second objective is to implement a prediction model to an individual coil. For this, and Artificial Neural Network will be used to predict loss in said coil. The different parameters of a coil: tape type, number of turns, inner radius, operating current and frequency, will be used as inputs for the network and the only output will be the corresponding AC loss.

The third objective is to implement a prediction model for a stack of similar coils. For this, another Artificial Neural Network will be used and the inputs will be the number of coils in a stack and the AC loss in an individual coil.

The final objective is to put the different models together in order to predict loss in a stack given only the tape parameters, the operating current and frequency, the coil parameters (inner radius and number of turns) and the number of coils in the stack. By separating the prediction for a single coil and a stack will possibly make it easier to predict loss for an individual coil as treating a coil as a stack of one coil in unnecessary.

1.4 Expected Contributions

The following contributions are expected to result of this work:

- A large dataset of AC loss values for different HTS tapes, coils and stacks of coils;
- An Artificial Intelligence based model that is able to accurately predict the AC loss in different configurations of HTS devices.

1.5 Document Structure

This document is divided into five Chapters. It starts with the present Chapter, an **Introduction** do the document itself.

After this introductory chapter, a description of the current **State of Art** relating to the study subject is given. This description is subdivided into three sections: the first explains the basics of superconductivity, the second explains basic aspects of Artificial Neural Networks and the final one presents related studies.

This is followed by a Chapter describing the **Preparation and Setup** that went into this work. This Chapter is divided into four Sections: the first describes the preparation of the datasets used in the work; the second describes the training process used for the Neural Networks; the third explains the setup of combining both Networks and the final section explains the Network performance analysis method.

Then, in the fourth Chapter the **Results** from the experimental work are presented. This Chapter is divided into four sections: the first presents the results from the ANN used for predicting loss in HTS coils; the second presents results from the ANN used for predicting loss in HTS coil stacks, the third presents the results from the combination of both ANNs and, finally, the fourth sections presents a comparison of said results.

The final Chapter presents the **Conclusion** from this study and ideas on what can be done to improve the results.

2

State of Art

2.1 Superconductivity

Superconductivity is a different state of matter in which a material has no electrical resistance and, in some cases, presents perfect diamagnetism. For these reasons, especially the first one, these materials are of very high interest for their possible applications in energy transmission. For this reason, a contextualization on superconductivity is provided.

2.1.1 Introduction to Superconductivity

The phenomenon of superconductivity was first observed by Dutch physicist Heike Kamerlingh Onnes in 1911. Onnes had previously studied the resistance of metals at liquid helium temperatures and hypothesised that, for pure enough metals, their resistance would drop to 0Ω [3]. When experimenting with mercury in 1911, Onnes observed that bellow 4.2K, its electrical resistance plummeted to 0Ω [4]. The plot of his observations is depicted in Figure 2.1.

Later studies in the field of superconductivity led to better understanding this state. An important aspect is the conditions that need to be met to achieve and maintain this state [5]:

1. $T < T_c$: the temperature of the material must be lesser than its critical temperature;

2. $B < B_c$: the magnetic field surrounding the material must be smaller than the materials critical field;

3. $J < J_c$: the current density that flows through the material must be inferior to the critical current density.

If all of the above stated conditions are met, then the material is able to enter and maintain its superconductive state.

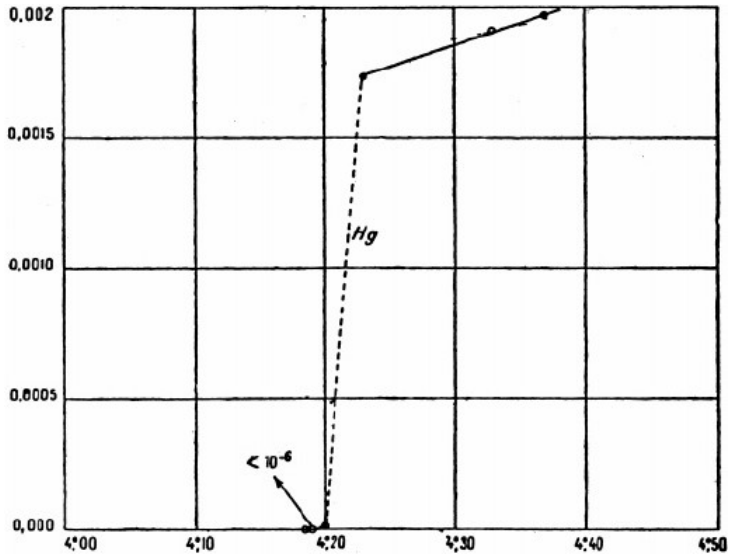

Figure 2.1: Resistance-Temperature plot of Onnes' experiment with Mercury [6]

2.1.1.1 The Meissner Effect

The Meissner Effect was discovered by German physicists Walther Meissner and Robert Ochsenfeld in 1933. While Onnes had discovered that, in the superconductive state, materials had no electrical resistance, Meissner and Ochsenfeld observed that, in that same state, these materials would expel any magnetic field that tried to penetrate them [7].

This expulsion of the incident magnetic field is what is called the Meissner effect. By expelling all magnetic field, the material exhibits perfect diamagnetism meaning its behaviour is not influenced by the outside field. While some superconductors are what is commonly called Full-Meissner, meaning that they are always perfect diamagnets in their superconductive state, some are not. This difference is explained in section 2.1.1.2.

2.1.1.2 Type I and Type II Superconductivity

While in the early stages, all the discovered superconductors exhibited the aforementioned behaviour, in 1957 soviet physicist Aleksei Abrikosov predicted the existence of a new type type of superconductor to which he gave the name of Type II [8].

This new type of superconductor also has to be maintained in conditions 1. ($T < T_c$) and 3. ($J < J_c$), the second condition varies. This is because Type II superconductors have two distinct critical magnetic field values B_{c1} and B_{c2}, where $B_{c2} > B_{c1}$. Type II superconductors are just like Type I superconductors if the magnetic field $B < B_{c1}$.

But, if the magnetic field values is comprehended between the two critical values $B_{c1} < B < B_{c2}$ the material exhibits a new behaviour. While its electrical resistance is still virtually 0Ω, the material is penetrated by the magnetic field and, therefore, is not a perfect diamagnet. This difference was introduced in the previous section:

- **Type I Superconductors** present zero electrical resistance and Full-Meissner Effect in their superconductive state. They only have one critical magnetic field value B_c, above which they are no longer superconductive. The magnetization characteristic of these materials is depicted in Figure 2.2.

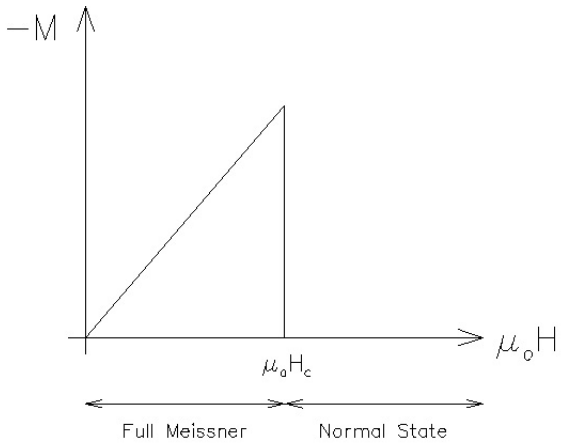

Figure 2.2: Magnetization in a Type I superconductor

- **Type II Superconductors** present zero electrical resistance in their superconductive state. They have 2 distinct critical magnetic field values and only exhibit Full-Meissner Effect bellow B_{c1}. Above this value, they start to increasingly be penetrated by the outside magnetic field until B_{c2} is reached where the material leaves its superconductive state. The magnetization characteristic of a Type II superconductor is depicted in Figure 2.3.

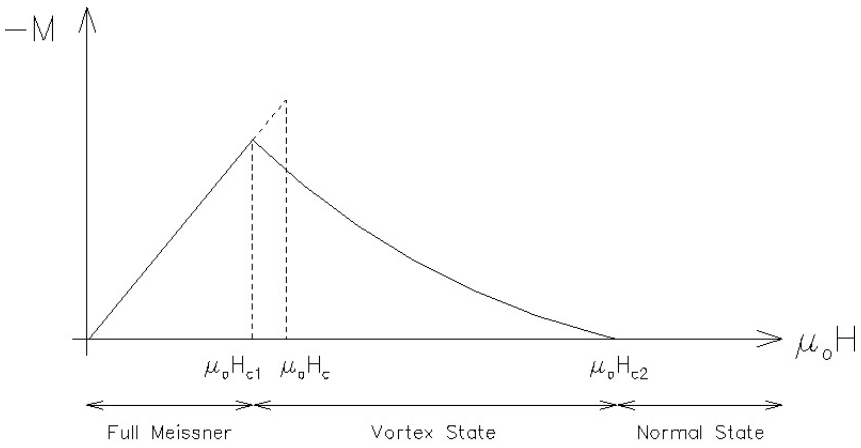

Figure 2.3: Magnetization in a Type II superconductor

2.1.2 High Temperature Superconductivity

Another important discovery on the field of Superconductivity was that of High Temperature Superconductors. These are also the materials we are mainly focused on this work as their application is much greater due to their relatively high critical temperature values.

As the name indicates, High Temperature Superconductors (HTS) have a relatively high critical temperature value (T_c). While the first high temperature superconductor, discovered by Bednorz and Müller, had a critical temperature of $T_c = 35\text{K}$ [9], in present days is common to call high temperature superconductors to materials that present a critical temperature higher than that of liquid nitrogen $T = 75\text{K}$.

So, as was previously mentioned, the first ever high temperature superconductor was discovered by german physicist Johannes Bednorz and swiss physicist Karl Müller in 1986. These scientists observed that, in the Ba – La – Cu – O system, the critical temperature was $T_c = 35\text{K}$ [9].

Since the discovery of HTS, several new high temperature superconductors have been discovered. In this work, we are focused on tapes of HTS materials presented in Section 2.1.2.1.

2.1.2.1 HTS Tapes

HTS tapes present an inviting form of using superconductive material as the tape format allows for multiple applications. Since the problem of low critical temperatures is solved by HTS and the tape format allows for a multitude of applications, the study of these tapes is of high interest amongst the scientific community.

Having said this, there are currently two distinct generations of HTS tapes with different materials and construction among them.

First Generation HTS Tape (1G)

First Generation HTS tapes are manufactured using compounds of $Bi_2Sr_2Ca_nCu_{n+1}O_{6+2n}$ (BSCCO). The process of constructing these tapes consists of having filaments of superconductive material encased in silver or silver alloy sheaths [5].

The main drawback of these tapes is that their critical current will severely drop when an outside magnetic field is present [10]. This led to the creation of a new kind of HTS tape, the second generation.

Second Generation HTS Tape (2G)

Second Generation HTS tapes are constructed out of $YBa_2Cu_3O_{7-\delta}$ compounds (YBCO). Their construction process also differs from first generation tapes as they are formed by different thin layers deposited in a substrate. The layers alternate between superconducting material and adhesion layers [11].

These second generation tapes perform better in the presence of magnetic fields as the critical magnetic field for these tapes is much greater than 1G [10].

As stated before, the main advantage of the tape format in superconductors is the possibility to employ them in diverse applications. These applications may include:

- **HTS transformers:** Because superconductors have little loss, this makes them ideal for use in transformers. There have been some HTS transformers that have already been built. One example of one of these transformers is a 630kVA three-phase transformer developed and operated by ABB in the grid of Geneva's utility (SIG) during the year of 1997 [12].

  Another example of a real HTS transformer is one that powered an entire power substation in China for six months in 2011. This transformer was also three-phase with a rated power of 630kVA [13].

- **HTS motors:** HTS also offers a great alternative to building motors. There have also been several HTS motors built during the years (*e.g.*): in 2005, a three-phase, 100hp, 4 pole, 1800rpm synchronous motor with a superconducting rotor was built and tested in Korea Electrotechnology Research Institute (KERI) [14]; in 2014, the first fully HTS synchronous motor was built and tested at Cambridge University [15].

2.1.3 Loss in superconductors

As previously stated, the scope of this work is to accurately predict loss in superconductive systems. This is of upmost importance when developing said systems for one very important reason: loss in electric conduction takes the form of heat.

This heat generated is especially harmful when taking into consideration that these systems are operated at cryogenic temperatures. This means that these losses have to be accounted for when designing the cooling system for these devices. Having said this, the logical conclusion is that there has to be a way of calculating the possible loss that will occur in one of these devices as to design a suitable cooling system. This is where the purpose of this work takes form: as we will see, current ways of modelling loss in superconductive systems are either inaccurate or very time consuming opening the door for a faster, but still accurate approach.

2.1.3.1 Cooling Systems

As stated previously, superconductors need to be kept at a temperature lower that their critical on in order to maintain their superconductive state. This means that there is a need for a cooling system when operating on this type of materials.

These cooling systems need to be able to combat not only the zeroth law of thermodynamics, meaning that the heat from the neighbourhood of the system will be absorbed by said system, but also the heat that comes from the system itself.

2.1.3.2 Types of Loss in Superconductors

Even though superconductors have virtually no loss when operating with Direct Current (DC) as their resistivity is very close to 0Ω, this is not true for Alternating Current (AC). When in the presence of an Alternating Current, Faraday's Law and the subsequent Maxwell Equation (Equation 2.1) states that the variation in the magnetic field will lead to an electric field.

$$\nabla \times \mathbf{E} = -\frac{\partial \mathbf{B}}{\partial t} \tag{2.1}$$

Loss in superconducting systems takes various forms. These losses can be divided into two separate groups depending on the mechanisms that originates them [11]. These groups are:

- **Magnetization losses:** originated by variations in the magnetic field;
- **Transport current losses:** originated by variations in the current flowing through the superconductor.

Magnetization Losses

Magnetization losses are originated by variations in the applied magnetic field. The different magnetization losses are:

Hysteretic Losses

When a varying magnetic field is applied in a material, the work done by the magnetization performs an hysteresis cycle. After one full cycle, the potential energy must return to its initial state leaving the remaining energy to be dissipated as heat. This energy lost through heat is called hysteresis loss [16].

In superconducting devices, hysteretic losses can be both superconducting and ferromagnetic in nature. This is because 2G tapes have a ferromagnetic substrate making them liable to ferromagnetic hysteretic loss. Superconducting hysteresis, on the other hand, happen because variations in the applied magnetic field lead to the formation of hysteresis cycles in the superconducting material itself [5].

An example of an hysteresis cycle is presented in Figure 2.4.

The loss through hysteresis between two different values of magnetic field can be calculated by:

$$Q = \oint M dB_a \tag{2.2}$$

Coupling current losses

As mentioned in 2.1.2.1, 1G tapes are formed by superconducting filaments surrounded by a silver matrix. When this tapes are subjected to varying magnetic fields, they produce shielding currents in the superconductive filaments and coupling currents

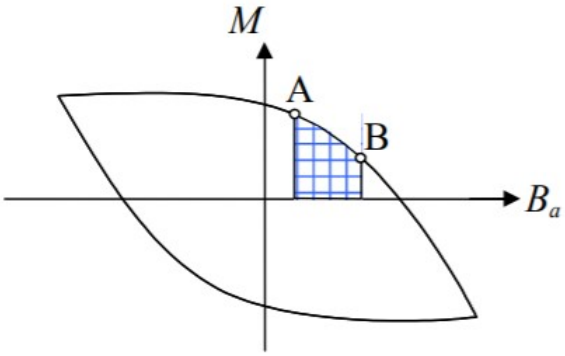

Figure 2.4: Hysteresis cycle [5]

in both the filaments and the silver matrix [11]. This behaviour is portrayed in Figure 2.5.

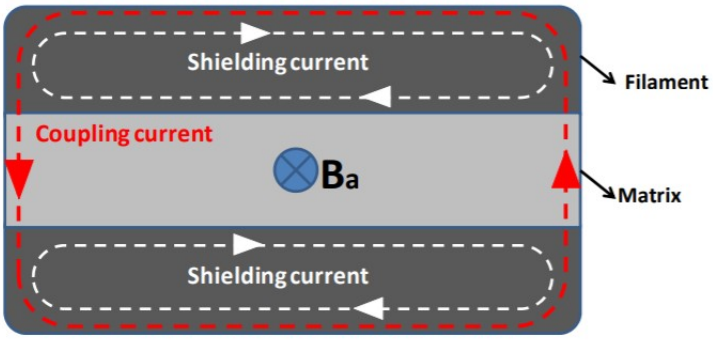

Figure 2.5: Shielding and coupling currents [5]

Eddy current losses

Second Generation HTS tapes have, in their construction, that react to changes in the magnetic field generating Eddy currents. These currents are induced in these layers and flow in closed loops in planes perpendicular to the magnetic field.

The power lost through Eddy Currents per unit of mass of conductor can be calculated by [17]:

$$P = \frac{\pi^2 B_p^2 d^2 f^2}{6\rho D} \tag{2.3}$$

where B_p is the peak magnetic field, d is the thickness of the sheet and D is the density of the material.

We can derive from Equation 2.3 that the amount of power lost through Eddy currents is directly proportional to the frequency. This means that, for low frequencies, the power lost through Eddy currents is negligible.

Transport Current Losses

Transport current losses are caused by the current that flows in the superconducting tape. There are different mechanisms that lead to this kind of loss:

Self-field losses

These losses are derived from Ampère's Law that states that current flowing through a conductor generates a magnetic field. This is also true for superconductors and the field that is generated takes the name of self-field. This self-field also generates hysteresis loops leading to loss.

Flux-flow losses

This type of loss occurs only when the superconductor is operated in its mixed state. But this is usually the state in which HTS are operated as its much easier to achieve than the purely dieletric state.

When superconductors are operated in their mixed state, some of the magnetic field penetrates the material in some specific areas called vortexes. When current flows through the material, these vortexes tend to move towards the edges of the superconductor. If the interaction force caused by the current flow surpasses the force that pins the vortexes, these vortexes will start to move freely inside the superconductor leading to energy dissipation [18].

Resistive losses

These losses occur when the current density being put through the HTS tapes surpasses the critical value for the superconductor being used. When this happens, some of the current will start to flow through the non-superconductive parts of the tape. Because these are not superconductive, they have resistance and, therefore, Ohm's Law will apply.

2.1.4 Modelling Loss in Superconductors

Presently, there are already various different ways of modelling loss in superconductive devices. The problem with current models is either their relative inaccuracy or their slow computing times. This is where the subject of this document will come into play.

But, before introducing the project itself, an explanation on current models will be provided in order to properly comprehend their drawbacks.

Current models are divided into two groups: **analytical** and **numerical** models. Analytical models are much simpler than numerical ones requiring much less time to compute. They are, on the other hand, inaccurate in their predictions. Numerical models are very accurate but take a great amount of time to compute. This means that, usually, both kinds of models are employed when developing a superconducting system: At an early stage, analytical models are used in order to have a general idea of what kind of

loss to expect in said system; At a later stage, numerical models are employed to properly predict the loss of the system [11].

2.1.4.1 Analytical Models

As the name suggests, this type of model takes an analytical approach to calculating possible loss in a superconducting device. This means finding an approximation of the system and making suppositions in order to greatly simplify the calculations.

Some analytical models are, but are not limited to:

Norris Model

This model was first introduced by W. T. Norris in 1970. It assumes that the superconductor is infinite in length and that the critical current density is independent from the applied magnetic field [19].

In his article published under the Journal of Physics, "*Calculation of hysteresis in hard superconductors carrying ac: isolated conductors and edges of thin sheets*"[19], Norris gives loss approximation formulae to different superconductor configurations. As this work is mainly focused on superconducting tapes, the appropriate formula to approximate loss is the one for losses in a thin strip of finite width is given by Equation 2.4.

$$Q_c = \frac{I_c^2 \mu_0}{\pi}\{(1-F)\ln 1 - F + (1+F)\ln 1 + F - F^2\} \quad (\mathrm{J/m}/cycle) \tag{2.4}$$

where $F = \frac{I_{peak}}{I_c}$, I_{peak} being the peak current and I_c the critical current associated with the superconducting tape.

The problem is that this model makes two assumptions that are highly untrue for real applications. The first one being that the tape is infinite in length which clearly wrong and that the critical current density is independent of the magnetic field, which is also wrong as stated by Norris himself [19]:

> "*The critical current density is independent of ambient magnetic field although it is well known that critical currents depend not only on the magnitude but also the direction of the field.*"

Clem Model

This model was introduced by John R. Clem, J. H. Claasen and Yasunori Mawatari in 2007. It also assumes that the critical current density is independent from the applied magnetic field. The critical difference between this model and Norris' is that it estimates loss for a finite vertical stack of tape. This is a great improvement as it allows for estimation of loss in pancake coils as a coil can be approximated as a stack of tape.

An important assumption made in this model is that, for an individual strip of tape, the width of the superconducting strip is much greater than its thickness [11]. The model

considers different levels of flux penetration, using the Bean model to estimate the impact of said penetration.

The formula achieved to estimate the loss in an individual strip is given by Equation 2.5.

$$Q'_{strip} = \frac{16\mu_0 {J_c}^2 a^2 b^2}{\pi}[(1-F)\ln 1 - F + (1+F)\ln 1 + F - F^2] \quad (\mathrm{J/m}/cycle) \tag{2.5}$$

where $F = \frac{I_{peak}}{I_c}$, I_{peak} being the peak current and I_c the critical current associated with the superconducting tape, $2a$ is the width of the tape and $2b$ is its heigth [20].

Pardo Model

This final model was first introduced by Enric Pardo in his pubication "*Modelling of coated conductor pancake coils with a large number of turns*"[21]. This model also aims to simplify the calculations of loss in superconducting tapes but taking into account the separation between different superconducting layers. This makes this model interesting as 2G tapes have the superconducting layers stacked between adhesion layers having, therefore, a physical space between them.

This model starts from Norris' equation for superconducting slabs [11] and makes the following assumptions:

1. The magnetization is independent from the outside magnetic field;
2. The pancake coil has a high number of turns;
3. The inner radius of the coil is much greater than its width.

The problems of analytical models

As seen before, analytical models make several assumptions in order to reach a conclusion. This makes the final equation much easier to compute as several variables are removed in this simplification but leads to a highly inaccurate final result.

2.1.4.2 Numerical Models

Numerical models, on the other hand, are very precise in their predictions. This is mainly due to the fact that they make no false assumptions in other to simplify the computing process.

One example of a widely used numerical model is:

H-formulation numerical model

This model is based on the Finite Element Method (FEM) and the H-formulation of Maxwell's equations [22]. It uses Finite Element Method (FEM) to solve Faraday's equation (Equation 2.1), using the magnetic field **H** as a state variable. It also uses non-linear resistivity [23] meaning that the resistivity is not constant when the magnetic field is varied:

$$\rho = \frac{E_0}{J_c}\left|\frac{\mathbf{J}}{J_c}\right|^{n-1} \tag{2.6}$$

where E_0 is a characteristic electric field and n is a factor indicating the steepness of the transition from the superconducting to the normal state [23].

From the analysis of Equation 2.6, it is easily comprehended that the resistivity in the superconductor is a function of the current density that flows in said superconductor. Because, as stated in section 2.1.4.1, the current density is dependent on the applied magnetic field, the conclusion is that the materials resistivity is also influenced by said magnetic field.

The superconductor is modelled as a material with a relative magnetic permeability of $\mu_r = 1$ [23]. This leads to Faraday's equation in terms of the magnetic field:

$$\frac{\partial(\mu_0\mu_r\mathbf{H})}{\partial t} + \nabla \times (\rho \nabla \times \mathbf{H}) = 0 \tag{2.7}$$

Equation 2.7 can then be solved using FEM software.

The problems of numerical models

The problem with numerical models is not their accuracy as they yield results that are very close to reality. The problem is, instead, the time and resources they take in order to compute said results. Because these models take a great amount of variables, they require a lot of computing power and time to make the necessary calculations.

2.1.4.3 The proposed framework

The scope of this work is to introduce a different framework to compute loss in a superconducting device. Instead of trying to describe these devices as mathematical models, the objective is to train an Artificial Neural Network to predict the loss given the device's configuration.

This framework will, in theory, mitigate the problems of both kinds of models as a properly trained neural network yields very accurate results and, despite taking a great amount of time to train, after the training stage these networks are very fast in calculating these predictions.

In the next section, a theoretical background on Artificial Neural Networks will be provided in order to better comprehend this powerful tool.

2.2 Artificial Neural Networks

The Merriam-Webster dictionary defines a Neural Network as *"a computer architecture in which a number of processors are interconnected in a manner suggestive of the connections between neurons in a human brain and which is able to learn by a process of trial and error"* [24].

The theory of an Artificial Neural Network (ANN) is to artificially mimic the workings of the human brain [25]. This is achieved by implementing a mathematical representation of the biological neuron, denominated artificial neuron, and creating connections between said artificial neurons.

2.2.1 Artificial Neuron

As stated previously, the artificial neuron is a mathematical representation of its biological counterpart. It works by taking an input, or several, applying a transfer function to that input and returning the result in the form of a scalar output [26].

Each artificial neuron receives one or more inputs, each with an associated weight value w. The neuron then sums these inputs with a bias value b. A transfer function is then applied to this sum resulting in the scalar output a. A visual representation of this mechanism is provided in Figure 2.6.

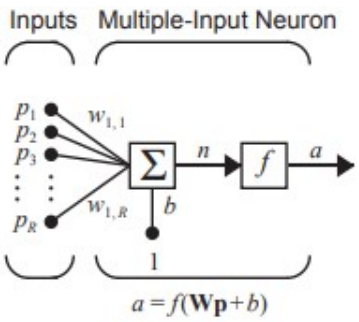

Figure 2.6: Artificial Neuron

By having multiple artificial neurons in parallel, a neuron layer is formed. In this layer, multiple neurons can share multiple inputs with different weight values for each neuron. A representative example of a neuron layer is portrayed in Figure 2.7.

As we can see in Figure 2.7, an input vector is shared between each neuron, but each input carries a weight vector with a possibly different value for each neuron. Each neuron also has its own bias value. This assures that the outputs will be different for each artificial neuron.

The stacking of multiple neuron layers in series results in an Artificial Neural Network. The nomenclature for the different layers in an ANN is:

- **Input Layer:** the first layer in an ANN that directly receives the network inputs;
- **Output Layer:** the final layer in an ANN that yields the final outputs of the network;

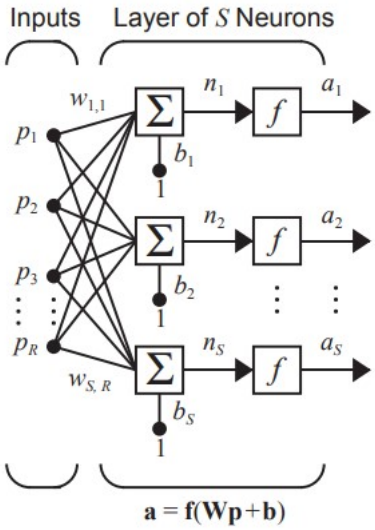

Figure 2.7: Neuron Layer

- **Hidden Layers:** the middle layers between the input and output layers. Its inputs and outputs are never known by the user, hence the name.

An ANN is comprised by one input and one output layers, and by one or more hidden layers. The number of inputs in the input layer is the same as the number of different inputs for the network. Since an artificial neuron only generates one output, the number of neurons in the output layer is equal to the number of outputs of the network.

2.2.2 Classification of Neural Networks

Currently, most researchers agree that Artificial Neural Network (ANN)s are divided into two distinct categories [26]:

1. Feed-Forward Neural Networks
2. Recurrent Neural Networks

Feed-Forward Neural Networks:

In this first kind of Neural Network architecture signals are unidirectional, meaning that they can only go from input to output and not the other way around. This means that the output of a given layer can only have an impact of the following layers [27].

This architecture is simpler that recurrent networks. A graphical illustration of a simple feed-forward neural network is portrayed in Figure 2.8.

Recurrent Neural Networks:

This type of architecture is not unidirectional. The output of an artificial neuron is passed on to the next layer but is also fed into the same neuron again. This is called

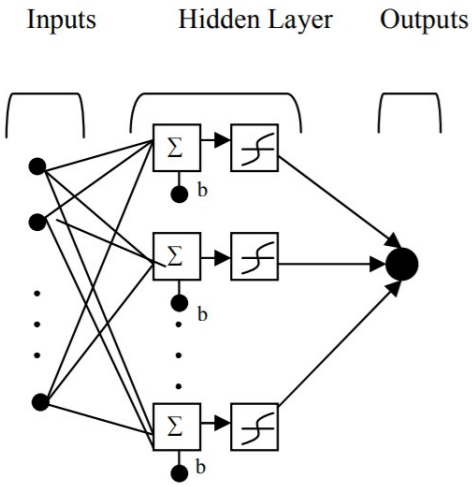

Figure 2.8: Simple feed-forward neural network

a feedback loop [27]. This allows neurons with feedback loops to store information on previous inputs.

This architecture is more complex than feed-forward networks due to the existence of these feedback loops. An illustration of a simple recurrent network is portrayed in Figure 2.9, where **D** represents a delay box that feeds the output of a given artificial neuron back to it with a certain amount of delay.

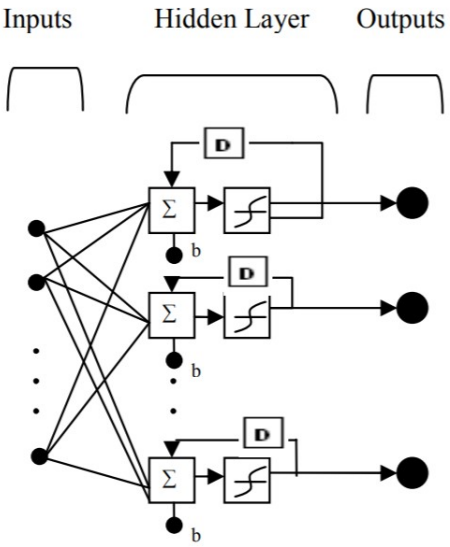

Figure 2.9: Simple recurrent neural network

Convolutional Neural Networks:

Convolutional Neural Networks have a similar architecture to simple feed-forward ones. The main difference is that, some of the hidden layers are convolutional layers. These layers convolve its inputs before passing the through to the next layer. This allows

for far less artificial neurons in the hidden layers.

2.2.3 Machine Learning Algorithms

Now that the different network architectures have been presented, there is also a need to differentiate the multiple algorithms used to train said networks. These algorithms are also divided into different categories. These are:

- Supervised Learning Algorithms;
- Unsupervised Learning Algorithms;
- Semi-supervised Learning Algorithms;
- Reinforcement Learning Algorithms.

Supervised Learning Algorithms:

These algorithms make use of labeled datasets to train neural networks. They are mostly used to classify data or predict outcomes [28].

Supervised Learning works by having a dataset that provides the correct outputs for a given set of inputs. These algorithms constantly measure their accuracy during the training phase, adjusting until the error has reached a small enough value [28].

Since this work makes use of a labeled dataset having as inputs the characteristics of the superconducting device and as output the loss associated with said characteristics, and the purpose of the network is to predict loss for different configurations, the safe conclusion is that supervised learning algorithms will be best for the use case at hand. Even so, a brief explanation on the other algorithms will be given for contextualization.

Unsupervised Learning Algorithms:

This type of algorithm is employed when using unlabeled datasets. They are especially useful in pattern recognition or data grouping applications without the need for human input [29].

Semi-supervised Learning Algorithms:

These algorithms are used when a small part of the available dataset is labeled. They use this small labeled dataset to guide feature extraction and classification from a larger unlabeled dataset [30].

Reinforcement Learning Algorithms:

These final algorithms use state-action pairs that represent the state of an environment at a given time and the possible actions for that given state. They explore these pairs in order to find the actions that lead to a goal state [31].

Now that all algorithms have been briefly introduced, the conclusion presented in the first one is still valid. Since the dataset used for this work is labeled, having the value of loss for each configuration of HTS device, and the goal is to predict loss values for different, unseen configurations, Supervised Learning is clearly the way to go.

2.2.4 Training Artificial Neural Networks

The training stage is where an ANN learns to yield the required results. In the case of Supervised Learning, the Network examines the inputs and outputs and adjusts the weight and bias values to achieve the outputs with the inputs. This adjustment process is called training and it is done several times, each iteration given the name of epoch.

Different training algorithms have different ways of going about these adjustments and different criteria to determine when to stop the training. In this work, only the algorithms present in the MATLAB^{TM} Neural Network Toolbox will be explained as this is the tool that will be used for the experimental part.

2.2.4.1 Levenberg-Marquardt Algorithm

The first training algorithm is the **Levenberg-Marquardt** algorithm. This algorithm was developed in the 1960's with the purpose of solving nonlinear least squares problems [32]. It is the combination of two different algorithms: the gradient descent method and the Gauss-Newton method.

This algorithm actively varies the parameter updates between the two methods [32]. The algorithm contains a damping parameter λ that is varied throughout the fitting process:

- When λ **is large**, the algorithm results in a gradient descent update.
- When λ **is small**, the algorithm results in a Gauss-Newton update.

To understand this variation, a contextualization of both methods must be provided:
The **gradient descent method**

2.2.4.2 Bayesian Regularization Algorithm

The main advantage of this algorithm, and the main reasoning for using it in this work, is that the resultant models are robust and the validation stage is unnecessary [33]. The algorithm is based on Bayes' probability theorem [34]:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}, \quad (P(B) \neq 0) \tag{2.8}$$

The Bayesian Regularization Algorithm makes use of Equation 2.8 during the training stage in the regularization scheme. The use of this theorem yields a great advantage that is the shortening of the interactive procedure when opposed to unregularized ANNs [33].

2.2.4.3 Scaled Conjugate Gradient Algorithm

This algorithm is based on conjugate directions. It is similar to other conjugate gradient algorithms except for the fact that it does not perform a line search for each iteration [35]. This makes it less computationally expensive than its counterparts.

This method can train an ANN as long as its weights, net input and transfer functions have derivative functions [35].

2.2.5 Artificial Neural Network Performance

Finally, after training an Artificial Neural Network, a performance analysis must be conducted in order to see how correct its predictions are.

The MATLAB toolbox used in this work calculates the individual errors of each value in the dataset. The error matrix stores all the individual errors. If this error matrix is called E, the output matrix is called X and the prediction matrix is called Y, then:

$$E_i = Y_i - X_i \tag{2.9}$$

Because these errors are stored in a matrix, they can all be mapped to the corresponding values. This is useful in detecting intervals where the network is less precise.

2.3 Related Work

The study of AC Loss in superconductors is a field with high demand. This is especially true in configurations that make them suitable for use in power systems. For example, in [36], a study on different configurations of superconductive fault current limiter (SFCL) coils is made. The purpose of the study is to determine which configurations translate to the least amount of AC loss. This translates to a difference with this paper, as all configurations are of solenoidal coils as these are more effective in dissipating the tremendous amounts of fault energy in a short amount of time [36]. The study is, nonetheless, interesting as the subject matter is closely related to what is studied in this paper.

This article makes use of H formulation to accurately estimate the loss in the different configurations as using real superconductors would be a tremendously expensive endeavour. Using this numerical model ensures that the values will be very close to the real loss in each configuration.

This study concludes that the best configuration in minimizing AC loss in SCFL coils while still relatively easy to manufacture is to arrange half of the conductors vertically in one layer in one direction and the other half in another layer in the opposite direction. This ensures some cancellation of the magnetic field leading to smaller AC loss [36].

But while several studies have been made on AC loss in different HTS configurations, the main interest of this work is to have ANNs do this. This begs the question of relating this framework with superconductors.

Machine Learning has already been employed in several studies with superconductors as they help with the complicated properties of these materials [37–43]. The use of Artificial Intelligence models leads to a great time reduction when solving complex superconductor problems as, once trained, these models are much faster than the calculations required to solve said problems.

One field of superconductivity where Artificial Intelligence has been heavily employed is in predicting critical temperatures of new materials. The main focus of the research in this field is to search for materials that are superconductive at ambient temperatures. While these materials have not yet been found, the use of AI to find new materials beats the current method of experimenting with first-principle calculations [37].

In [37], for example, an Artificial Neural Network is combined with Decision Trees in hope for being able to predict critical temperatures of unseen materials. There are two types of material representation: one based in the molecular formulae and other based on crystal structures. This means that the first allows for exploration across the entire chemical composition space [37], while the second allows for property inference based on the structural properties. In this study, the molecular formula representation is used in conjunction with statistical properties of the given materials.

The properties used in material representation in this study are shown in Figure 2.10. An attribute matrix T is employed, in which each property is given one of six

statistical values ($T \in \mathbb{R}^{22\times6}$) [37]. To extract features from T, 32 convolutional kernels with size 1×6 are used. Feature extraction for each material is achieved through the use of a Convolutional Neural Network with two row-scanning convolutional layers, one fully convolutional layer and two fully connected layers. A schematic of this network is presented in Figure 2.11.

Atom Number MendeleevNumber AtomicWeight MeltingTemperature Period Table Column Period Table Row	CovalentRadius Electronegativity GSvolume_per atom GSbandgap GSmagmom SpaceGroupNumber
NsValence NpValence NdValence NfValence NValence	NsUnfilled NpUnfilled NdUnfilled NfUnfilled NUnfilled

Figure 2.10: Elemental attributes used in materials representation in [37]. [44]

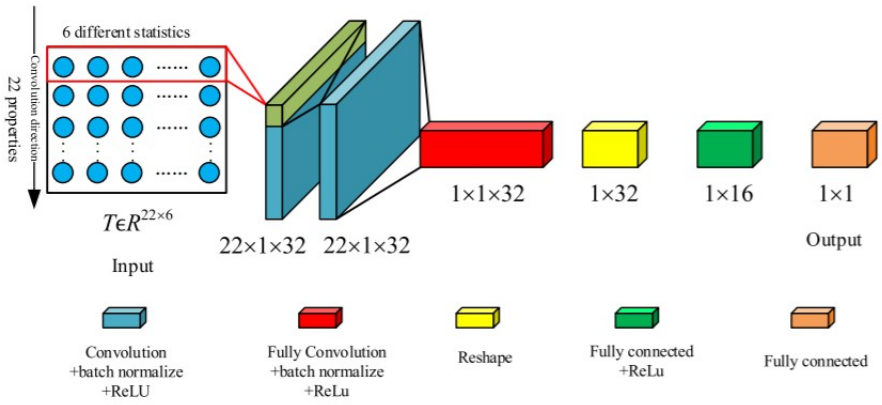

Figure 2.11: CNN model in [37]

A Gradient Boosting Decision Tree is then employed as to combine multiple learners into a strong one. This leads to a decrease in variance and deviation in the final model. This results in a more accurate prediction model, as evidenced by the results of the study. The conclusion is that this new model is able to cluster different types of superconductors together meaning that the network will know that the T_c for a given superconductor must be in the neighbourhood of superconductors of the same type [37].

Even though critical temperature is not dependant on configuration, it is still a factor that is mathematically difficult to achieve and Artificial Intelligence helps to solve this problem with very good results. If a Network is able to predict characteristics of unseen materials, then it must also be able to predict characteristics of a given configuration of superconducting material.

Another example of use of ANNs in predicting critical temperatures in superconductors is [38]. In this study, a new approach is used in predicting critical temperature on superconductors based on their chemical formulae. This new approach is named a variational bayesian neural network and leads to the following results: RMSE = 3.83, $R^2 = 0.94$.

Another field of superconductivity where Artificial Intelligence has been employed is in predicting certain characteristic curves for given superconductors. We have, in this area, a study that predicts the resistance-temperature of a superconducting film [39] and one that predicts the Current-Voltage curves of that same film [40].

The first of these articles, [39] uses various forms of machine learning for approximating resistance-temperature curves in superconductors. The method that is closer to the one used in this work is the shallow learning process that yields a MSE = $8.04 * 10^{-5}$. The second,[40] uses artificial intelligence to predict IV curves (Figure 2.12) for superconductors and yields a MSE = $2.3 * 10^{-8}$.

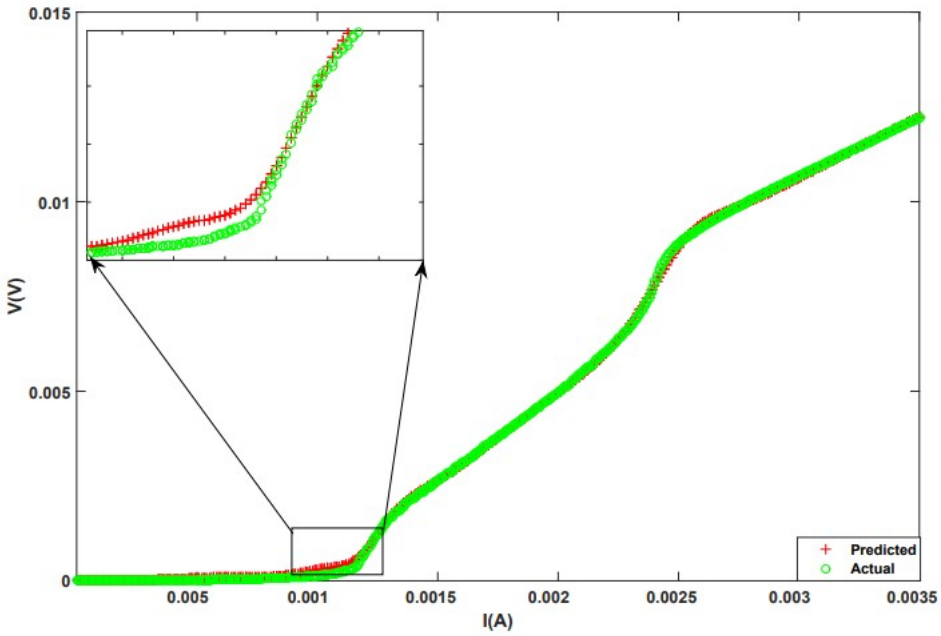

Figure 2.12: The predicted and cross validated IV curves for $T = 8.65$K and $H = 41$Oe in [40]

Artificial Intelligence has also been employed in superconductors to predict something different than properties. For example, in [41], a Neural Network that uses Backpropagation and a Genetic Algorithm is used in order to predict the optimal charging scheme of a flux pump.

Even though the subject of the study is not, in itself, superconductor in nature, it aims to be used in charging superconductive coils. These flux pumps are used to inject DC current into HTS tapes since they more thermal efficient than regular DC power supplies [41].

Research has already been conducted in using Neural Networks to predict AC loss in superconductors. Even though the aim of this work has not yet been implemented, that is, using neural networks for prediction of AC loss in different configurations of HTS tapes based solely on configuration parameters, some articles are close to this and are, therefore, worth tapping into.

For example, in [42], Artificial Neural Networks are used to predict non-sinusoidal AC loss in superconducting tapes. This is the study closest to the one presented in this document. For a feed-forward neural network, similar to what is used in this work, the best result of this study (3^{rd} harmonic) is RMSE = $4.0 * 10^{-3}$ and $R = 0.9999$.

Other studies to use Artificial Intelligence in predicting loss in superconductors are, for example, [43] that uses Artificial Neural Networks to predict the AC loss of a 150kJ Superconducting Magnetic Energy Storage System (SMES). In this study, two separate networks are developed and tested against 4 data points. The best of the networks, ANN2, yields an error below 4% for all data points.

While it is true that there are many more uses of Artificial Intelligence in matters relating to both power devices and superconductivity, these cannot be all enumerated in this work as it is virtually impossible to come across them all. It is, nonetheless, important to reference this fact as all contributions to any field are positive and deserve the acknowledgement.

To summarize this final Section, Table 2.1 presents the results for some of the studies presented. This serves as a benchmark as to what is to be expected as a possible outcome for this work.

Table 2.1: Article Result Comparison

Article	MSE	R^2
[38]	1.957	0.94
[39]	$8.04 * 10^{-5}$	N/A
[40]	$2.3 * 10^{-8}$	N/A
[42]	$1.6 * 10^{-5}$	0.99

3

Preparation and Setup

In this chapter, an insight on the preparation and setup used to achieve the results is given. The chapter is divided into four sections according to the workflow taken in its realization: **Dataset Acquisition and Preparation**, **Neural Network Training**, **Combination of Neural Networks** and **Network Performance Analysis**.

3.1 Dataset Acquisition and Preparation

The first stage of the work was to prepare the data to use in the training, validation and testing of both neural networks. There are three different types of data utilized in this work: **data from individual tapes**, **data from coils** and **data from stacks of coils**.

This data was gathered from applying numerical models to simulate different configurations of HTS coils. This means that, in this work, the final model actually predicts predicted values and not real ones. But, as the purpose of the work is to prove the ability of Neural Networks in predicting something as complex as AC Loss in HTS coils, predicting the values yielded from the use of numerical models or real ones is virtually the same. So, because of this, the use of models was employed instead of testing real HTS coils, portrayed in Figure 3.1, as this is considerably less costly.

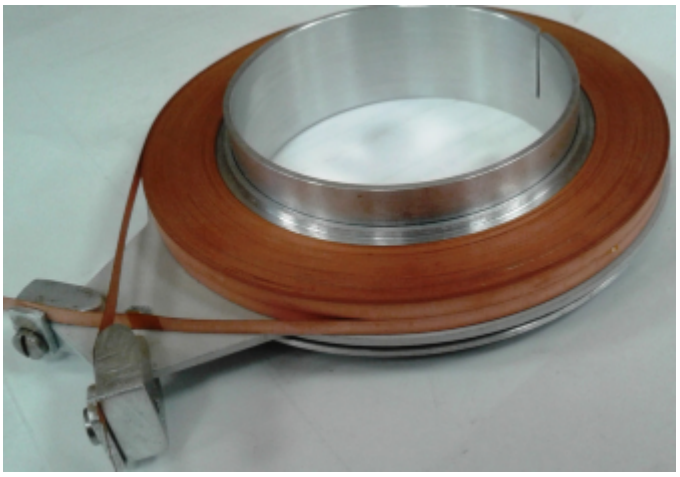

Figure 3.1: HTS pancake coil [45]

3.1.1 Data from individual tapes

The first type of data used is the loss in an individual piece of HTS tape. This type of data is used to characterize different tape variations as to form a loss model for each individual tape. This data was not fed into any Neural Network as its purpose was only to yield parameters that characterized the tape.

The aim of this data was: seeing that loss in an individual piece of tape took the form $Q(I) = A * I^B$(J/m/cycle), where A and B are constants, the conclusion was that these constants could be used in the characterization of the tape itself.

The data used in this was gathered from simulations of the given tape using numerical modelling. This model calculated the energy lost for very small time intervals for the duration of one wave cycle for different values of peak current. This yielded a spreadsheet that was then used to calculate the power lost during each cycle. To achieve this, the trapezoidal method of integration (Equation 3.1) was used. Figure 3.2 displays the power at each moment in a 1 meter strip of 4mm HTS tape at 10% of critical current at 50Hz.

$$\int_a^b f(x)dx = (a-b)\cdot\frac{1}{2}(f(a)+f(b)) \tag{3.1}$$

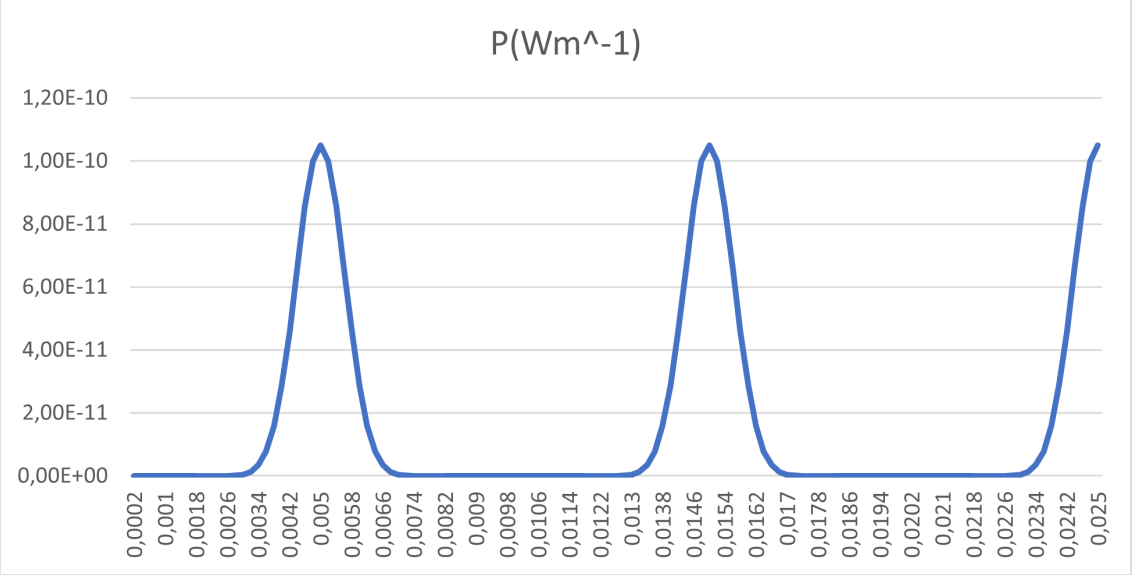

Figure 3.2: Power evolution with time in an HTS tape strip

Adding the power for each small time interval in a current cycle leads to total power in a cycle (Wm^{-1}/cycle) for a given current value. This procedure is repeated for different current values in order to be able to describe the loss power as a function of current. Figure 3.3 displays this plot for a 1 meter strip of 4mm HTS tape at 50Hz.

The plots like those of Figure 3.3 allow for the retrieval of both parameters (A and B). Having this two parameters that characterize the loss in each tape makes it possible to differentiate tapes with only these two parameters and not tape geometry. This means

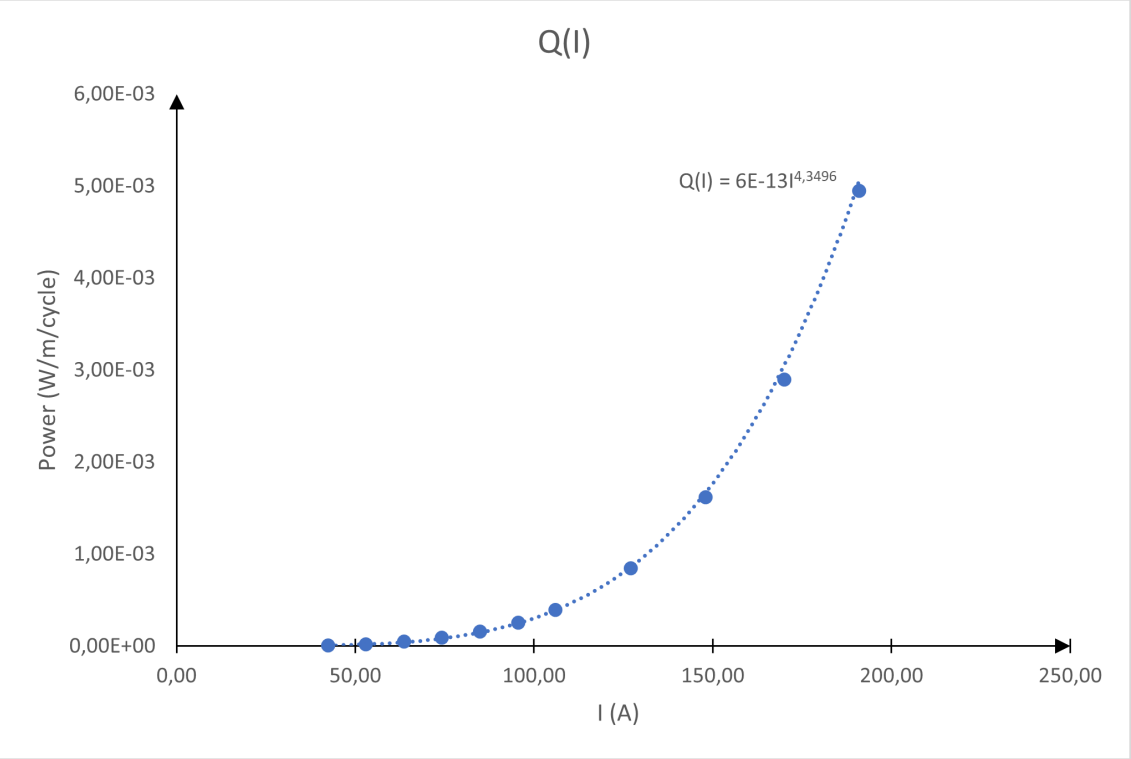

Figure 3.3: Loss power as a function of current

less inputs to the ANNs. Note that this reduction in characterization parameters does not translate to possible loss in information as each different tape has a different loss function and, therefore, different characterization parameters.

3.1.2 Data from coils

The second type of data utilized in this work is individual coil data. This data is used to train the first ANN in order to predict loss in an individual coil. For that, a dataset was built using the data divided as shown in Table 3.1.

Table 3.1: Inputs and Outputs of the single coil ANN

Input/Output	Name	Explanation
Input	A	One of the constants that define the HTS tape
Input	B	The other constant that defines the HTS tape
Input	d	The inner diameter of the coil
Input	N	The number of turns in the coil
Input	I	The current flowing through the coil
Output	Q	The AC loss in the coil

Note that the value of Q is given in J/cycle, meaning that the operating frequency does not have an impact on said value. To know the amount of energy lost in a period (T_d) of time it is crucial to know how many cycles of current happen in said period:

$$n_{\text{cycles}} = \frac{T_d}{1/f} \tag{3.2}$$

This dataset is comprised from multiple coil configurations of multiple different HTS tapes. The tapes, as stated in the previous Section, are characterized by the two constants A and B. This means that, in order for a new tape to be added to this dataset, it first need to be tested as a strip as to gather these parameters. The loss values used in this dataset are also gathered from simulations using numerical modelling.

This model is also ran for each different configuration for a single current wave period. The method used in calculating the loss is the same as before, meaning that the simulation yields the energy lost in small time intervals and the trapezoidal method of integration is used in order to calculate the power for each of these intervals. In the end, the different power values are added to result in the total power lost within a current cycle.

Figure 3.4 shows some points gathered from this dataset, for the same 12mm width HTS tape and an inner coil radius of 160mm. The graph show different colour regressions for different number of turns in each coil.

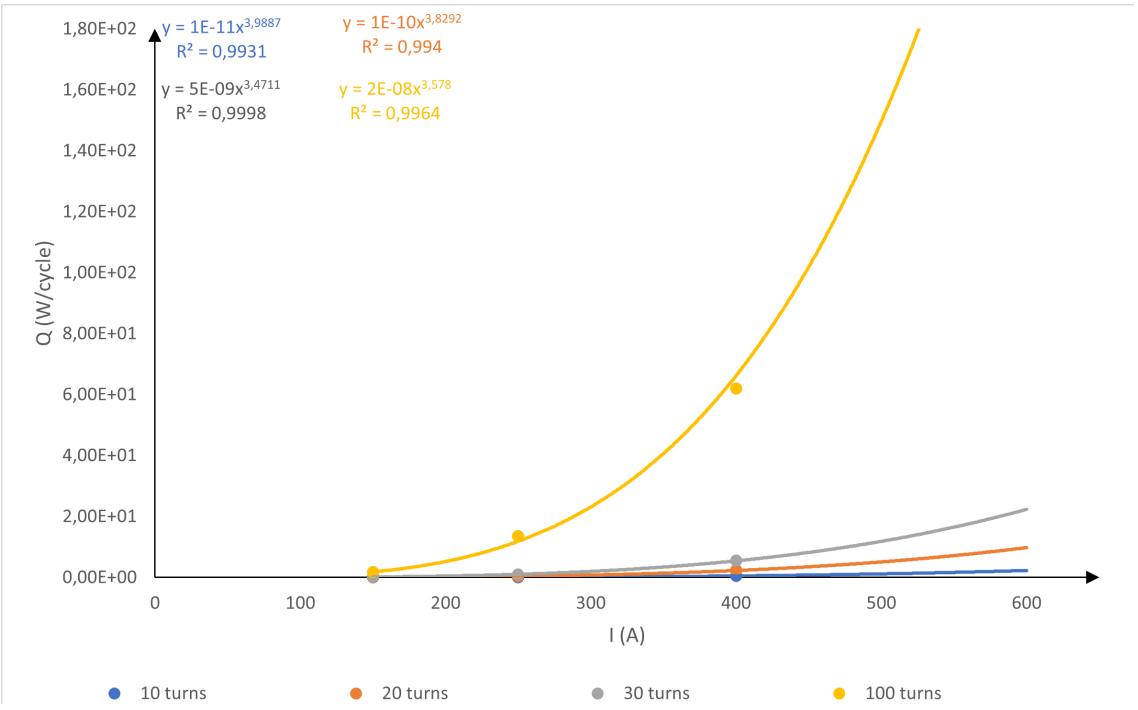

Figure 3.4: Loss variation with current and number of turns

The final dataset is presented as a table in Annex I.

3.1.3 Data from stacks of coils

This last type of data is used for training the ANN that handles prediction of loss in stacks of tapes. This data contains loss in various configurations of stacks, meaning, different

coils used and different number of coils. For each configuration, the data provides the number of coils in the stack, the loss in the first coil and the loss in the entire stack.

The purpose of having the loss in one single coil is to have the output from the previous network feed into this one as an input. This means that this second network has fewer inputs, as is portrayed in Table 3.2.

Table 3.2: Inputs and outputs of the coil stacks ANN

Input/Output	Name	Explanation
Input	Q_1	The AC loss value from one single coil
Input	n	The number of coils in a stack
Output	Q_t	The AC loss value from the entire stack

This dataset is also comprised of simulated loss values, achieved by modelling stacks of the preciously simulated coils. These simulations are more time consuming as they are modelling each coil individually for better results. This also means that, the greater the stack, the more time consuming the simulation becomes.

The process of transforming simulation data for this dataset is very similar to those before. The energy is converted to power through the means of trapezoidal integration, the main difference being that for each spreadsheet, this has to be done twice: once for the loss in a single coil and once for the total loss in the stack.

The final dataset is presented as a table in Annex II.

3.2 Neural Network Training

The second stage in this work is to prepare and train the different Artificial Neural Networks. As stated in Chapter 2, the tool used for handling ANNs is MATLAB's Neural Network Toolbox. Also, as briefly mentioned in Section 3.1, there are two different Neural Networks implemented in this work: **single coil AC loss prediction network** and **stack AC loss prediction network**. The process of training both is explained next.

3.2.1 Single coil AC loss Prediction Network

The dataset used in training the first Neural Network is the one explained in Section 3.1.2. This dataset has 5 inputs and 1 output. The Neural Network Toolbox has a Supervised Learning-dedicated Network tool that is used to implement the network.

The implemented network has two layers: a hidden layer with 25 artificial neurons and finally an output layer with one artificial neuron corresponding to the number of outputs. A schematic of this ANN is portrayed in Figure 3.5.

The number of neurons in the hidden layer was defined through a process of trial and error. Having a greater number of artificial neurons in this layer had a very similar or worse outcome to this number while having a greater training time.

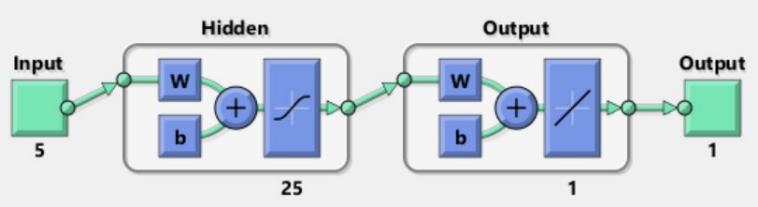

Figure 3.5: Schematic of the single coil ANN

This network is trained using the Bayesian Regularization algorithm present in the Neural Network Toolbox. This allows for a smaller validation dataset making it possible to use a greater training dataset. This is useful as the available data is limited.

The dataset subdivision will be: 85% to the training set, 5% to the validation set and 10% to the testing set. This means that this final 10% will not be used during the training stage and will only come into effect when testing the resultant network.

3.2.2 Stacks of coils AC loss Prediction Network

The dataset used for this second network is the one presented in Section 3.1.3. This dataset, however, is only comprised of three variables: number of coils (n_{coils}), loss in the first coil (Q_1) and total loss (Q_t). Of these three variables, two will be used as inputs: n_{coils} and Q_1; and the total loss as output. As stated previously, this dataset allow for taking the output from the previous network as an input for this one. This network connection will be explained further ahead.

This second ANN is also comprised of two distinct layers. There is one artificial neuron in the output layer corresponding to the number of outputs while hidden layer has 15 artificial neurons. The schematic representation of this network is portrayed in Figure 3.6.

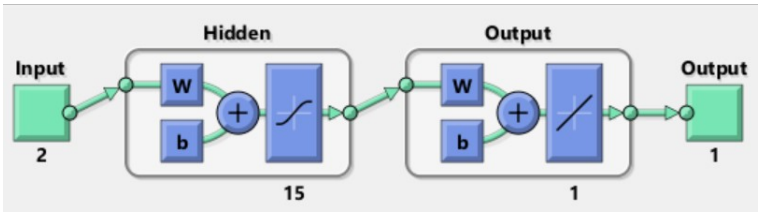

Figure 3.6: Schematic of the coils stacks ANN

The number of neurons in the hidden layer, as in Section 3.2.1, was defined through a process of trial and error. Having a greater number of artificial neurons in this layer had a very similar or worse outcome to this number while having a greater training time.

The training algorithm used for this second network is the same as the previous and the possibility for increased training dataset size comes even more handy here given that the stack data available is even thinner. Given that this dataset is smaller, it is expected that the prediction error will be greater that the previous but, hopefully, not much. Having small errors in both networks is fundamental as in its final form of having

both networks, the error will propagate meaning a greater deviation from the expected values.

Finally, in terms of dataset subdivision, the validation set will be kept at 5% but the testing set will be increased to 15% leading to a 80% training set. This increase in testing set sizes comes from the smaller dataset available creating the need for a smaller training set in order to have a large enough testing set.

3.3 Combination of Neural Networks

The final stage in this work is to combine both neural networks in order to predict loss in different configurations of HTS power devices. This is done by having a simple program that stores the necessary data and feeds it into the networks correctly. This program is essentially a state machine runing the different networks consecutively hence feeding the output of one as an input of the other.

To analyze the quality of these networks together, the intersection of both previous datasets will be used. To calculate the error from this combination of networks, the formula for error propagation will be used. As this is virtually a linear system with two error values, final error is the sum of both values.

3.4 Network performance analysis

In order to properly analyse if the results yielded by the network are good, there needs to be a way of validating these values. The MATLAB toolbox presents various tools to analyse the deviation of the values from the expected ones.

The first and most common way of analysing deviation in statistics is with the Mean Squared Error (MSE). The mathematical formula of calculating this error is presented in Equation 3.3.

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2 \tag{3.3}$$

Equation 3.3 says that the Mean Squared Error is the average of the squared difference between the actual value Y and the estimated one $\hat{Y}$. The squared difference is used to ensure that both positive and negative individual errors contribute to a positive increase in total error.

Another way of analysing the deviation is through the coefficient of correlation R. The formula for calculating this coefficient is given by Equation 3.4.

$$R = \frac{n * (\Sigma(X,Y) - (\Sigma(X) * \Sigma(Y)))}{\sqrt{(n * \Sigma(X^2) - \Sigma(X)^2) * (n * \Sigma(Y^2) - \Sigma(Y)^2)}} \tag{3.4}$$

This value is the correlation between the predictor variable X and response variable Y.

A more common value than the coefficient of correlation is its squared value R^2, known as the coefficient of determination. This is a widely used coefficient when studying the accuracy of prediction models. This is because this value measures how well the model replicates the studied data. Not counting it being the squared value of the coefficient of correlation, the formula for calculating the value of R^2 is provided in Equation 3.5.

$$R^2 = 1 - \frac{\sum_i^n (x_i - y_i)^2}{\sum_i^n (x_i - \bar{x})^2} \tag{3.5}$$

where n is the number of elements in the dataset, x is the dataset value, y is the predicted value and $\bar{x}$ is the mean value of all dataset values.

The MATLAB Neural Network Toolbox also presents a great tool for evaluating the accuracy of the prediction model in the form of an Error Histogram. This tool calculates the error of each individual predictions and groups this errors in small intervals. Then, the histogram displays different bars for each interval. Having the larger histogram closer to 0 means a better prediction. A good prediction, not only has the larger error bars closer to 0, but also forms a Gaussian distribution shape.

The performance analysis will be done for both Neural Networks. But there needs to be a way of analysing the performance of the combination of both networks. As there is no dataset containing data for this final model, the individual error cannot be computed. For this, we can make use of the individual correlation coefficients. For combinations of linear models, the coefficient of correlation of this combination will be the product of the individual coefficients. This is true for the coefficient of determination as well as $A^2 * B^2 = (A * B)^2$.

Despite not having the individual error values, there is the MSE of each model. So, even though there is no way of propagating the error of each individual observation, there is a possibility of approximating the combined MSE by adding the value of each model through Equation 3.6.

$$\mathrm{MSE}_{comb} \approx \mathrm{MSE}_A + \mathrm{MSE}_b \tag{3.6}$$

4

Results

In this chapter, the results from the training of both networks is presented, as well as the combination of both. A brief discussion of these results is also presented. This chapter follows the same division of the work, meaning that first are the results of the ANN for individual coils, than are the results of the ANN for stacks of coils. Finally, the last results to be presented are those of the combination of both networks.

For both Neural Networks, the Matlab Toolbox was used when validating results as the toolbox calculates the individual errors of each prediction. Another validation step is to compare the loss variation with current change to the theoretical variation. It is understood that, for superconductors, the AC Loss increases exponentially as the current approaches its critical value.

4.1 Individual coil ANN results

As stated previously, in this section are presented the results of the ANN for individual coils. The dataset used to train this network featured coils of two different tapes and of two distinct inner diameters (d = 80mm and d = 160mm). As such, the accuracy of the network cannot be verified for other types of tapes as there is not a great enough interval.

With this information in mind, Figures 4.1 and 4.2 depict a three-dimensional graph of AC loss as a function of current and number of turns.

As can be seen in the aforementioned Figures, the model reveals an exponential growth in AC loss with the increase in both current and number of turns, with a steeper exponential in terms of current. This coincides with the reality as AC loss increases exponentially with current as this tends to its critical value. It is also expected that the growth in loss is exponential in terms of number of turns since the amount of superconductive material does not increase linearly with number of turns (the radius for each turn is larger).

The result of the training process of this network is depicted in Figure 4.3. The training process was limited to 1000 Epochs by MatLab and it is visible that the minimum error in the training dataset was achieved in this final Epoch. The value of the error at

this stage is MSE = $2.21 * 10^{-6}$.

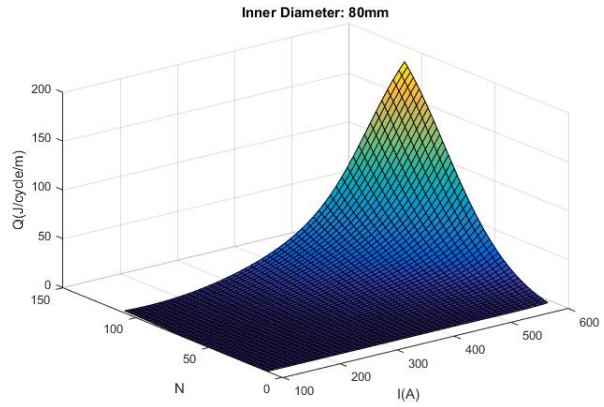

Figure 4.1: AC Loss for an individual coil with 80mm of inner diameter

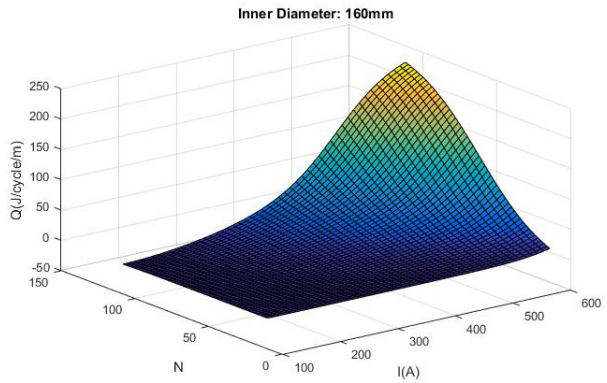

Figure 4.2: AC Loss for an individual coil with 160mm of inner diameter

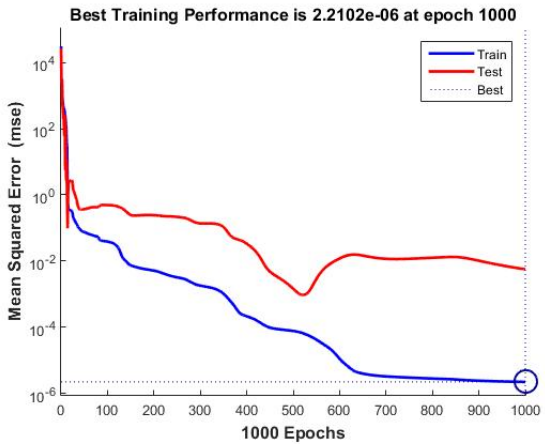

Figure 4.3: Network performance throughout the training process for the first ANN

In Figure 4.4, the error histogram of this network is presented. This histogram shows that, for the entire dataset, most of the error is extremely close to 0, while the greatest

deviation from the expected value is around 0.26. It is also apparent that the error of predictions with values from the training dataset is virtually 0.

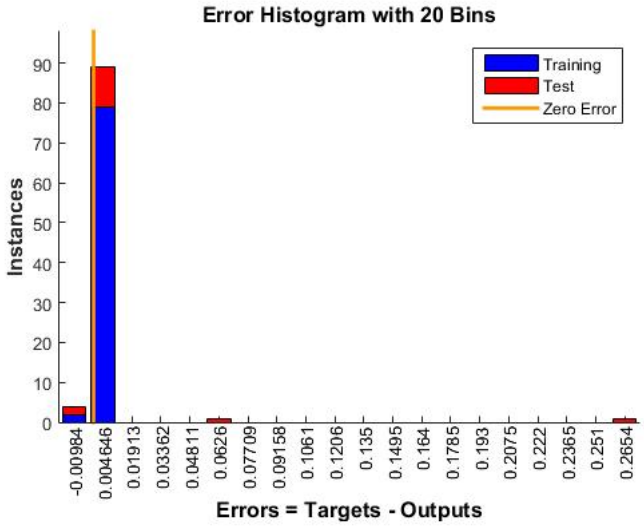

Figure 4.4: Error histogram for the individual coil ANN

Finally, in Figure 4.5, a depiction of the deviation of the actual values from the predictions is presented. This Figure contains three separate graphs: one that only depicts the deviation from the training dataset, one that depicts the deviation from the testing dataset and one that depicts the deviation from the entire dataset. Having said this, from the analysis of these graphs, it is inferable that the dataset values (presented as circles) fit almost perfectly in the prediction line (coloured line). This is also apparent from the coefficient of correlation being virtually 1 in all cases.

From the analysis of the results of this Neural Network, it is concluded that these values are very optimal as they are very close to the initial values. Having a small error in this model is also important as this error is carried onto the final one.

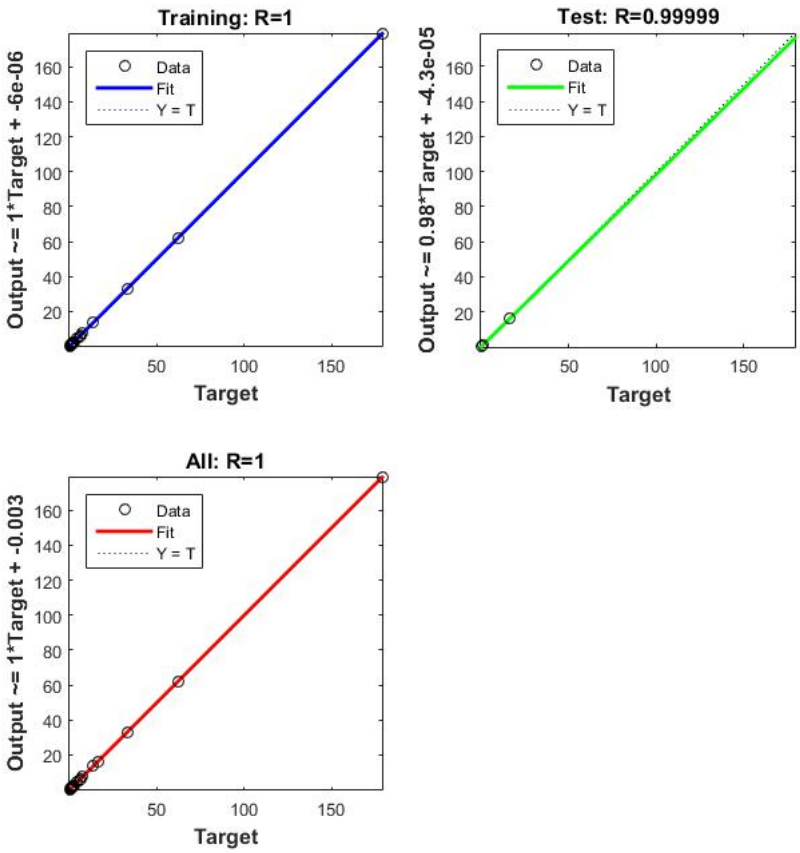

Figure 4.5: Error regression for the individual coil ANN

4.2 Stacks of coils ANN results

In this section, the results from the second neural network are presented. As stated in the previous Chapter, this second network is dedicated to predicting AC Loss values in stacks of coils when knowing the value of loss in one individual coil. While the dataset in itself contains information on the coils themselves, this information is not passed through as an input so that this information will not be required post-training.

The training of this second network resulted in a final MSE of $2.73 * 10^{-6}$. While this error is very small, it is important to note that the dataset used in the training of this network was considerably smaller. While this dataset contained multiple loss values for different configurations, only two configurations were available for a myriad of coils: stacks of 5 coils and stacks of 10 coils. This means that the predictions might be inaccurate for a number of stacks that greatly deviates from this values. The training performance for this network is portrayed in Figure 4.6.

Subsequently, in Figure 4.7 is portrayed the error hystogram that results from this second ANN. It is noticeable that most error values are extremely close to 0, while the

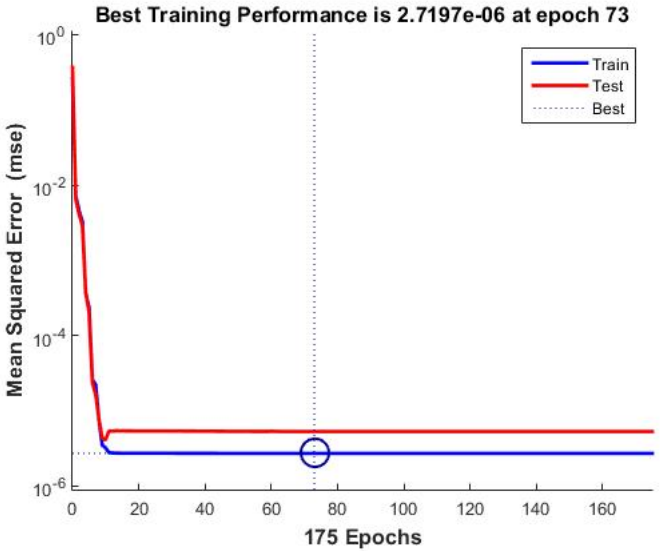

Figure 4.6: Network performance throughout the training process for the second ANN

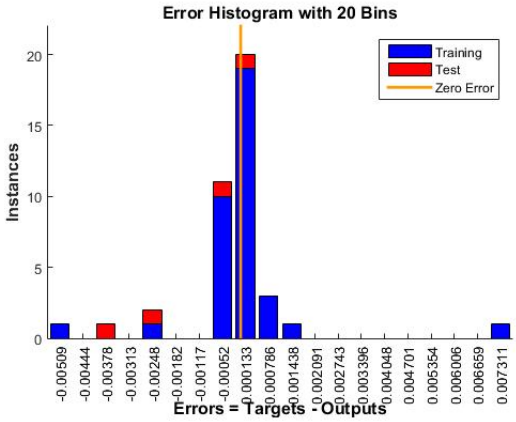

Figure 4.7: Error histogram for the coil stacks ANN

largest deviations from the actual values are from values contained in the training set. Even so, the greatest error is in the order of 10^{-3}. This small value is important since the smallest loss value present in the dataset is in the order of 10^{-2}. This translates to a noticeable but small deviation.

Finally, a depiction of the error regression for this network is displayed in Figure 4.8. It is noticeable that, in this network, the value of the coefficient of correlation is worse that the previous. The value of this coefficient is now 0.99983, meaning that the coefficient of determination becomes $0.99983^2 = 0.99966$.

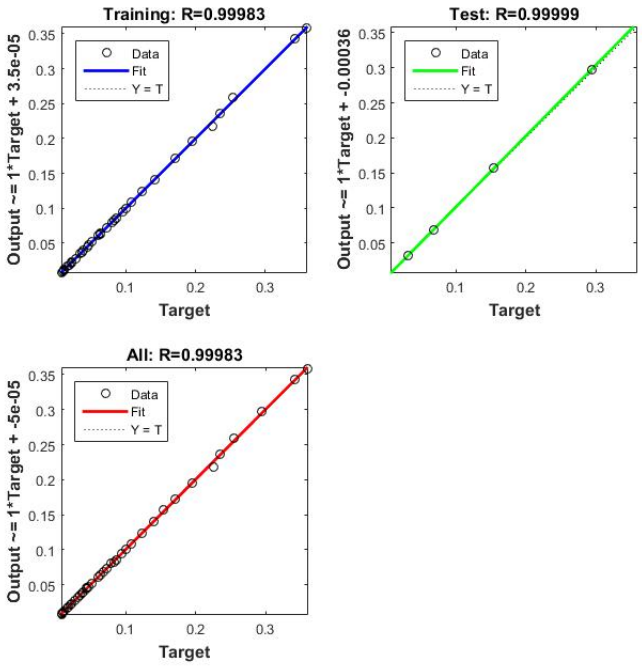

Figure 4.8: Error regression for the coil stacks ANN

4.3 Combination of Neural Networks Results

In this final section, the results of the combination of both previous sections are presented. Since the aim of this work is to predict AC loss in multiple configurations of HTS coils, these results are the most relevant for this study. As explained before, there is no way of computing the individual errors for each prediction since there is no dataset for this part.

Having said this, using the Equations presented in 3.4, the MSE for this combination will be:

$$\mathrm{MSE}_{comb} \approx \mathrm{MSE}_{4.1} + \mathrm{MSE}_{4.2} = 2.21 * 10^{-6} + 2.72 * 10^{-6} = 4.93 * 10^{-6} \tag{4.1}$$

As stated in 3.4, there is also a way of approximating the coefficient of correlation of this model by multiplying both coefficients:

$$\mathrm{R}_{comb} \approx \mathrm{R}_{4.1} * \mathrm{R}_{4.2} = 1 * 0.99983 = 0.99983 \tag{4.2}$$

Finally, having the value of coefficient of correlation allows for the calculation of the coefficient of determination:

$$\mathrm{R^2}_{comb} = \mathrm{R}^2_{comb} = 0.99983^2 = 0.99966 \tag{4.3}$$

4.4 Result Comparison

Having provided the individual results of each model in the previous sections, now is time to present them side by side. This allows for an easier understanding of the entire model and where the biggest problems lie in said model. Table 4.1 illustrates this side by side display of results.

Table 4.1: Result Comparison

Value	Individual Coil	Stacks of Coils	Combination
MSE	$2.21 * 10^{-6}$	$2.72 * 10^{-6}$	$\approx 4.93 * 10^{-6}$
R	≈ 1	0.99983	≈ 0.99983
R^2	≈ 1	0.99966	≈ 0.99966

As is visible in Table 4.1, the Mean Squared Error is greater in the second network. This is due to the smaller dataset used in training this network. This also results in worse approximations ($R, R^2 < 1$).

As the values in the combination of both networks are dependant on the individual ones, having worse individual values will result in worse combinations values. This is also illustrated in Table 4.1, as the Mean Squared Error of the combination is much greater than the individual MSE's. As the approximation quality values for the first network are virtually 1, then the results for the combination of both are the same as for the second network.

5

Conclusion

The two main focus points on employing ANNs on AC Loss in superconductive devices was to be able to predict loss in different configurations of said devices in a much faster manner. While in no way have all possible configurations been studied, it is safe to conclude that, for stacks of HTS coils, this aim has been met. Even though this main objective has achieved, several other complementary objectives were defined.

The first of these objectives was to find a way to be able to characterize HTS tapes in a simple mathematical fashion. This was achieved through finding the exponential regression that characterizes AC Loss in said tape and using its two constants as tape identifiers. It is also logical that these parameters will differ between different tapes as AC Loss behaves differently making these parameters simple and unique.

The second objective was to create a model able to predict AC loss in HTS coils given the configuration of said coils. This was achieved through the use of an Artificial Neural Network based model that takes into account every element in the coil's configuration. While, in this case, the model can be improved with further enlargement of the training dataset, the results are optimistic.

The next objective was to create a model that predicts AC loss in stacks of HTS coils, given the loss of an individual coil. While this model was implemented and behaves well for configurations close to those present in the dataset, further configurations need to be studied to perfect this model. This is because the present dataset is too small for the network to properly make predictions that stray too far from the ones studied.

The final objective was to combine both models as one. This has also been achieved but, because the second model is currently far from perfect, this one also is. It is understood that, to improve this combinations of models, the most important step is to enlarge the dataset used to train the stack of coils ANN as this is the main injuring factor in its performance.

As for contributions, both expectations have been met since both datasets that have been developed for this project contain the loss for vast configurations of HTS tapes and both ANN models have been implemented.

So, to summarize, while the objectives set out for this project have been met, further work needs to be put into increasing dataset size. Nonetheless, this work lays the ground for the idea that using ANNs to reduce time in estimating AC Loss in HTS devices is feasible with very accurate results.

Bibliography

[1] J. M. Lourenço. *The NOVAthesis LaTeX Template User's Manual*. NOVA University Lisbon. 2021. URL: https://github.com/joaomlourenco/novathesis/raw/master/template.pdf (cit. on p. ii).

[2] *Transforming Losses Calculation in High Temperature Superconducting Power Systems (tLOSS)*. URL: https://www.uninova.pt/project/transforming-losses-calculation-high-temperature-superconducting-power-systems-tloss (cit. on p. 2).

[3] H. K. Onnes. "Further experiments with liquid helium. G. On the electrical resistance of pure metals, etc. VI. On the sudden change of the rate atwhich the resistance of mercury disappears". In: *Commun.* 124c (1911) (cit. on p. 4).

[4] H. K. Onnes. *Nobel Prize Acceptance Speech*. 1913 (cit. on p. 4).

[5] J. Ceballos. "Analysis of AC Losses In Superconducting Electrical Components For Application In The Design Of Electrical Systems". In: (2010), p. 170 (cit. on pp. 4, 7, 9, 10).

[6] M. Boerhaave. *Archive Heike Kamerlingh Onnes, Research notebooks, inv. nr. 56 and 57* (cit. on p. 5).

[7] W. Meissner and R. Ochsenfeld. "Ein neuer Effekt bei Eintritt der Supraleitfähigkeit". In: *Die Naturwissenschaften* 21 (1933), pp. 787–788. ISSN: 00281042. DOI: 10.1007/BF01504252 (cit. on p. 5).

[8] A. Abrikosov. "On the Magnetic Properties of Superconductors of the Second Group". In: *Journal of Experimental and Theoretical Physics* 6.5 (1957), p. 1174 (cit. on p. 5).

[9] J. G. Bednorz and K. A. Müller. "Possible high Tc superconductivity in the Ba-La-Cu-O". In: *Zeitschrift für Physik B Condensed Matter* 64.2 (1986), pp. 189–193 (cit. on p. 7).

[10] Z. Q. Jiang and J. X. Jin. "Critical Current Measurement and Experimental Comparison of 1 G and 2 G HTS Tapes". In: *Proceedings of 2011 IEEE International Conference on Applied Superconductivity and Electromagnetic Devices*. 2011 (cit. on p. 7).

[11] N. Amaro. "Study of AC losses in medium-sized high temperature superconducting coils". In: (2015), p. 105. URL: https://run.unl.pt/handle/10362/16554 (cit. on pp. 7, 9, 10, 12, 13).

[12] H. Zueger. "630kVA high temperature superconducting transformer". In: *Cryogenics* 38.11 (1998), pp. 1169–1172. DOI: 10.1016/S0011-2275(98)00104-0 (cit. on p. 8).

[13] L. Xiao et al. "Development of the World's First HTS Power Substation". In: *IEEE Transactions on Applied Superconductivity* 22.3 (2012), pp. 5000104–5000104. DOI: 10.1109/TASC.2011.2176089 (cit. on p. 8).

[14] Y. K. Kwon et al. "Development of a 100hp Synchronous Motor With HTS Field Coils". In: *IEEE Transactions on Applied Superconductivity* 15.2 (2005), p. 2194.2197. DOI: 10.1109/TASC.2005.849610 (cit. on p. 8).

[15] Z. Huang et al. "Trial Test of a Bulk-Type Fully HTS Synchronous Motor". In: *IEEE Transactions on Applied Superconductivity* 24.3 (2014), pp. 1–5. DOI: 10.1109/TASC.2013.2296142 (cit. on p. 8).

[16] S. Chikazumi. *Physics of Ferromagnetism*. Oxford University Press, 1997. ISBN: 0-19-851776-9 (cit. on p. 9).

[17] F. Fiorillo and I. D. Mayergoyz. *Characterization and Measurement of Magnetic Materials*. Academic Press, 2004. ISBN: 978-0-012-257251-7. DOI: 10.1016/B978-0-12-257251-7.X5000-X (cit. on p. 10).

[18] C. P. Poole Jr. et al. *Superconductivity*. 2nd ed. 1995, p. 646. ISBN: 9780120887613 (cit. on p. 11).

[19] W. T. Norris. "Calculation of hysteresis losses in hard superconductors carrying ac: isolated conductors and edges of thin sheets." In: *Journal of Physics D: Applied Physics* 3.4 (1970), pp. 489–507 (cit. on p. 12).

[20] J. R. Clem, J. H. Claassen, and Y. Mawatari. "AC losses in a finite Z stack using an anisotropic homogeneous-medium approximation". In: *Superconductor Science and Technology* 20.12 (2007), pp. 1130–1139 (cit. on p. 13).

[21] E. Pardo. "Modeling of coated conductor pancake coils with a large number of turns". In: *Superconductor Science and Technology* 21.6 (2008), p. 065014 (cit. on p. 13).

[22] J. C. Maxwell. *A Treatise on Electricity and Magnetism*. Oxford University Press, 1873 (cit. on p. 13).

[23] B. Shen, F. Grilli, and T. Coombs. "Overview of H-formulation: a versatile tool for modelling electromagnetics in high-temperature superconductor applications". In: *IEEE Access* 8 (2020), pp. 100403–100414. DOI: 10.1109/ACCESS.2020.2996177 (cit. on pp. 13, 14).

[24] Merriam-Webster. *Neural Network*. 2020. URL: https://www.merriam-webster.com/dictionary/neural%20network (visited on 04/12/2020) (cit. on p. 15).

[25] M. T. Hagan et al. *Neural Network Design*. 2014, p. 1012. ISBN: 978-0-9717321-1-7 (cit. on p. 15).

[26] C. Salini et al. "Detection of critical diagnosis faults in automobiles using Convolutional Neural network architecture". In: *Materials Today: Proceedings* (2021). DOI: 10.1016/j.matpr.2021.02.167 (cit. on pp. 15, 16).

[27] N. Chowdhury and M. A. Kashem. "A comparative analysis of Feed-forward neural network & Recurrent Neural network to detect intrusion". In: *2008 International Conference in Electrical and Computer Engineering*. 2008, pp. 488–492. DOI: 10.1109/ICECE.2008.4769258 (cit. on pp. 16, 17).

[28] I. C. Education. *Supervised Learning*. 2020. URL: https://www.ibm.com/cloud/learn/supervised-learning (visited on 05/16/2021) (cit. on p. 18).

[29] I. C. Education. *Unsupervised Learning*. 2020. URL: https://www.ibm.com/cloud/learn/unsupervised-learning (visited on 05/16/2021) (cit. on p. 18).

[30] I. C. Education. *Machine Learning*. 2020. URL: https://www.ibm.com/cloud/learn/machine-learning%7B%5C#%7Dtoc-machine-le-SzgJbkmk (visited on 05/16/2021) (cit. on p. 18).

[31] M. T. Jones. *Models for machine learning*. 2017. URL: https://developer.ibm.com/articles/cc-models-machine-learning/%7B%5C#%7Dreinforcement-learning (visited on 05/16/2021) (cit. on p. 18).

[32] H. P. Gavin. "The Levenberg-Marquardt method for nonlinear least squares curve-fitting problems c ©". In: 2013 (cit. on p. 19).

[33] F. Burden and D. Winkler. "Bayesian Regularization of Neural Networks". In: *Artificial Neural Networks Methods and Applications*. Ed. by D. J. Livingstone. Humana Press, 2008, pp. 25–44. DOI: 10:1007/978-1-60327-101-1 (cit. on p. 19).

[34] R. Routledge. *Bayes's theorem*. 2018. URL: https://www.britannica.com/topic/Bayess-theorem (visited on 05/17/2021) (cit. on p. 19).

[35] L. Babani, S. Jadhav, and B. Chaudhari. "Scaled Conjugate Gradient Based Adaptive ANN Control for SVM-DTC Induction Motor Drive". In: *12th IFIP Internation Conference on Artificial Intelligence Applications and Innovations (AIAI)*. 2017, pp. 382–395. DOI: 10.1007/978-3-319-44944-9_33 (cit. on p. 20).

[36] W. Song et al. "AC Losses in Noninductive SFCL Solenoidal Coils Wound by Parallel Conductors". In: *IEEE Transactions on Applied Superconductivity* 30.8 (2020). DOI: 10.1109/TASC.2020.3021339 (cit. on p. 21).

[37] Y. Dan et al. "Computational Prediction of Critical Temperatures of Superconductors Based on Convolutional Gradient Boosting Decision Trees". In: *IEEE Access* 8 (2020). DOI: 10.1109/ACCESS.2020.2981874 (cit. on pp. 21, 22).

[38] T. D. Le et al. "Critical Temperature Prediction for a Superconductor: A Variational Bayesian Neural Network Approach". In: *IEEE Transactions on Applied Superconductivity* 30.4 (2020), pp. 1–5. DOI: 10.1109/TASC.2020.2971456 (cit. on pp. 21, 23, 24).

[39] T. Akram et al. "A novel framework for approximating resistance-temperature characteristics of a superconducting film based on artificial neural networks". In: *Results in Physics* 24.104088 (2021). DOI: 10.1016/j.rinp.2021.104088 (cit. on pp. 21, 23, 24).

[40] M. Kamram et al. "Prediction of IV curves for a superconducting thin film using artificial neural networks". In: *Superlattices and Microstructures* 95 (2016), pp. 88–94. DOI: 10.1016/j.spmi.2016.04.018 (cit. on pp. 21, 23, 24).

[41] J. Wei et al. "Optimizing the Charging Parameters of Linear Motor Flux Pump with BP Neural Network and Genetic Algorithm". In: *IEEE International Conference on Applied Superconductivity and Electromagnetic Devices*. 2020. DOI: 10.1109/ASEMD49065.2020.9276361 (cit. on pp. 21, 23).

[42] M. Yazdani-Asrami et al. "Prediction of Nonsinusoidal AC Loss of Superconducting Tapes Usinf Artificial Intelligence-Based Models". In: *IEEE Access* 8 (2020). DOI: 10.1109/ACCESS.2020.3037685 (cit. on pp. 21, 24).

[43] Z. Zhang et al. "AC Loss Prediction Model of a 150kJ HTS SMES Based on a Multi-Scale Model and Artificial Neural Networks". In: *IEEE Transactions on Magnetics* 54.11 (2018). DOI: 10.1109/TMAG.2018.2829711 (cit. on pp. 21, 24).

[44] L. Ward et al. "Matminer: An open source toolkit for accurate and interpretable prediction of material properties". In: *Computational Material Science* 152 (2018). DOI: 10.1016/j.commatsci.2018.05.018 (cit. on p. 22).

[45] E. P. Krasnoperov et al. "2G HTS tape and double pancake coil for cryogen-free superconducting magnet". In: *Electrical Engineering* 102 (2020), pp. 1769–1774 (cit. on p. 25).

I

Annex 1

A	B	N	d(mm)	I(A)	Q(J/$cycle$)
$6*10^{-3}$	4.2496	10	160	150	$1.53*10^{-2}$
$6*10^{-3}$	4.2496	10	160	250	$9.03*10^{-2}$
$6*10^{-3}$	4.2496	10	160	400	$5.45*10^{-1}$
$6*10^{-3}$	4.2496	20	160	150	$7.06*10^{-2}$
$6*10^{-3}$	4.2496	20	160	250	$4.15*10^{-1}$
$6*10^{-3}$	4.2496	20	160	400	2.35
$6*10^{-3}$	4.2496	30	160	150	$1.86*10^{-1}$
$6*10^{-3}$	4.2496	30	160	250	1.02
$6*10^{-3}$	4.2496	30	160	400	5.31
$6*10^{-3}$	4.2496	100	160	150	1.73
$6*10^{-3}$	4.2496	100	160	250	13.5
$6*10^{-3}$	4.2496	100	160	400	62.0
$6*10^{-3}$	4.2496	10	80	150	$7.13*10^{-3}$
$6*10^{-3}$	4.2496	10	80	250	$4.14*10^{-2}$
$6*10^{-3}$	4.2496	10	80	400	$2.51*10^{-1}$
$6*10^{-3}$	4.2496	10	80	600	1.89
$6*10^{-3}$	4.2496	20	80	150	$3.52*10^{-2}$
$6*10^{-3}$	4.2496	20	80	250	$1.95*10^{-1}$
$6*10^{-3}$	4.2496	20	80	400	1.10
$6*10^{-3}$	4.2496	20	80	600	7.45
$6*10^{-3}$	4.2496	30	80	150	$8.81*10^{-2}$
$6*10^{-3}$	4.2496	30	80	250	$4.80*10^{-1}$
$6*10^{-3}$	4.2496	30	80	400	2.62
$6*10^{-3}$	4.2496	30	80	600	16.7
$6*10^{-3}$	4.2496	100	80	150	1.23
$6*10^{-3}$	4.2496	100	80	250	6.22
$6*10^{-3}$	4.2496	100	80	400	33.1
$6*10^{-3}$	4.2496	100	80	600	179
$6*10^{-3}$	4.2496	10	80	100	$1.83*10^{-3}$

A	B	N	d(mm)	I(A)	Q(J/$cycle$)
$6*10^{-3}$	4.2496	10	80	120	$3.40*10^{-3}$
$6*10^{-3}$	4.2496	10	80	180	$1.30*10^{-2}$
$6*10^{-3}$	4.2496	10	80	200	$1.88*10^{-2}$
$6*10^{-3}$	4.2496	10	80	220	$2.63*10^{-2}$
$6*10^{-3}$	4.2496	10	80	300	$8.13*10^{-2}$
$6*10^{-3}$	4.2496	10	80	350	$1.47*10^{-1}$
$6*10^{-3}$	4.2496	20	80	100	$9.55*10^{-3}$
$6*10^{-3}$	4.2496	20	80	120	$1.72*10^{-2}$
$6*10^{-3}$	4.2496	20	80	180	$6.44*10^{-2}$
$6*10^{-3}$	4.2496	20	80	200	$9.12*10^{-2}$
$6*10^{-3}$	4.2496	20	80	220	$1.26*10^{-1}$
$6*10^{-3}$	4.2496	20	80	300	$3.72*10^{-1}$
$6*10^{-3}$	4.2496	20	80	350	$6.58*10^{-1}$
$6*10^{-3}$	4.2496	10	160	100	$3.87*10^{-3}$
$6*10^{-3}$	4.2496	10	160	120	$7.29*10^{-3}$
$6*10^{-3}$	4.2496	10	160	180	$2.84*10^{-2}$
$6*10^{-3}$	4.2496	10	160	200	$4.13*10^{-2}$
$6*10^{-3}$	4.2496	10	160	220	$5.74*10^{-2}$
$6*10^{-3}$	4.2496	10	160	300	$1.77*10^{-1}$
$6*10^{-3}$	4.2496	10	160	350	$3.19*10^{-1}$
$6*10^{-3}$	4.2496	20	80	100	$9.55*10^{-3}$
$6*10^{-3}$	4.2496	20	160	120	$3.71*10^{-2}$
$6*10^{-3}$	4.2496	20	160	180	$1.34*10^{-1}$
$6*10^{-3}$	4.2496	20	160	200	$1.96*10^{-1}$
$6*10^{-3}$	4.2496	20	160	220	$2.65*10^{-1}$
$6*10^{-3}$	4.2496	20	160	300	$7.96*10^{-1}$
$6*10^{-3}$	4.2496	20	160	350	1.41
$2.8*10^{-3}$	4.2069	10	80	90	$8.21*10^{-3}$
$2.8*10^{-3}$	4.2069	10	80	100	$1.20*10^{-2}$
$2.8*10^{-3}$	4.2069	10	80	110	$1.70*10^{-2}$
$2.8*10^{-3}$	4.2069	10	80	120	$2.35*10^{-2}$
$2.8*10^{-3}$	4.2069	10	80	130	$3.20*10^{-2}$
$2.8*10^{-3}$	4.2069	10	80	150	$5.77*10^{-2}$
$2.8*10^{-3}$	4.2069	10	80	160	$7.70*10^{-2}$
$2.8*10^{-3}$	4.2069	10	80	180	$1.32*10^{-1}$
$2.8*10^{-3}$	4.2069	20	80	90	$3.70*10^{-2}$
$2.8*10^{-3}$	4.2069	20	80	100	$5.21*10^{-2}$
$2.8*10^{-3}$	4.2069	20	80	110	$7.43*10^{-2}$
$2.8*10^{-3}$	4.2069	50	80	25	$4.35*10^{-3}$
$2.8*10^{-3}$	4.2069	50	80	35	$1.20*10^{-2}$

A	B	N	d(mm)	I(A)	Q(J/$cycle$)
$2.8*10^{-3}$	4.2069	50	80	50	$3.86*10^{-2}$
$2.8*10^{-3}$	4.2069	50	80	60	$6.81*10^{-2}$
$2.8*10^{-3}$	4.2069	50	80	75	$1.42*10^{-1}$
$2.8*10^{-3}$	4.2069	50	80	82	$1.90*10^{-1}$
$2.8*10^{-3}$	4.2069	50	80	90	$2.60*10^{-1}$
$2.8*10^{-3}$	4.2069	50	80	100	$3.71*10^{-1}$
$2.8*10^{-3}$	4.2069	54	80	50	$4.51*10^{-2}$
$2.8*10^{-3}$	4.2069	54	80	60	$8.08*10^{-2}$
$2.8*10^{-3}$	4.2069	54	80	75	$1.65*10^{-1}$
$2.8*10^{-3}$	4.2069	54	80	90	$3.03*10^{-1}$
$2.8*10^{-3}$	4.2069	54	80	100	$4.33*10^{-1}$
$2.8*10^{-3}$	4.2069	50	100	50	$4.83*10^{-2}$
$2.8*10^{-3}$	4.2069	50	100	60	$8.49*10^{-2}$
$2.8*10^{-3}$	4.2069	50	100	75	$1.77*10^{-1}$
$2.8*10^{-3}$	4.2069	50	100	90	$3.25*10^{-1}$
$2.8*10^{-3}$	4.2069	50	100	100	$4.63*10^{-1}$
$2.8*10^{-3}$	4.2069	54	100	50	$5.63*10^{-2}$
$2.8*10^{-3}$	4.2069	54	100	60	$1.01*10^{-1}$
$2.8*10^{-3}$	4.2069	54	100	75	$2.07*10^{-1}$
$2.8*10^{-3}$	4.2069	54	100	90	$3.78*10^{-1}$
$2.8*10^{-3}$	4.2069	60	100	100	$5.40*10^{-1}$
$2.8*10^{-3}$	4.2069	60	80	120	$9.40*10^{-1}$
$2.8*10^{-3}$	4.2069	60	80	130	1.24
$2.8*10^{-3}$	4.2069	60	80	150	2.09
$2.8*10^{-3}$	4.2069	60	80	160	2.66
$2.8*10^{-3}$	4.2069	60	80	180	4.26

Total Simulation Time: 20h 11min 12s

II

Annex 2

$Q_1(\mathrm{J}/cycle)$	n	$Q_t(\mathrm{J}/cycle)$
$2.46*10^{-3}$	5	$1.19*10^{-2}$
$2.36*10^{-3}$	5	$1.14*10^{-2}$
$2.05*10^{-3}$	5	$9.84*10^{-3}$
$1.78*10^{-3}$	5	$8.53*10^{-3}$
$4.63*10^{-3}$	5	$2.26*10^{-2}$
$4.46*10^{-3}$	5	$2.16*10^{-2}$
$3.86*10^{-3}$	5	$1.87*10^{-2}$
$3.35*10^{-3}$	5	$1.61*10^{-2}$
$7.96*10^{-3}$	5	$3.92*10^{-2}$
$7.62*10^{-3}$	5	$3.74*10^{-2}$
$6.59*10^{-3}$	5	$3.22*10^{-2}$
$5.71*10^{-3}$	5	$2.77*10^{-2}$
$1.28*10^{-2}$	5	$6.40*10^{-2}$
$1.23*10^{-2}$	5	$6.10*10^{-2}$
$1.05*10^{-2}$	5	$5.19*10^{-2}$
$9.10*10^{-3}$	5	$4.46*10^{-2}$
$1.99*10^{-2}$	5	$1.00*10^{-1}$
$1.89*10^{-2}$	5	$9.51*10^{-2}$
$1.61*10^{-2}$	5	$8.00*10^{-2}$
$1.38*10^{-2}$	5	$6.82*10^{-2}$
$4.55*10^{-3}$	10	$4.68*10^{-1}$
$4.38*10^{-3}$	10	$4.50*10^{-2}$
$3.86*10^{-3}$	10	$3.95*10^{-2}$
$3.41*10^{-3}$	10	$3.47*10^{-2}$
$8.31*10^{-3}$	10	$8.67*10^{-2}$
$8.01*10^{-3}$	10	$8.33*10^{-2}$
$7.06*10^{-3}$	10	$7.30*10^{-2}$
$6.23*10^{-3}$	10	$6.40*10^{-2}$
$1.48*10^{-2}$	10	$1.54*10^{-1}$

$Q_1(\mathrm{J}/cycle)$	n	$Q_t(\mathrm{J}/cycle)$
$1.34*10^{-2}$	10	$1.41*10^{-1}$
$1.18*10^{-2}$	10	$1.23*10^{-1}$
$1.04*10^{-2}$	10	$1.08*10^{-1}$
$2.18*10^{-2}$	10	$2.35*10^{-1}$
$2.02*10^{-2}$	10	$2.25*10^{-1}$
$1.83*10^{-2}$	10	$1.95*10^{-1}$
$1.62*10^{-2}$	10	$1.70*10^{-1}$
$3.27*10^{-2}$	10	$3.59*10^{-1}$
$3.13*10^{-2}$	10	$3.42*10^{-1}$
$2.73*10^{-2}$	10	$2.94*10^{-1}$
$2.39*10^{-2}$	10	$2.54*10^{-1}$