Institute of Electrical Engineering and Information Technology
Paderborn University
Department of Power Electronics and Electrical Drives
Prof. Dr.-Ing. Joachim Böcker

Student Project

Development of Reluctance Model, Core Material Database and GUI for FEM Magnetics Toolbox

by

Shobhit Ramkripal Sharma, Aniket Singh, Arjun Ambat

Student ID: 6898232, 6908616, 6921123

Supervisors: Till Piepenbrock, Nikolas Förster

Filing Date: December 9, 2022

Abstract

This project adds and updates the already developed FEM Magnetics Toolbox. A reluctance model for the 2D-axis symmetric inductor is designed using the Schwarz-Christoffel transformation. The motivation to implement it comes from the fact that it enables a fast and straightforward inductance calculation. The model is then used along with the FEM simulation to create a rapid and accurate application named *automated design* (in this project) to optimize the design process of the inductor. Subsequently, it plots a graph between core volume, total loss, and cost where the optimized designs are found on the Pareto front.

Further, a separate material database is implemented instead of the previous static data file. This database consists of the JSON data file and python files, which can access and process the information in the JSON file as per the input parameters of the FEM simulations. New material data can be easily updated using the defined structure of the JSON file. The connection between the database and FEMMT is implemented.

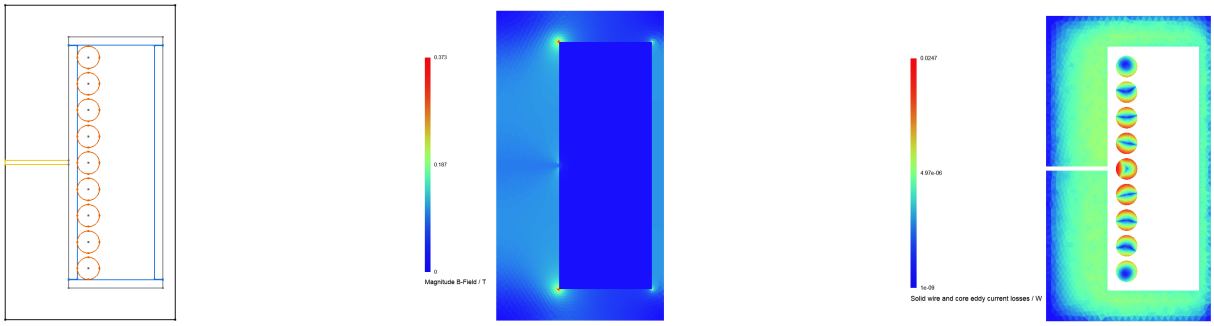
The Graphical User Interface (GUI) of the FEMMT is also updated. In GUI, the manual design tab is extended for transformer design, performing magnetic and thermal simulations. The *automated design* and core material database are integrated into GUI, making them easy for the user to design. The tab material database compares the materials available in the database with the help of different graphs. The datasheet and measurement reading of the materials can be compared, which can help in choosing the material for the simulation.

Contents

1 Introduction

A magnetic component (inductor or transformer), is an essential element of a power electronic device. Its optimum design is crucial for the proper working of the converter. Thus, it is imperative to have an accurate and realistic model during the design process. A way to do this is to simulate the magnetic component using the Finite Element Method (FEM). FEM is a numerical technique that provides easy and accurate modeling of complex geometrical shapes.

The Finite Element Method Magnetics Toolbox (FEMMT) is an open-source initiative for the simulation of power electronics' magnetic components. The toolbox is built using python programming, which interfaces with ONELAB (Open Numerical Engineering LABoratory [1]) to perform the FEM simulations.



(a) **(b)** **(c)**

Fig. 1.1: Inductor simulation result (PQ 40/40 core; 9 turns; center air-gap of 0.5 mm) with a) Geometry overview, b) Magnetic flux density (b), and c) Solid wire and eddy current losses

The toolbox simulates rotational symmetric core geometries. The reason is that most cores used in power electronics can be equivalently seen as rotational-symmetric.This specification converts a 3D problem into 2D and reduces the computation time with a minor effect on the model's accuracy. FEM simulation of the inductor, transformer, and integrated transformer has already been implemented in the toolbox. Moreover, it also

incorporates thermal simulation, making it much more substantial. Figure 1.1 shows one such simulation result of an inductor.

1.1 Task description

This project adds to the already developed toolbox. The features which are added or updated in this project can be broadly classified as follows,

- Reluctance model
- Core material database
- Automated design
- Graphic User Interface (GUI)

This report is divided into four chapters based on the classification described above. Chapter 2 is devoted to the inductor's reluctance model calculation and its implementation. It describes the theory of magnetic circuits and how it enables a fast and straightforward inductance calculation. The chapter also describes the limitation of the model and compares its result with that of FEM.

The upgradation of the core material database is explained in Chapter 3. It starts with a description of the previous version of the database system. The need for conversion from a static data file to a dynamic data file is stated, and the implementation process is documented. Furthermore, a separate material database repository has been created, which creates a dynamic data file every time simulation requires data from the database depending on the supplied input parameters. This chapter also describes the implementation of the connection between the solver and the database.

Automated design is the name of the application created in this project which is explained in Chapter 4. The reluctance model supplies a fast pre-calculation and filtration, whereas FEM simulation provides an accurate simulation of a 2D-axis symmetric inductor. The result is a graph between core volume, total losses, and cost consisting of all valid design cases. Where the user can find the optimum designs on the Pareto front.

Finally, Chapter 5 presents the GUI and its implementation. The manual design (single simulation) tab has been extended for transformers, performing magnetic and thermal simulations. Two new tabs are added which integrate the reluctance model and material database with the GUI. The 'automated design' (sweep simulation) tab is added for handling the application via GUI. Integration with the core material database is accomplished by the 'database' tab. It can be used to compare different core material properties. All the plots in it are interactive and are presented in a single window, making the comparison user-friendly.

2 Reluctance model

This chapter is intended to provide an insight of analytical magnetic circuit modeling for a 2D-axis symmetric inductor. The motivation and basic theory of the model is presented first. Subsequently, the calculations for core and air-gap reluctance are described. After that, the results obtained from the model are analysed and compared with FEM simulation results.

2.1 Overview

The motivation to design a reluctance model in the FEMMT comes from the fact that it enables a fast and straightforward inductance calculation. It also allows one to predict the flux density in each section of the core, thereby making it possible to avoid saturation of the core. These two features help to give approximately accurate feedback to the designer if there is any need for FEM simulation.

A magnetic circuit model, also named simply *reluctance model*, of an inductive component is analogous to an electric circuit. The reluctance is analogous to the electrical resistance and offers opposition to the flow of magnetic flux (ϕ). The analogies of the two circuits are summarized in the table 2.1. Furthermore, the reluctance is defined as $R_\mathrm{m} = \theta/\phi$, where θ is the magneto motive force and ϕ is the magnetic flux through the reluctance R_m. These analogies lead to the concept of magnetic circuits.

Electric circuit	**Magnetic circuit**
Electro-motive force (e.m.f.)	Magneto-motive force (m.m.f.)
Resistance (R)	Reluctance (R_m)
Current (I) = $\frac{e.m.f.}{R}$	Flux (ϕ) = $\frac{m.m.f.}{R_\mathrm{m}}$

Tab. 2.1: Analogies between electric and magnetic circuits

The inductance of an inductive component with N winding turns and a total magnetic reluctance $R_\mathrm{m,tot}$ is calculated as

$$L = \frac{N^2}{R_{\mathrm{m,tot}}}. \tag{2.1}$$

The reluctance of each section of the flux path has to be calculated first in order to calculate $R_{\mathrm{m,tot}}$. The calculation of the reluctance $R_{\mathrm{m}i}$ of the core sections is simple: for a core section of length l_{i}, cross-section A_{i}, and core material permeability $\mu_{\mathrm{r}}\mu_0$ it is

$$R_{\mathrm{m}i} = \frac{l_{\mathrm{i}}}{\mu_{\mathrm{r}}\mu_0 A_{\mathrm{i}}}. \tag{2.2}$$

Apart from core reluctance, there is also air-gap reluctance which is not so straightforward to calculate. Air-gaps are used in cores to get desired inductance value as air-gap length affects the reluctance of the magnetic circuit. Also from (2.1) and (2.2), it is observed that the inductance depends on the core geometry, material permeability, and number of turns. Thus, the above mentioned parameters can be put together to find the inductance of a magnetic component. A simple flow diagram representing the inputs and outputs of the implemented reluctance model is shown in Figure 2.1.

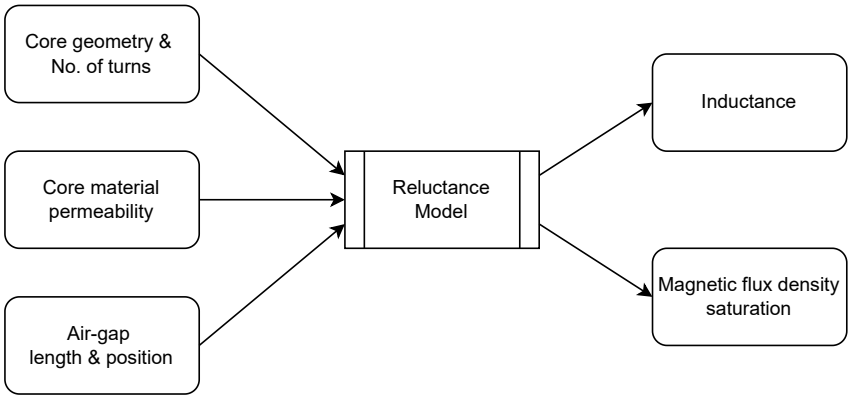


Fig. 2.1: Inputs and outputs of reluctance model

Magnetic flux in each section can also be easily calculated from the magnetic circuit ($\phi = \frac{\theta}{R_{\mathrm{m}}}$). Magnetic flux density is then given by

$$b = \frac{\phi}{A}. \tag{2.3}$$

The flux density obtained from Equation (2.3) is compared with the material B-H curve to find that if the magnetic flux in the given section is saturated or not.

2.2 Modeling

Before starting with the core reluctance, the core geometry that has been implemented in the FEMMT will be discussed. The 3D rotation symmetric magnetic core is simulated as

a 2D-axis symmetric core, as shown in Figure 2.2a, where in the 3D cylindrical geometry is represented by rectangular geometry in 2D if observed from side view. Figure 2.2b shows the 2D axis-symmetric core upon which the reluctance model has been designed. It requires only three parameters (*core_inner_diameter*, *window_h*, and *window_w*) to describe the complete geometry.

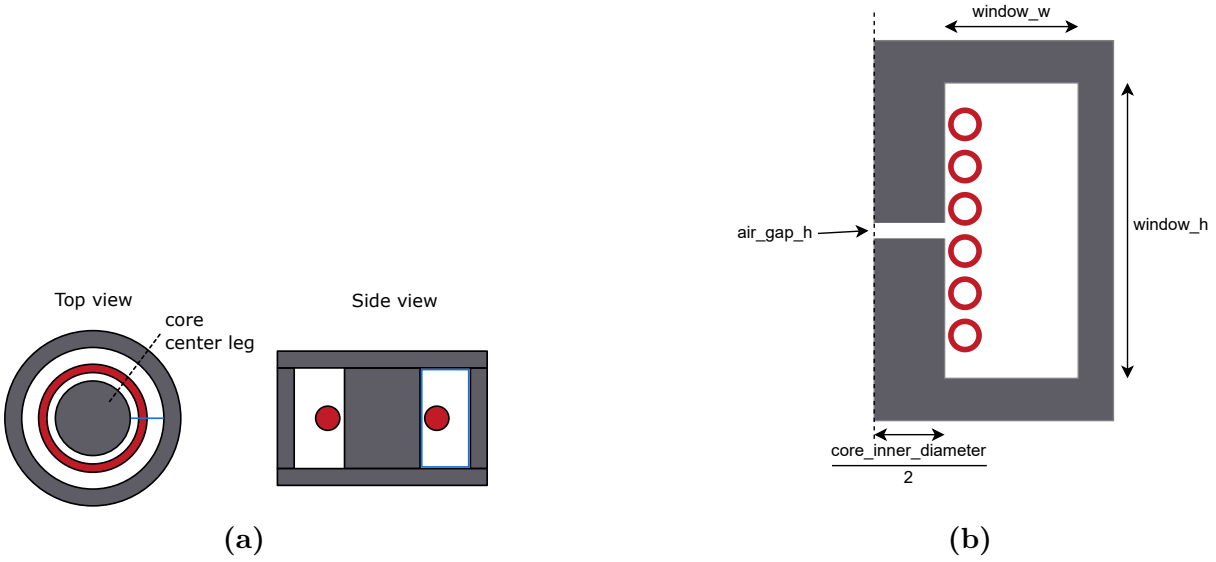


Fig. 2.2: core geometry (Rotation symmetric) with a) Top and side view, b) Parameters required to generate the 2D-axis symmetric geometry

2.2.1 Core Reluctance

The core reluctances have to be determined in order to achieve the complete reluctance model. The reluctance of a core section i can be calculated using the equation (2.2). Hence, for every section the magnetic path length and the cross-sectional area have to be calculated first. This is difficult for corner sections, as the flux tends to take shorter path, thus shortening the mean magnetic path.

The 2D axis-symmetric core is divided into five sections as shown in the Figure 2.3. Reluctance calculation for core section I and V are simple and straightforward. Core section III consists of the middle section where the cross-section area of the flux increases continuously. Therefore integration with respect to the radial component (r) is required (Figure 2.3). We get the differential equation using Equation (2.2)

$$dR_{\mathrm{m,III}} = \frac{dr}{\mu_{\mathrm{r}}\mu_0 2\pi r h} \tag{2.4}$$

and after defining the limits from r_1 to r_2, the integral equation becomes

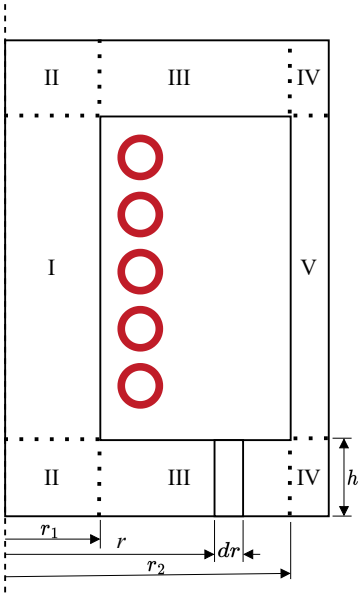


Fig. 2.3: Division of magnetic core into sections for reluctance calculation

$$R_{\mathrm{m,III}} = \int_{r_1}^{r_2} \frac{dr}{\mu_{\mathrm{r}}\mu_0 2\pi r h}. \tag{2.5}$$

Thus after the integration, we get the reluctance of core section III as

$$R_{\mathrm{m,III}} = \frac{1}{\mu_{\mathrm{r}}\mu_0 2\pi h} \ln \frac{r_1}{r_2}. \tag{2.6}$$

The accurate modeling of the corner reluctance (core section II and IV) is a bit complex, but it represent only a minor part of the core. Therefore, for this work the simple approximations from [2] have been taken. The mean path length of core section II (in Figure 2.3) is described by

$$l_{\mathrm{II}} = \frac{\pi}{8}(r_1 + h), \tag{2.7}$$

while the cross-section area is modified as per the cylindrical nature of the core, and is given by

$$A_{\mathrm{II}} = 2\pi r_1 \frac{(r_1 + h)}{2}. \tag{2.8}$$

Similarly, the reluctance for core section IV is calculated. Once the total core reluctance is determined, the air-gap reluctance is calculated.

2.2.2 Air-Gap Reluctance

The calculation of the air-gap reluctance in the project is carried out as per [3], which has been previously implemented by [4] for integrated transformer. This report pertains to the design of the reluctance model for only 2D axis-symmetric inductor, but at the same time can be extended for integrated transformer as a future scope.

Under the assumption of a homogeneous flux density distribution in the air gap and no fringing flux, the air gap reluctance can be calculated as

$$R_{\mathrm{m,g}} = \frac{l_{\mathrm{g}}}{\mu_0 A_{\mathrm{g}}} \tag{2.9}$$

where l_{g} and A_{g} are the air-gap length and air-gap cross-section area respectively, and μ_0 is the permeability of free space. Equation (2.9) is only accurate when the fringing flux is small compared to the total flux, i.e. when the air gap length is very small compared to the dimensions of the air gap cross-section.

The simple basic geometry from [3] is shown in Figure 2.4. It is taken as a basis to calculate more complex air gap structures. This basic geometry is used as a building block to describe different three dimensional air gap shapes.

The 2D reluctance of the geometry shown in Figure 2.4 is

$$R'_{\mathrm{basic}} = \frac{1}{\mu_0 \left[\frac{w}{2l} + \frac{2}{\pi}\left(1 + \ln\frac{\pi h}{4l}\right)\right]} \tag{2.10}$$

where the parameters are as illustrated in Figure 2.4. The 2D reluctance has the unit m/H and corresponds to the permeance per-unit length. The derivation of Equation (2.10) has been found in [5].

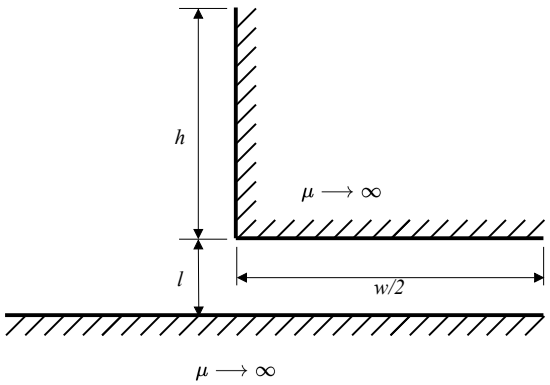


Fig. 2.4: Basic geometry for air gap calculation [3].

For the considered geometry of 2D axis-symmetric inductor, the air-gaps have been considered to be present only on the center leg. This results into only two types of air-gaps:

- Air-gap type 1: round-round cross-section (Figure 2.5a)
- Air-gap type 2: round-infinite cross-section (Figure 2.5b)

The two air-gap types are represented in Figure 4.2, where R' is the equivalent 2D reluctance of the given air-gap type, 'a' is the air-gap length, and 'r' is the radius of the center core leg.

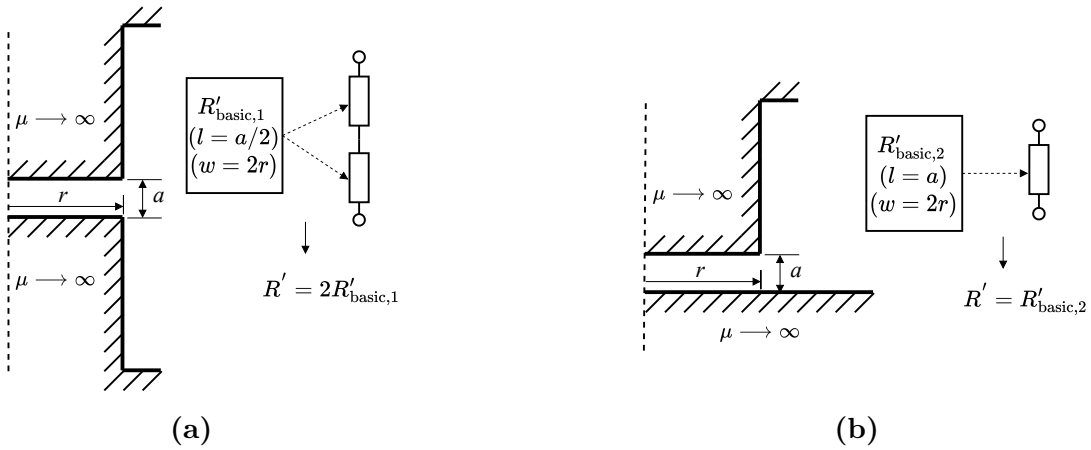


Fig. 2.5: Air-gap types possible on the center leg of the core, where a) Air-gap type 1 , b) Air-gap type 2 [3]

After that, a fringing factor is calculated which, as the name suggests, incorporates the fringing effect due to the air-gap. The fringing factor that considers fringing effects in r-direction is determined by calculating the corresponding 2D air-gap reluctance R' (as illustrated in Figure 4.2) and dividing it by the 2D reluctance that neglects any fringing effects. The fringing factor considering fringing effects in the r-direction (polar coordinate system) is

$$\sigma_{\mathrm{r}} = \frac{R'}{\frac{a}{\mu_0 r}}. \tag{2.11}$$

The fringing factor σ_r describes by which factor the air-gap reluctance decreases due to fringing flux in radial direction comparing to the idealized reluctance of (2.9). To derive the 3D fringing factor, σ_r is squared, giving the increased air-gap cross-sectional area due to fringing. The accurate reluctance of air-gap with circular cross-section can then be calculated as

$$R_{\mathrm{m,g}} = \sigma_r^2 \frac{a}{\mu_0 r^2 \pi}. \tag{2.12}$$

2.3 Simulation Results

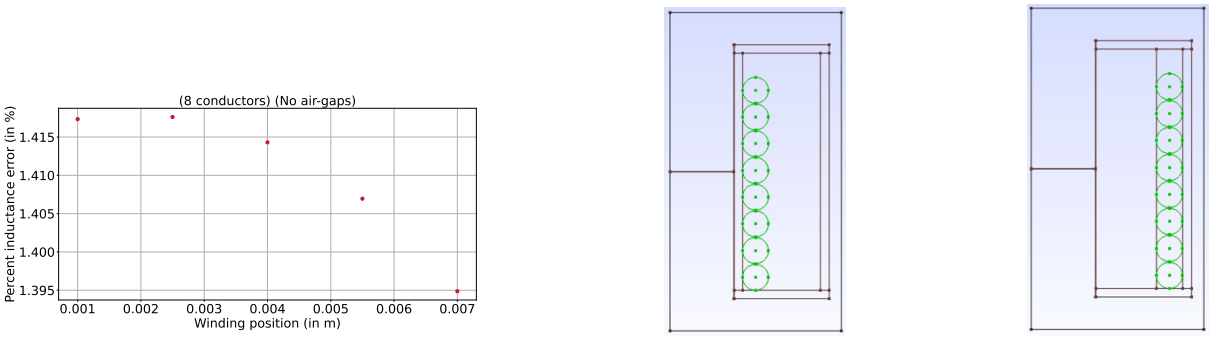
The designed reluctance model based on [3] is verified against the FEM simulation. The percent error of the calculated inductance with respect to FEM simulated inductance is given by

$$e_{\mathrm{ind}} = \frac{L_{\mathrm{cal}} - L_{\mathrm{FEM}}}{L_{\mathrm{FEM}}} \cdot 100. \tag{2.13}$$

The comparison is performed based on parameters such as winding position, number of turns, and winding type (litz or solid). This is because though the reluctance model is quite accurate tool, it does not consider the winding effect on the fringing field near the air-gaps. The core geometry selected for the purpose of simulation is PQ40/40 and N95 core material.

2.3.0.1 Inductance due to the Core

Figure 2.6 represents the plot between inductance percent error and winding position for a core without air-gaps. The winding consists of single turn solid conductor. It is clearly observed that the percent error stays near to 1%, showing that the calculations done for the core reluctance are quite accurate.



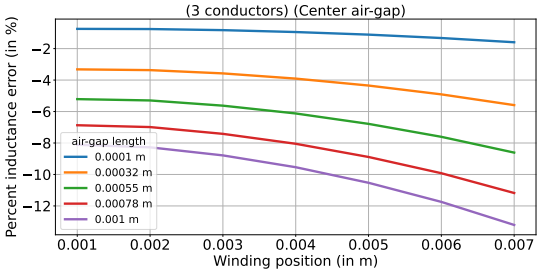
(a) **(b)** **(c)**

Fig. 2.6: a) Inductance percent error vs winding position, b) winding position: 0.001 m c) winding position: 0.007 m

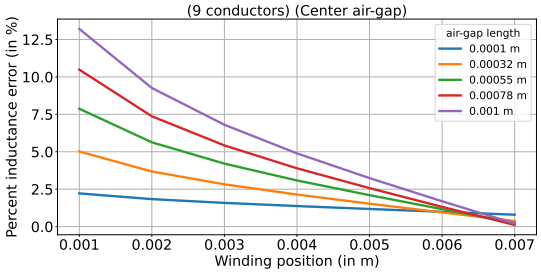
2.3.0.2 Inductance with Air-Gap (Solid Conductor Type)

The inductance percent error for a core with a center air-gap and solid conductor type is shown in Figure 2.7, where each subplot corresponds to different conductor number. The percent error increases with the increase of conductor number and also varies a lot with

respect to the winding position. In addition to that, the error increases with the increase of air-gap length.



(a) 3 conductors

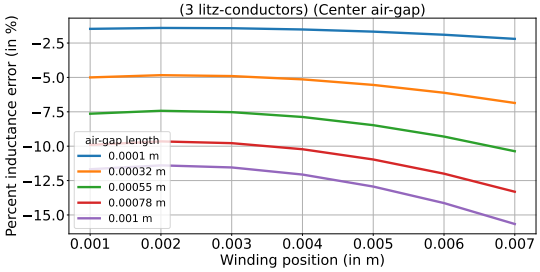


(b) 9 conductors

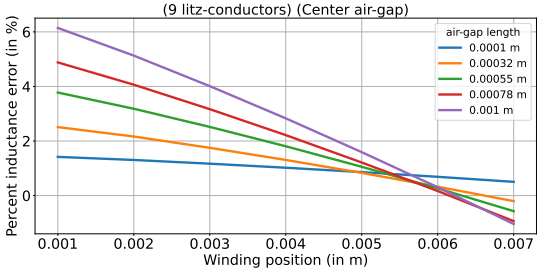
Fig. 2.7: PQ40/40 core (N95 material) with center air-gap and solid conductor type (conductor radius: 0.0013 m).

2.3.0.3 Inductance with Air-Gap(Litz Conductor Type)

With the same simulation parameters as that of the solid conductor type, Figure 2.8 shows the percent error for litz conductor type (with conductor radius = 0.0013 m; strand radius = 100e-6 m; strands number = 150; fill factor = None). The trend of percent error is opposite to that of solid conductor type, in the sense that it reduces with the increase of conductor number. The trend with respect to the winding position appears random. Where as the trend with regards to the air-gap length is same as that of the previous case (increase of percent error with increase in air-gap length). Overall the deviation of the inductance is less for litz than solid conductor type.



(a) 3 conductors



(b) 9 conductors

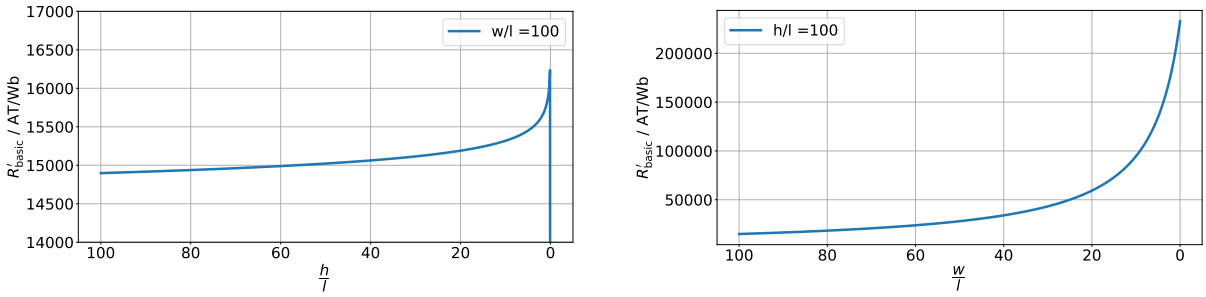
Fig. 2.8: PQ40/40 core (N95 material) with center air-gap and litz conductor type.

2.3.1 Model's Limitation

As it was shown in the previous subsection, that the deviation of the reluctance model is highly dependent on the external parameters (such as conductor number and winding

position), some constraints need to be placed so as to use the model. This subsection attempts to find the theoretical limitations with respects to the primary formula of (2.10). The expression of R'_{basic} is dependent on three variables, namely air-gap length (l), core leg width (w), and core leg height(h). But it can also be stated that it depends on two ratios, which is h/l and w/l. In this section, dependency of R'_{basic} on above mentioned ratios is analysed. Based on that, the usage constraint of the model is defined.

Figure 2.9a shows the effect of h/l on R'_{basic}. Where ratio h/l is varied, while w/l is kept constant. Similarly, Figure 2.9b shows the effect of w/l with h/l kept constant. The function exponentially increases as both the ratios independently approaches zero. The function even encounters singularity near zero in case of h/l.



(a) $\frac{w}{l} = 100$ **(b)** $\frac{h}{l} = 100$

Fig. 2.9: Change in R'_{basic} with respect to ratio a) (h/l) and b) (w/l)

The combined effect of the two (with five w/l ratio) is represented in the Figure 2.10. The y-axis of the graph is logarithmic so as to capture the complete curve. It is noted that the function is constant at higher ratios (bottom left point on blue curve), and becomes unstable as both ratios tends toward zero (top right point on violet curve).

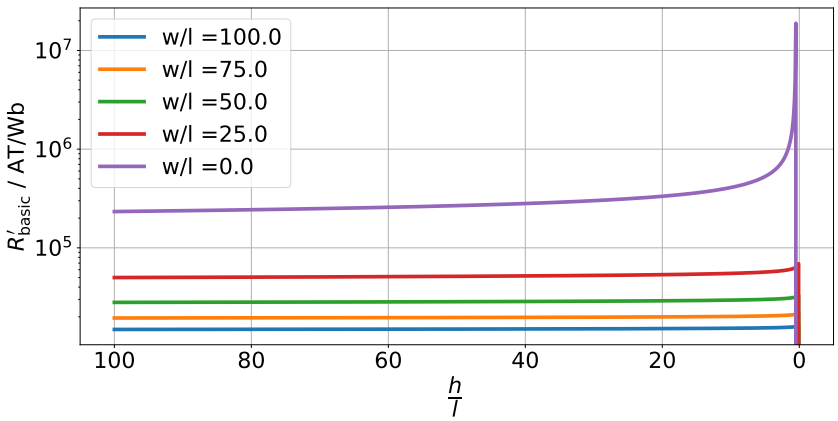


Fig. 2.10: Combined effect of ratio h/l and w/l on R'_{basic} (with logarithmic y-axis)

The results obtained in this and the previous sub-section are related to each other. The PQ40/40 core used for the simulation has $\frac{h}{l} = 59$ and $\frac{w}{l} = 29.8$ for a corner air-gap and an air-gap length of 0.5 mm. Now, as the air-gap length is increased, the ratios tend towards zero leading to reduced accuracy. Hence, it can be stated that the constraint defined in the previous sub-section depends not on the air-gap length, but on the ratio h/l and w/l. That said, the current designed reluctance model is constrained by the constant value of the air-gap length of 0.5 mm, rather than by the two ratios.

2.4 Summary

The designed reluctance model provides decent results for a 2D axis-symmetric inductor under the given constraints. The features of the model are summarised below

- From section 2.3, it is observed that maximum allowable single air-gap length should be maintained below 0.5 mm, so that percent inductance error remains below $\pm 10\%$.
- Air-gap with a h/l ratio less than 1 is solved as a corner air-gap.
- Multiple/Distributed air-gaps are calculated as superposition of single air-gaps positioned at different positions.
- Large air-gap length is realizable using distributed small air gaps positioned equidistant from each other.
- Maximum inductance error margin: $\pm 10\%$

As the reluctance model acts as a pre-filter or pre-check for narrowing down the FEM simulation design cases, the resulting high error margin still offers a large reduction in computation time. This will be apparent in Chapter 4 where the reluctance model is used for design optimization.

3 Material Database

This chapter provides information about Material Database, which is newly added to FEMMT. An overview of the previous version of the data file is presented first. After that, the structure of the database is described. Later, the magnetic loss coordination and connection of the database with the solver are explained.

3.1 Overview

In the previous version of FEMMT, the data of single material extracted from the datasheet was stored in a static text file which was used by the solver for core loss calculation. The task of storing more materials and different forms of data is accomplished by the creation of a material database repository which consists of data files and python files for processing the stored data.

This database stores data from different materials which are used by the toolbox during the calculation of core losses and reluctance model. It can also compare the data of different materials, which helps the user to choose between the materials available in the database for their simulations.

The python function in the database can process the data according to the loss approach opted by the user for the simulation. The functions include interpolation of data points between available points and creating a dynamic data file that can be accessed by the solver. The Connection between the FEM Magnetics Toolbox and the material database repository is represented in figure 3.1. The parameters material, temperature, frequency, and loss type are input for the database to look for required material data and return it in the desired form to the FEMMT.

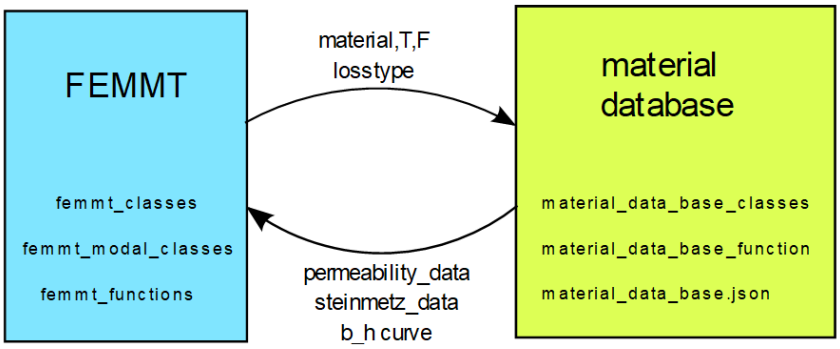


Fig. 3.1: Connection between FEMMT and material database

3.2 Structure of Database

The JSON file is used as a resource to store the data of different materials in a structured pattern. JSON files are easily readable, compact, and fast-processing files to be used in the database. Each material data is divided into two major sections based on the source of the data e.g manufacturer datasheet or measurements. The file structure is shown in the figure 3.2a. The database currently stores data of N95, N87, and N49 datasheets [6][7][8] and also measurement data from [9].

(a) File structure **(b)** JSON file

Fig. 3.2: Database .json-file structure

The data stored in the file are extracted from the datasheet of the materials and measurements. This JSON data file is read and processed with help of python scripts in

the repository to get the desired data as per set input parameters for the simulations in FEMMT.

3.3 Core Losses

The estimation of core losses is a very crucial task in the design of inductors, transformers, etc. In FEMMT, the core losses are either calculated by complex material parameters or with the empirical Steinmetz equation. The database consists of the information required by the FEMMT solver to estimate these losses. Both methods are presented 3.3.1 and 3.3.2.

3.3.1 Complex Core Parameter

The Complex core parameter is a frequency domain approach to determine the core losses. The complex permeability and permittivity are defined by $\underline{\mu} = \mu^{'} - \mathrm{j}\mu^{''}$ and $\underline{\epsilon} = \epsilon^{'} - \mathrm{j}\epsilon^{''}$. According to theorem of Poynting described in [10],

$$-\oint_{\delta V} (\underline{\boldsymbol{E}} \times \underline{\boldsymbol{H}}^{*}) \cdot \mathrm{d}\boldsymbol{S} = \int_{V} [(\kappa + \omega\epsilon^{''})|\underline{\boldsymbol{E}}|^2 + \omega\mu^{''}|\underline{\boldsymbol{H}}|^2] \cdot \mathrm{d}\boldsymbol{V} + \mathrm{j}\omega \int_{V} (\mu^{'}|\underline{\boldsymbol{H}}|^2 - \epsilon^{'}|\underline{\boldsymbol{E}}|^2) \cdot \mathrm{d}\boldsymbol{V}. \quad (3.1)$$

The eddy current loss can be expressed via

$$p_{\mathrm{eddy}} = -\omega \Im(\tilde{\underline{\epsilon}})|\underline{\boldsymbol{E}}|^2 \quad (3.2)$$

The hysteresis loss density can be expressed as

$$p_{\mathrm{hysteresis}} = -\omega \Im(\underline{\mu})|\underline{\boldsymbol{H}}|^2 \quad (3.3)$$

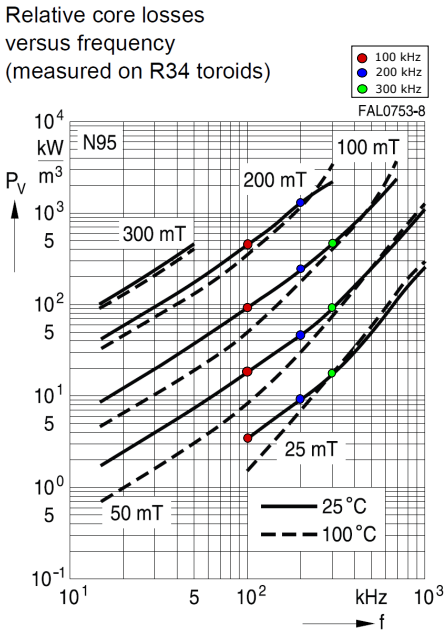
With both equations 3.2 and 3.3, the core losses can be estimated locally in the post-processing of the FEMMT simulation. The material parameter $\underline{\mu}$ and $\tilde{\underline{\epsilon}}$ are not given in the manufacturer datasheet. The permeability can be extracted via calculation on points from available core loss graphs provided in the datasheet as shown in figure 3.3. The complex permeability is calculated from the core loss at different flux points as seen in figure 3.3a using the equation 3.3 which is adjusted as shown in 3.4 [4]. The assumption is that all the losses shown in the graph are treated as hysteresis-loss(and eddy current loss is set to zero).

$$\begin{aligned} p_{\mathrm{hysteresis}} &= \frac{1}{2}\omega \Im(\underline{\mu}) \left(\frac{\underline{B}}{\underline{\mu}}\right)^2 \\ \Im(\underline{\mu}) &= \frac{p_{\mathrm{hysteresis}} \cdot \underline{\mu}^2 \cdot 2}{\omega \cdot \underline{B}^2} \end{aligned} \quad (3.4)$$

$$|\underline{\mu_r}|^2 = \mu^{'2} + \mu^{''2} \quad (3.5)$$

The imaginary part of permeability(μ'') obtained from 3.4 is used in 3.5 to calculate the real part of permeability(μ'), where $\underline{\mu_r}$ is the relative permeability of material from the datasheet. Thus, $\underline{\mu_r}$ is set to a constant value of 3000 for N95 from the datasheet[6]. This process is shown with an example of data obtained at 100 kHz and 25 mT represented by a green point in figure 3.3a. Solving equations 3.4 and 3.5 with $p_{\text{hysterisis}} = 4406.95\frac{\text{kW}}{\text{m}^3}$ and $\underline{\mu_r} = 3000$, we get $\mu' = 2993.0$ and $\mu'' = 202.0$ shown in figure 3.3b.

The data is stored at different values of temperatures and frequencies from the datasheet of N95. The format of data storage for complex permeability is shown in figure 3.3b.



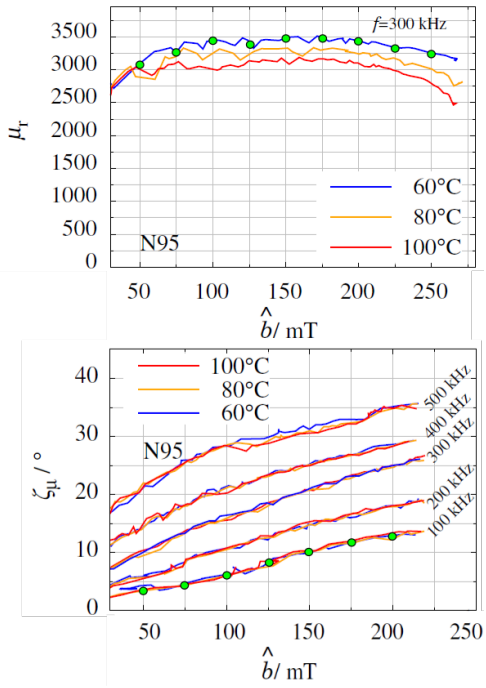
(a) Core loss graph in N95 datasheet[6]

(b) Data stored in JSON file

Fig. 3.3: Complex permeability data from material datasheet

The database also consists of the data obtained from measurements performed on different materials in the laboratory. The graph in figure 3.4a shows the measurements performed on N95 material by [9]. The data of amplitude and phase of the complex permeability of a material is extracted from the graph at different frequencies and temperatures and stored in JSON file as represented in figure 3.4b. The examples points of data are marked in green.

For the calculation of losses at particular values of temperature and frequency specified at the instance of simulation, the data is interpolated between available data in the database. The linear interpolation function is defined in python scripts of the material database, which processes the data and provides the values of the points lying between the known



(a) Graph from N95 measurements

```
"N95": {
  "Manufacturer": "Epcos",
  "manufacturer_datasheet": {"initial_permeability": 3000...},
  "measurements": [
    {"name": "ANSYS_extraction"...},
    {
     "data_type": "complex_permeability_data",
     "name": "Lukas Keuck",
     "company": "Paderborn University",
     "date": "xx.xx.xxxx",
     "test_setup_name": "Setup_Keuck",
     "test_setup_description": {"Toroid": "R24x28x5 Toroid"...},
     "data": [{"temperature": 30...},
     {
       "temperature": 30,
       "frequency": 100000,
       "b": [0.030469, 0.033157, 0.039973, 0.060137, 0.055713, 0.059741, 0.066848
       "mu_r": [[0.030594, 0.036927, 0.039828, 0.042145, 0.045301, 0.051149, 0.05
       "mu_phi_deg": [4.269, 4.7889, 4.9937, 4.8085, 5.7511, 6.065, 5.0982, 6.966
     },
     {"temperature": 30...},
     {"temperature": 30...},
     {"temperature": 30...},
     {"temperature": 60...},
     {"temperature": 60...},
```

(b) Measurement data stored in JSON file

Fig. 3.4: Complex permeability data from lab measurements

data points. The interpolated data is then written in the '.pro' extension text file which is readable by the 'General Environment for the Treatment of Discrete Problems'(GetDP) solver. This '.pro' file is a dynamic interpolated data file created whenever a simulation is run and material from the database is being used. The flow of connection between the FEMMT and material database repository is represented in figure 3.5. The parameter sets for the core during simulation is used to look through the database for the required information and processing on it is done by python scripts to create GetDP readable files for estimation of complex core parameter losses.

3.3.2 Steinmetz Equation

The steinmetz equation is an empirical equation to calculate power losses in magnetic materials when subjected to external varying magnetic flux. The power loss according to steinmetz equation is described with the three parameters k , a and b according to

$$\boldsymbol{P}_\mathrm{v} = \boldsymbol{k} \cdot \boldsymbol{f}^a \cdot \boldsymbol{B}^b \tag{3.6}$$

where $\boldsymbol{P}_\mathrm{v}$ is average power loss per unit volume, $\boldsymbol{f}$ is excitation frequency in kilohertz and $\boldsymbol{B}$ is the peak magnetic flux density. $\boldsymbol{k}, \boldsymbol{a}$ and $\boldsymbol{b}$ are Steinmetz's coefficients or material constants.

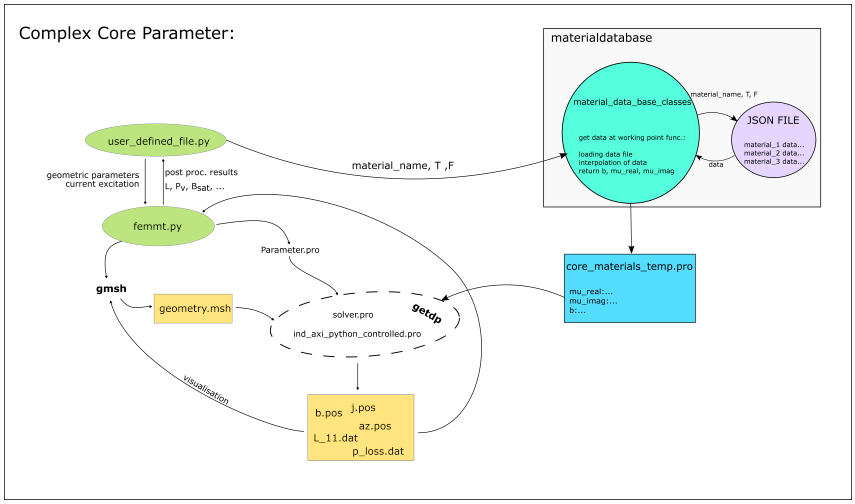


Fig. 3.5: Code structure of FEMMT with complex core parameter loss approach

The Steinmetz data is stored in the measurements section of the JSON file and distinguished depending on the source of material parameters. This data is accessed by the material database and sent to FEMMT when the loss approach type parameter for core loss calculation is set as Steinmetz. The python function looks in the database for selected material for simulation and returns the material parameters to FEMMT and then these parameters are updated into the "parameter.pro" file. The GetDP solver access the "parameter.pro" file for the material parameters for calculation of core loss using equation 3.6. The code structure for the Steinmetz loss approach is shown in figure 3.6.

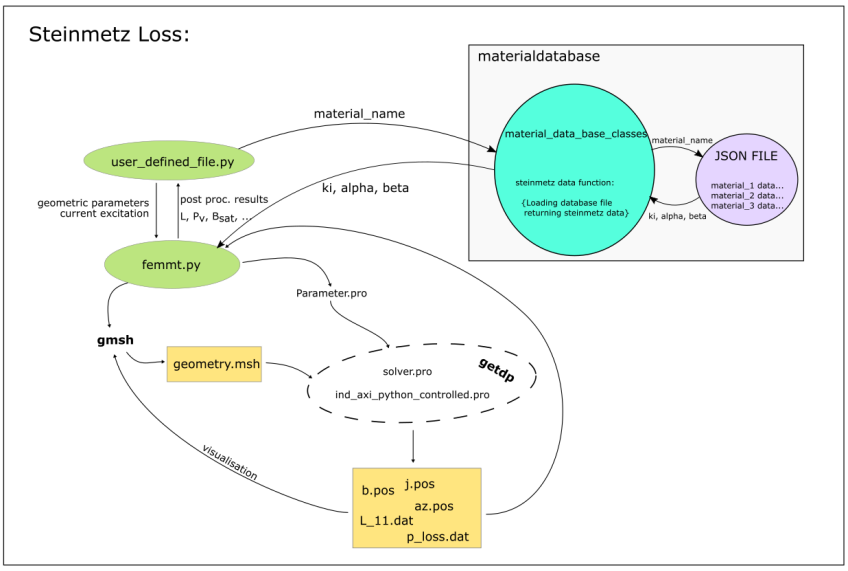


Fig. 3.6: Code structure of FEMMT with Steinmetz loss approach

3.4 Connection Between FEMMT and Database

The material database is implemented as a completely separate package which contains the data file and python files to access and process the data. This material database is used in FEMMT by installing it as a package in the main python file. The material database is initialized in core class of the FEMMT, where core model parameters such core dimensions, material, temperature, etc. are set for the simulation. After initializing as a variable, the python functions of the material database are accessible in FEMMT code, and required data can be accessed depending on simulation parameters as shown in listing 3.1.

```
import materialdatabase as mdb
# Initialize database
material_database = mdb.MaterialDatabase()
mu_rel = self.material_database.get_material_property(
    material_name="N95", property="initial_permeability")
steinmetz_data = self.material_database.get_steinmetz_data(
    material_name="N95", type="Steinmetz",
    datasource="measurements")
```

Listing 3.1: Initialization code for material database

During the FEM simulation if a material is to be used from the database, it can be initialized in core class definition with material name, loss approach type and data source (datasheet or measurement data) along with the other parameters as seen in listing 3.2. The frequency for simulation is taken from the excitation function of FEMMT.

```
core = fmt.Core(core_inner_diameter=core_db["core_inner_diameter"],
    window_w=core_db["window_w"],window_h=core_db["window_h"],
    material="N95", temperature=25, datasource="manufacturer_datasheet")
```

Listing 3.2: Initialization of core class

```
Include "Parameter.pro";
Function{
  b = {0, 0.025, 0.05, 0.1, 0.2, 0.3, 0.4, 1} ;
  mu_real = {3000.0, 2987.9333333333334, 2980.5366666666664, 2966.976, 2948.108, 2963.4946666666665, 1.0, 1.0} ;
  mu_imag = {1.0, 268.73, 340.86333333333334, 443.59333333333336, 551.0586666666667, 466.43066666666664, 466.43066666666664, 466.43066666666664} ;
  mu_imag_couples = ListAlt[b(), mu_imag()] ;
  mu_real_couples = ListAlt[b(), mu_real()] ;
  f_mu_imag_d[] = InterpolationLinear[Norm[$1]]{List[mu_imag_couples]};
  f_mu_real_d[] = InterpolationLinear[Norm[$1]]{List[mu_real_couples]};
  f_mu_imag[] = f_mu_imag_d[$1];
  f_mu_real[] = f_mu_real_d[$1];
}
```

Fig. 3.7: "core_materials_temp.pro" generated by the material database

These parameters are given as input to the database to get the required data for simulation as "core_materials_temp.pro" as shown in 3.7, which contains the interpolated material data is generated by the materialdatabase at a set temperature and frequency of simulation. In FEMMT, the interaction between the GetDP solver scripts and python is performed by

the file "Parameter.pro" (cf. figure 3.5), which contains all control variables and parameter values for the FEM simulation in the right format. The parameter values and control variables are set according to inputs of the user in "user _defined _file.py"(cf. figure 3.5). This ".pro" extension file is a non-python text file for the solver, which is rewritten with every simulation run. The main GetDP file "ind_axi _python _controlled.pro" internally includes all the solver and ".pro" files. Herein, "ind_axi _python _controlled.pro" contains formulas for both the loss approaches and overall problem description of FEM simulation. The overview of the connection between FEMMT and material database is shown in figures 3.5 and 3.6.

3.5 Magnetic Loss Coordination

This chapter focuses on magnetic loss coordination implemented in FEMMT, where material for the simulation can be from the database or can be defined by the user. This step is part of core class initialization, where the parameters of the core like material, core dimensions, and loss approach are set for the FEM simulation.

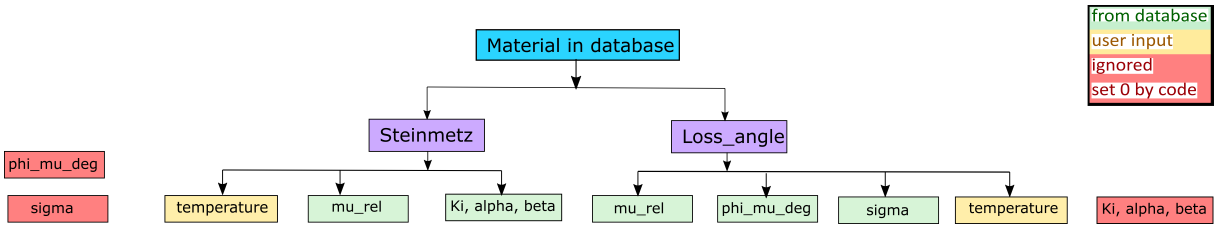


Fig. 3.8: Loss coordination for material in database

For material from the database and Steinmetz loss approach as Steinmetz, the relative permeability and Steinmetz's coefficient are accessed from the database by FEMMT python files using the database package, and temperature for the simulation is required as user input. The conductivity and imaginary part of the permeability of the material are ignored or set to zero by the code as it is not required by Steinmetz equation 3.6. For the complex core parameter or loss angle approach, the complex permeability and the conductivity of the material are looked up in the database, and the temperature and frequency for the simulation is required as user input to access the data of the material, which are stored as combination of different values.

There is also a possibility for entering custom material for the simulation, which is not available in the database. For the Steinmetz loss approach, the relative permeability and Steinmetz's coefficients of the materials are required as input from the user while initializing the core class. For the loss angle approach, the amplitude and phase of complex permeability and the conductivity of the material are required to run the simulation. The core class consists of the variable for the input of custom material properties as seen in listing 3.3.

```
class Core:
    def __init__(self, core_inner_diameter: float, window_w: float,
    window_h: float,
        material: str = "custom", loss_approach: LossApproach =
    LossApproach.LossAngle,
        mu_rel: float = None, temperature: float = None,datasource: str =
    None,
        steinmetz_parameter: list = None, generalized_steinmetz_parameter:
    list = None,
        phi_mu_deg: float = None,sigma: float = None, non_linear: bool =
    False,
        correct_outer_leg: bool = False, **kwargs)
```

Listing 3.3: Core class

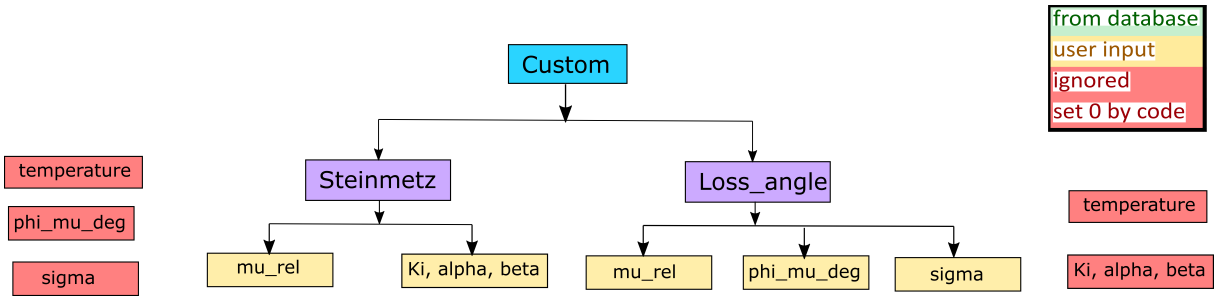


Fig. 3.9: Loss coordination for material based on user inputs

3.6 Process of Adding New Material Data

This section describes the process of adding new material to the material database. The JSON file is easy to read and can be updated with new data, which is divided into two sections, datasheet and measurement data for this project. The figure 3.10 explains the overall structure of the file, and the location of the graphs from the datasheet and measurement reading needs to be added which can be processed correctly by the database python scripts for integration with FEMMT and GUI. The datasheet graphs in figure 3.10 are used from N95 material datasheet [6] and measurement graphs are used from [9].

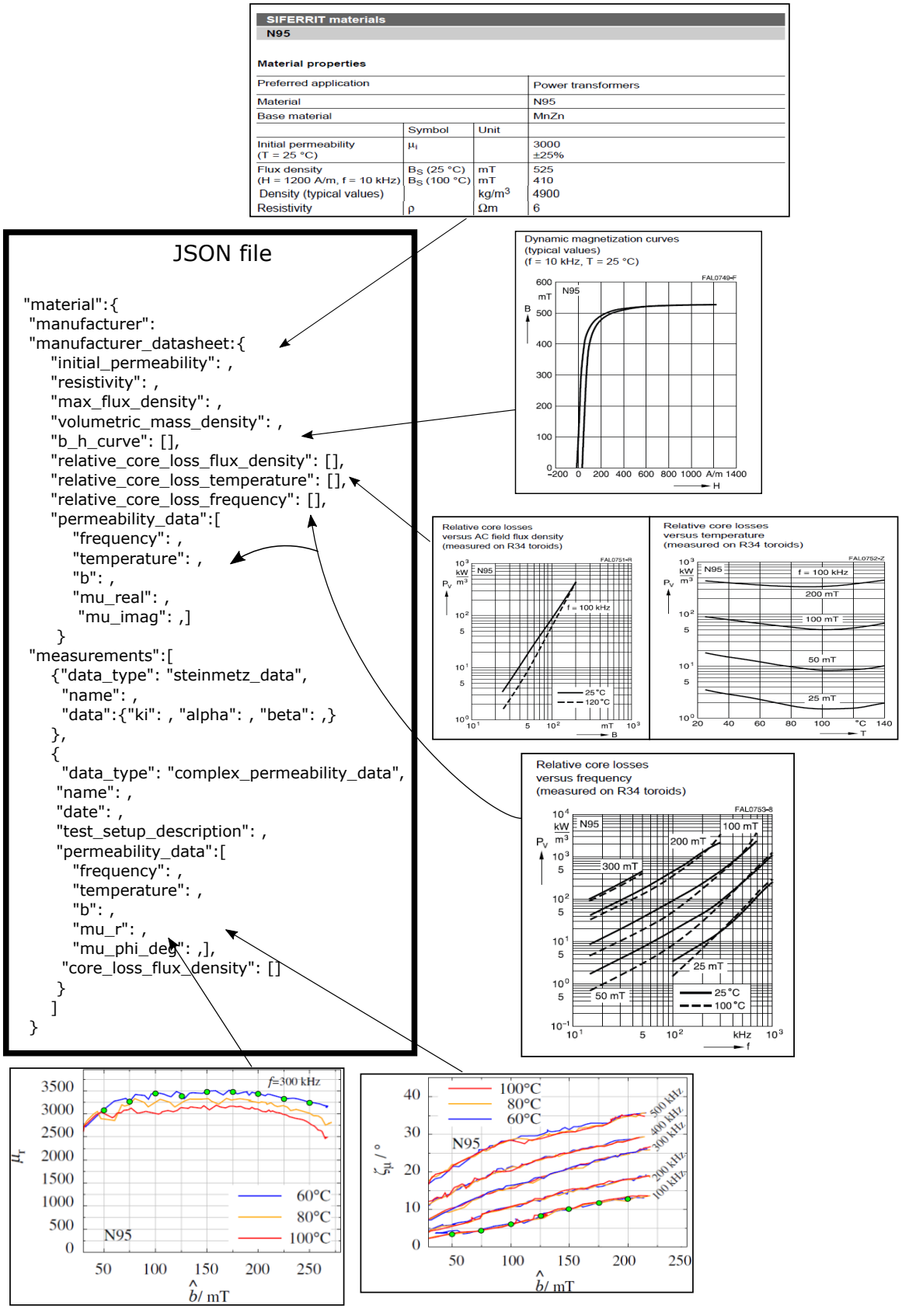


Fig. 3.10: Process to add new material in the database

4 Automated design

This chapter describes the development of a brute force optimization application using the reluctance model and the FEM simulation. *Automated design* is the name assigned to the application, as it automatically finds the optimum design solutions with the applied design parameters. It is defined in the file $reluctance\_fem\_integration.py$ of the FEM magnetic toolbox. The chapter defines the concept, the filters applied to reduce computation, and the results obtained from the application.

4.1 Concept of Automated Design

One approach in optimization is straightforward and requires considerable computation power: brute force methods which try to calculate all possible solutions and decide afterward which one is the best. These methods are feasible only for small problems since the number of possible states of the system increases exponentially with the number of dimensions. Despite these drawbacks, brute force methods do have a few benefits: they are simple to implement, and in the case of discrete systems, all possible states are checked.

The *automated design* follows the brute force method, as explained above. As the computation time of a brute force method is very high, *automated design* makes use of the reluctance model to reduce that time.

The overall workflow of the application is described in Figure 4.1. The process starts with setting the design parameters and constraints. A data matrix (NumPy array) is created with all the possible combinations of the design parameters. The columns of the matrix store the design parameters, whereas the rows, represent different cases/designs possible.

The reluctance model then uses the data matrix to calculate inductance and flux for all the cases. The filtration stage uses these quantities to filter out the designs as per the applied constraints. Thenceforth, the final matrix undergoes the FEM simulation. The advantage is that only $5\% - 10\%$ (depending on the applied constraints) of the total cases reach the FEM stage. Thus, fast pre-calculation (filtration) and very accurate FEM simulation provide the user with a fast and accurate method to design a magnetic inductor.

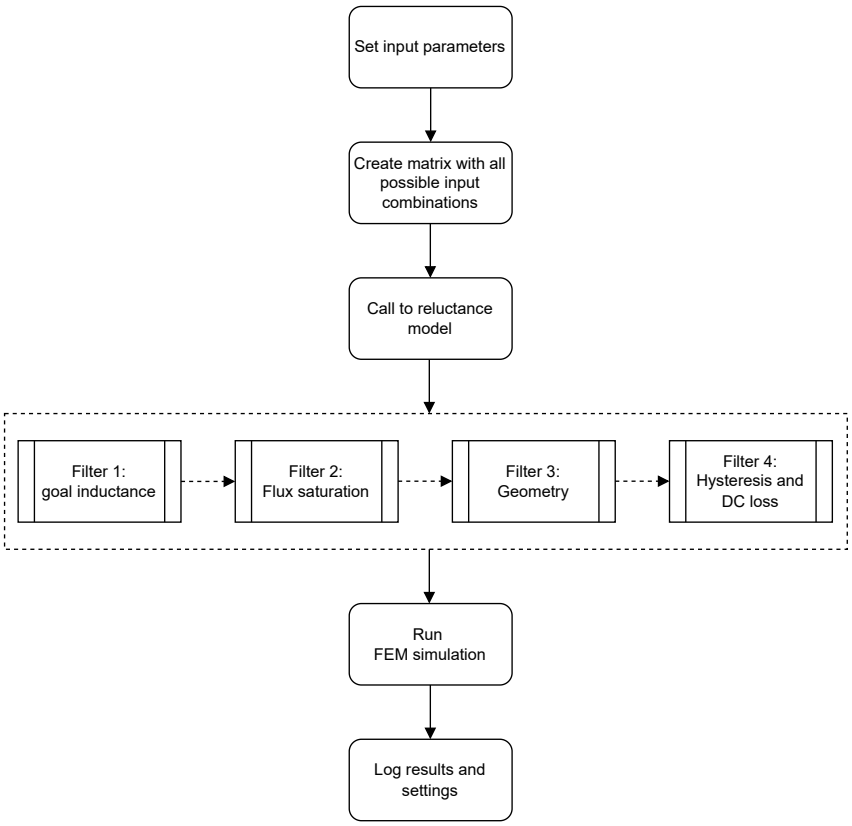


Fig. 4.1: Workflow of automated design

4.2 Design Parameters and Constraints

Table 4.1 summarizes the design parameters and the constraints. The rows in yellow represent the constraints. They are the conditions used during the filtration process and therefore consist of a single value. The ones in the green depict those parameters which can take sweep/multiple values (list). And lastly, the rows in red are those parameters that take a single value.

The parameter *multiple_air_gap_type* requires some additional description. It is necessary to specify the type of distributed air gaps. Distributed air gaps are multiple air gaps equidistant from each other. There are two possible ways of arranging such distribution. Figure 4.2 shows the two types of distributed air gaps. *Edge-distributed* consists of air-gaps in the center leg and on the edges. Whereas *center-distributed* have air-gaps only in the center leg. The two types of air gaps shown in Figure 4.2 (round-round and round-inf) are discussed in Section 2.2.2.

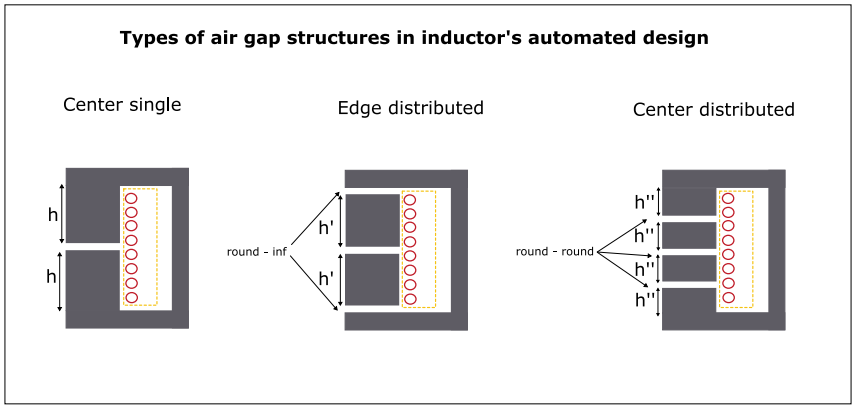


Fig. 4.2: Distributed air-gap types

Tab. 4.1: Design parameters and constraints;(yellow): constraints; (red): single value; (green): sweep value

Parameter	Symbol	Description
goal_inductance	L_{goal}	Sets the required inductance (in Henry).
%tolerance	-	goal inductance tolerance band: $\pm 10\%$ *or* $\pm$ 20%
frequency	f	Frequency of applied current (in Hz).
temperature	T	Temperature of the core (in °C)
peak_current	i_{peak}	Peak excitation current amplitude (in Ampere) required for calculating maximum flux
core_material	-	Required for calculations related to permeability and other material data:{N95, N87, N49}
multiple air-gap type	-	Sets the type of distributed air-gap 'center' or 'edge' distributed
% of flux saturation	$\%b_{\text{sat}}$	Sets the maximum allowed flux density
% of total loss	$\%P_{\text{loss}}$	Sets the maximum allowed total loss
Geometry	-	Required geometry details to construct the core
Air-gap details	-	Required for inductance calculation by the reluctance model
conductor details	-	Sets the type of conductor and its properties: 'litz' or/and 'solid'

4.3 Filtration

The filtration stage filters out cases which do not satisfy the given constraints. The basis of the filter functions is the results obtained from the reluctance model and general intuition. They are classified based on the conditions applied to them.

4.3.1 Goal Inductance

Inductance is one of the main results obtained from the reluctance model. As mentioned in Chapter 2, the percent error margin of the model is approximately ±10%. Therefore, a tolerance band is required to accommodate this approximation. Hence, the author recommends having a minimum percent tolerance of ±10% in this filter.

4.3.2 Flux Saturation

The flux through the inductor is another quantity that is obtained from the reluctance model. Saturation flux density (b_{sat}) is the property of a magnetic material, and so it is extracted from the core material database. Due to the saturation of the flux density, it is a good practice to limit it to $50\% - 70\%$ of the b_{sat}. Thus, the constraint $\%b_{\text{sat}}$ is set to filter the cases which might saturate.

$$b_{\text{core}} < (0.5 \cdots 0.7) \cdot b_{\text{sat}} \tag{4.1}$$

4.3.3 Winding and Core Window

The basis of the design of this filter is general intuition. As stated before, *automated design* is a brute force method. Thus, it is natural to have some invalid cases that are physically not realizable. The filter compares the area taken by the conductor and the available window area. It is simply given by

$$N \cdot \pi r_{\text{cond}}^2 < winding\,fill\,factor \times (window\,area - insulation\,area) \tag{4.2}$$

where N is the number of turns, r_{cond} is the conductor's radius, $window\,area$ is the total physical area, and $insulation\,area$ is the area consumed by the insulation across the window. $winding\,fill\,factor$ denotes the theoretical usable area by the winding (depending on winding scheme) and should not be confused with the fill factor of a litz-wire. For *Square* winding scheme, it is 0.785, while for a *Hexagonal*, it is 0.907 [11].

4.3.4 Hysteresis loss

The calculated losses consist of hysteresis and DC loss. It is not an accurate but an approximate measurement of the losses. This way, the design cases can be sorted, and only a percentage of those cases are sent to the FEM simulations. $\%P_{\text{loss}}$ is the constraint that controls the design analysis. Hysteresis loss is calculated using

$$P_{\text{hysteresis}} = \frac{1}{2}\omega Im\{\mu\}\left(\frac{b_{\max}}{\mu_0\mu_{\text{r}}}\right)^2 \tag{4.3}$$

where ω is the angular frequency, $Im\{\mu\}$ is the imaginary permeability (obtained from material database), $b_{\max}$ is saturated flux density, and μ_{r} is the relative permeability, also taken from the material database.

$$P_{\text{hysteresis density}} = P_{\text{hysteresis}} \cdot V_{\text{section}} \tag{4.4}$$

where V_{section} is the volume of the section. Figure 4.3 illustrates the volume considered for each section as per [4]. The flux does not flow through the corners of the core as it tends to bend near them. Thus, the corner volumes are ignored.

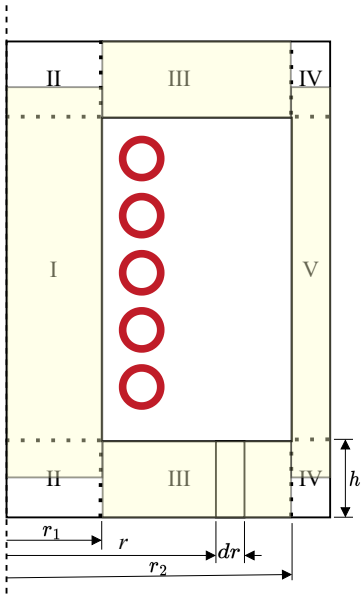


Fig. 4.3: Section volume considered for hysteresis loss density

Hysteresis loss density for sections I and V are solved using the equation (4.3). Section III does not constitute a constant flux density. The flux density is a function of the radial component r. In cylindrical coordinates, the differential volume is

$$dV = rdrdzd\theta.$$

For the section III, it becomes

$$dV = rdr\int_0^h dz\int_0^{2\pi} d\theta,$$

giving us

$$dV = (2\pi h)\, r dr.$$

Whereas, magnetic flux density is

$$b_{\max} = \frac{\phi_{\max}}{2\pi r h}$$

where $\phi_{\max}$ is the maximum flux. The hysteresis loss density of section III is then calculated by the integration

$$P^{III}_{\text{hysteresis density}} = \frac{1}{2}\omega Im\{\mu\} \int_{r_1}^{r_2} \left(\frac{\phi_{\max}}{(2\pi r h)\, \mu_0 \mu_{\text{r}}} \right)^2 dV. \tag{4.5}$$

Solving the above integral results into

$$P^{III}_{\text{hysteresis density}} = \frac{1}{2}\omega Im\{\mu\} \left(\frac{\phi_{\max}}{\mu_0 \mu_{\text{r}}} \right)^2 \left(\frac{1}{2\pi h} \right) ln \left(\frac{r_2}{r_1} \right). \tag{4.6}$$

All losses calculated using (4.3) are solved with an assumption of homogeneous flux density. Also, the imaginary permeability, which is a function of flux density, is assumed constant in solving the hysteresis loss density of section III. This is for the sake of simplicity and fast computation

4.4 Post Filtration

After the filtration stage, FEM simulation is run for the final data matrix. It checks for each case whether simulation is possible or not. If not possible, *automated design* skips that particular case. It then creates the following files and folders for user review and debugging purposes.

- example_results: This is the directory where FEM simulation results are stored.
- fem_simulation_data: The folder stores the JSON result files of all the cases simulated by the *automated design*
- automated_design_settings.JSON: The JSON files stores all the design parameters and constraints for later review.
- data_matrix_fem.CSV: The data matrix that is supplied to the FEM is stored in this CSV file. Figure 4.4 shows the screenshot of the data matrix. The rows represent the design case with all the design parameters at different columns.

Case_no.	core_inner_diameter	window_h	window_w	mu_rel	no_of_turns	n_air_gaps	air_gap_h	air_gap_position	mult_air_gap_type	inductance
0	0.05	0.01	0.005	3000	7	2	0.0005	0	2	0.000122981
1	0.05	0.01	0.005	3000	7	2	0.0005	0	2	0.000122981
2	0.045	0.01	0.005	3000	8	2	0.0005	0	2	0.000130769
3	0.03	0.01	0.005	3000	11	2	0.0005	0	2	0.000112441
4	0.05	0.01	0.0088889	3000	7	2	0.0005	0	2	0.000122815
5	0.045	0.01	0.005	3000	8	2	0.0005	0	2	0.000130769

Fig. 4.4: Partial screenshot of the data_matrix_fem.CSV file, consisting of 6 design cases and 11 of 33 columns (design parameters); (Simulation settings: Appendix A.1)

In addition to files and folders, one more filter (constraint) is added after the simulation, which again filters based on the goal inductance with a separate tolerance band (adjustable by the user). All the filters applied before are based on the reluctance model, which in itself is an approximate model. Thus, the final restriction on the simulated FEM cases removes any remaining ambiguous results.

4.5 Simulation Results

Automated design is run using the input settings as given in Appendix A.1. There are three major graphs generated. The first graph, shown in Figure 4.5, displays a plot between volume and loss. This graph is generated from the final data matrix (approximate losses).

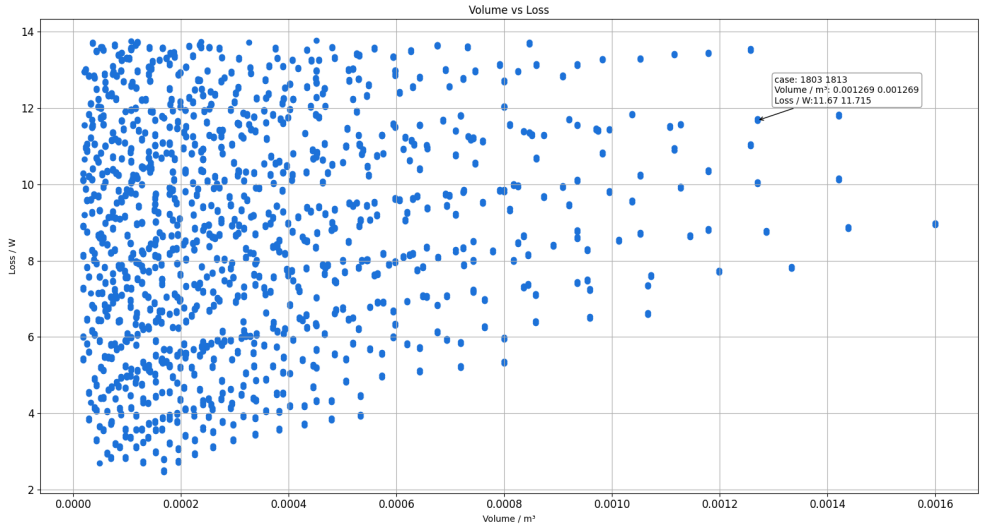


Fig. 4.5: Volume vs Hysteresis loss from the reluctance model

The second graph is also plotted between volume and total loss (core and winding loss) but is generated from the FEM results and is presented by Figure 4.6. The cost is the

third quantity that is represented using the color band. The following observations are made:

- In the result, the total cost appears to increase linearly with the volume, but this will not always be the case. The total cost consists of core and winding costs. Therefore, it is plausible that core cost is dominant for the simulated scenario.
- The graph is interactive such that information about any point is shown when the mouse hovers over it. This helps the user in analysing a particular case in detail.
- A Pareto front is visible at the bottom left corner of the graph. All points on it represent an optimized design.

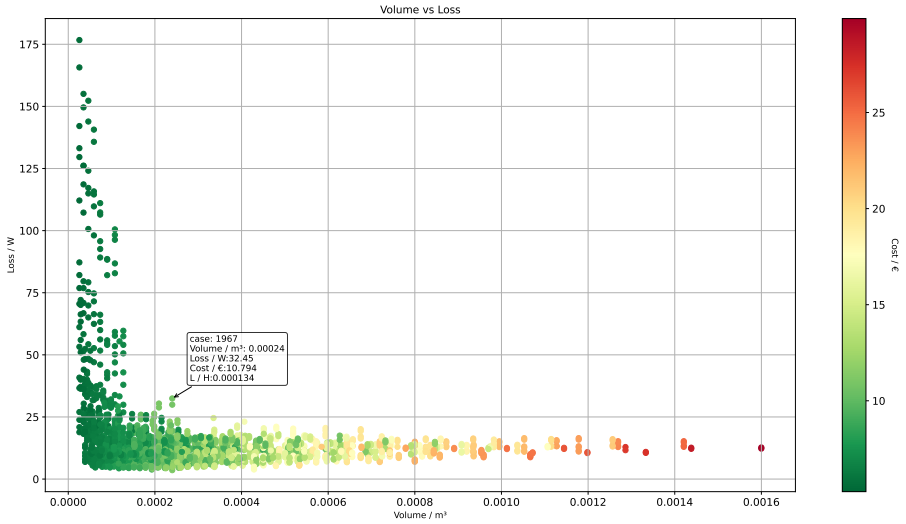


Fig. 4.6: Volume vs Total loss with points color based on Cost

Figure 4.7 shows the graph between volume and cost with the color based on the total loss. Both the figures illustrate three quantities (dimensions) in a 2D graph, viewed from different angles.

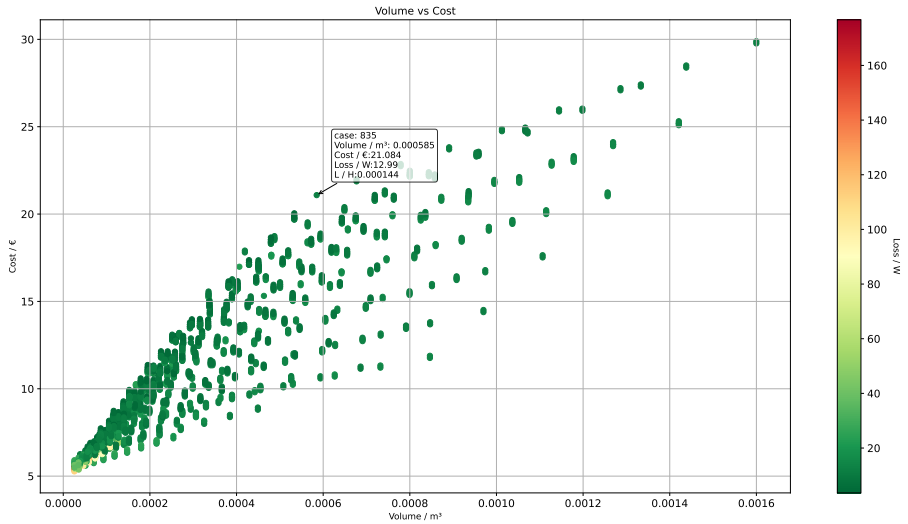


Fig. 4.7: Volume vs Cost with points color based on Total loss

5 GUI

This chapter intends to provide an understanding of the graphical user interface(GUI)[12], which is made for the mentioned functions in the chapters before.

5.1 Manual Design

The manual design tab for the FEM simulation of inductors has been extended for transformers, as shown in Fig 5.1. The first tab, definition, helps the user to select parameters of the respective magnetic component dynamically. The parameters like the selection of magnetic component, its core, winding, and air gap definition are selected or entered in addition to the winding insulation for the simulation, which thereby is performed in the subsequent tabs. Simulation tab is primarily for

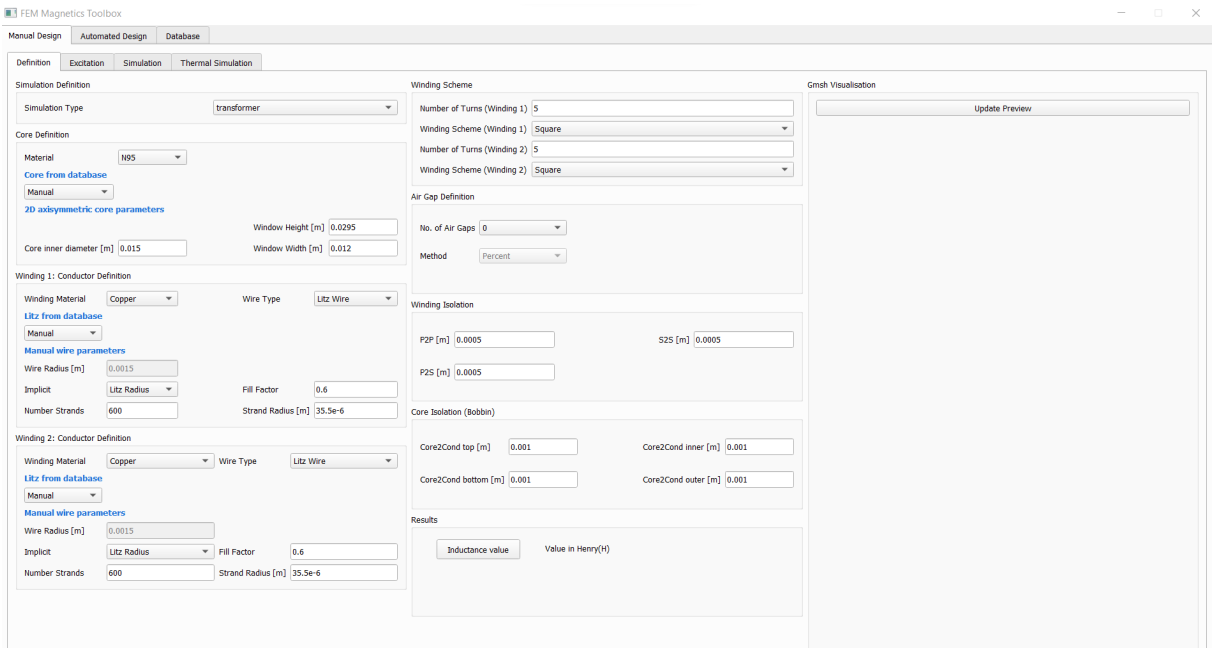


Fig. 5.1: Manual design tab

the FEM magnetic simulation and the thermal simulation tab, for the FEM thermal

simulation of the material. The simulated results are then displayed in their respective tabs.

5.1.1 Inductance Value Calculation

In the manual design tab, inductance value is calculated and displayed. The values from core, conductor, winding, and air gap definition are accepted to calculate the inductance value using the equation, (2.1), and is returned in Henry.

5.2 Automated Design

Automated design developed for the sweep simulation, primarily uses reluctance models for its simulations to reduce the possible number of FEM simulations, optimizing the design process. The first tab, *Definition tab*, facilitates user to choose the design parameters and constraints as shown in Fig 5.2. This is used to calculate the inductance and flux for all the cases. The second tab, *Reluctance Models*, as shown in Fig 5.3, displays multiple filtered values as per the applied constraints. The final cases, *FEM cases*, will then undergo FEM simulation which is performed in the third tab, *FEM simulations*, as shown in Fig 5.4, on the click of the *Simulate button*. A fourth tab, *Load(Results)*, as shown in Fig 5.5, has been implemented to display simulated results from the directory path entered [13].

Automated design features first of its type, list-box widget pairs, dynamic parameters and multiple value sweep.

1. List box widget pairs:
   The choice of multiple design parameters made in the definition tab are summarised, before the simulation. This is made possible using an add to basket model list box widget pairs. The first, *options list box*, as shown in Fig 5.6a, displays the possible choices and the second, *basket list box*, as shown in Fig 5.6b, acts as a basket to which the choices are made.
   The selection of choices could either be made by a double-click on the choice or by selecting the choice by a single-click and then press the add button to transport it to the basket list box. For much easier selection of all the choices, an additional *select all* button has been implemented to select all the choices to the basket list box. No multiple selection can be possible for the same choice. The choices made can be always taken back from the basket using either the clear button, after selecting the required choice or the entire selection can be cleared by the clear all button.
   Additionally, in the definition tab, selection of core is also done manually, as shown in Fig 5.7a, by entering the respective parameters, which are taken for the simulation along with the choice of standard cores. An add button for the same transports the choice to a different basket, called the *core geo manual choices basket*, as shown in Fig 5.7b.

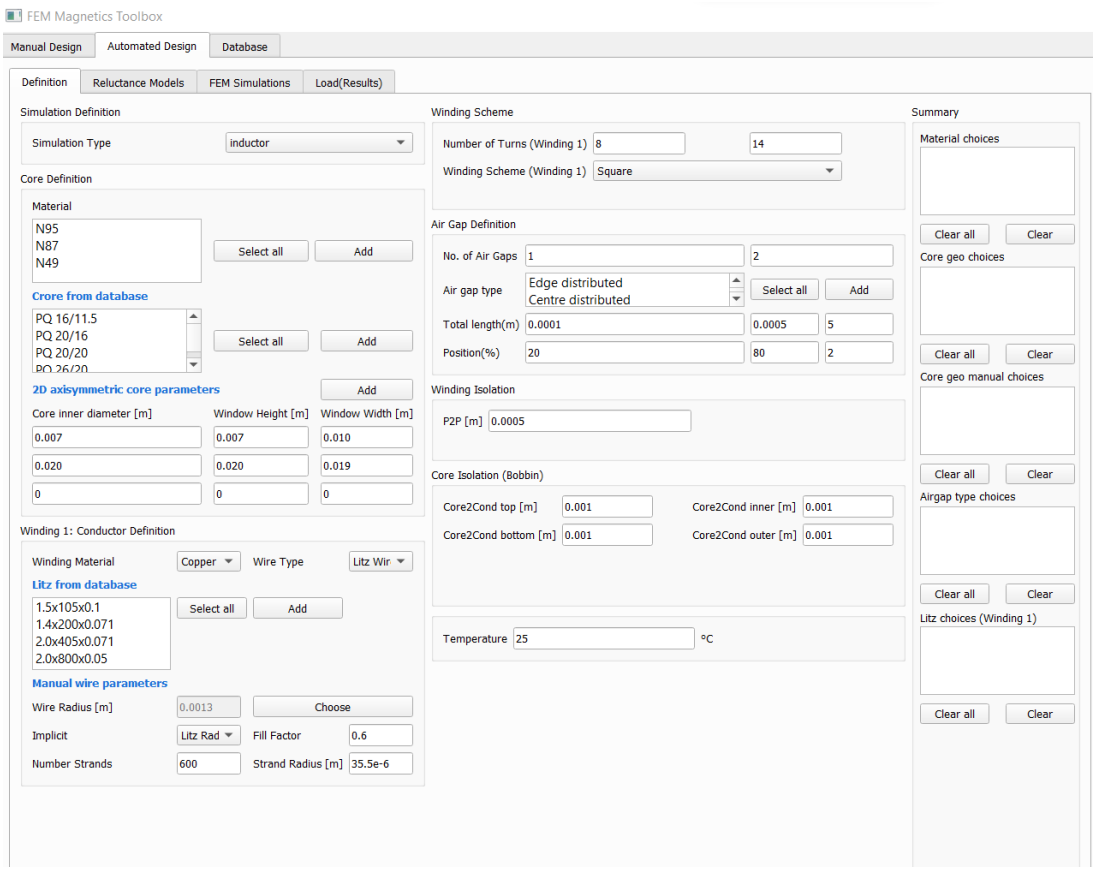


Fig. 5.2: Definition tab

2. Dynamic parameters:
   The parameters are dynamically chosen, behaving to the way choices are made. Selection of a particular parameter enables/disables the selection of the other relevant/irrelevant parameters as shown in Fig 5.8a and Fig 5.8b.

3. Multiple value sweep:
   The primary objective of the automated design to sweep the values, are accepted through three line edit boxes respectively for minimum value, maximum value and the step value represented in a placeholder text as shown in Fig 5.9.

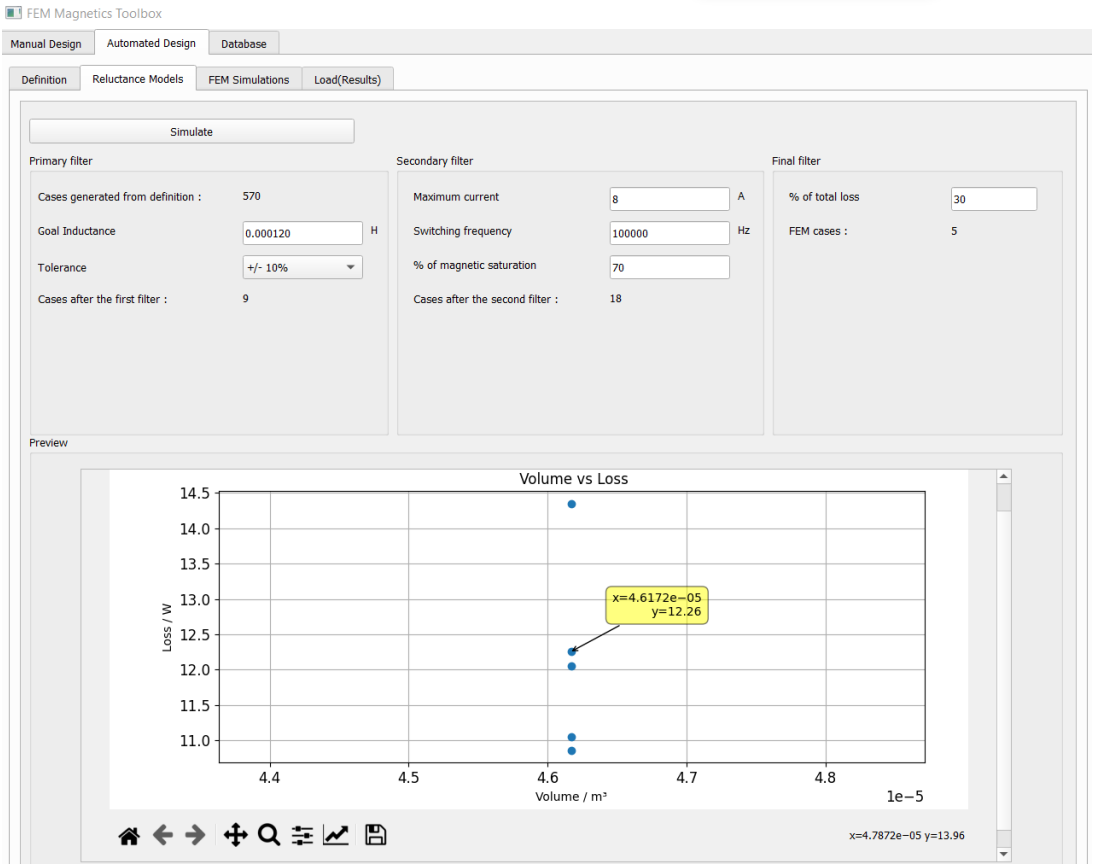


Fig. 5.3: Reluctance models tab

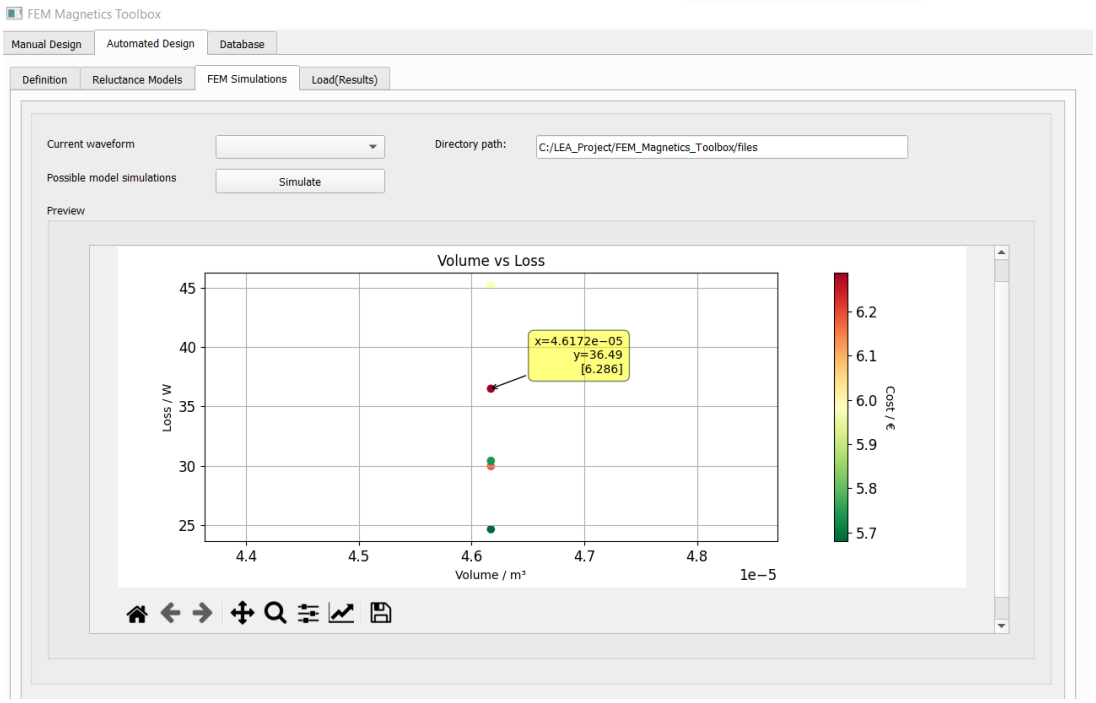


Fig. 5.4: FEM simulations tab

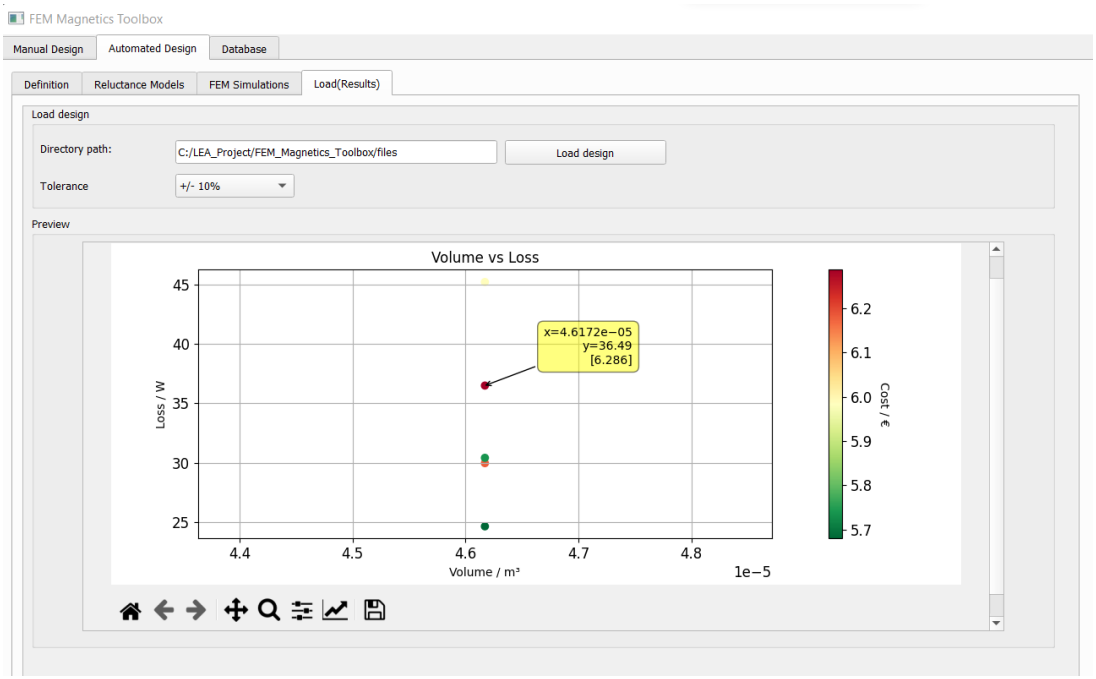
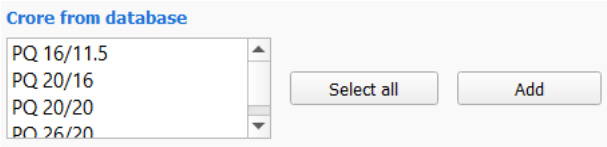
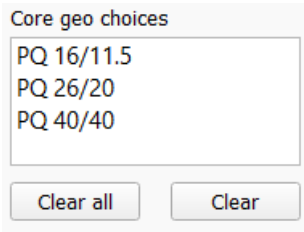


Fig. 5.5: Load(results) tab

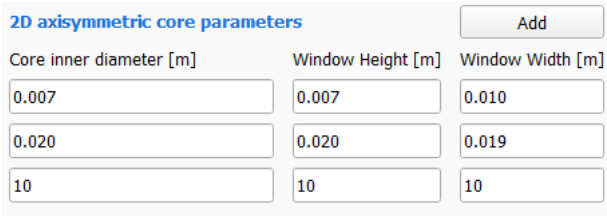


(a) Options list box

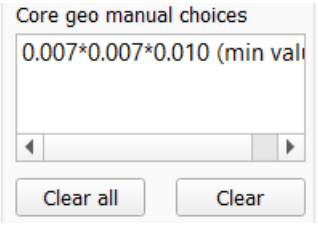


(b) Basket list box

Fig. 5.6: List box widget pairs

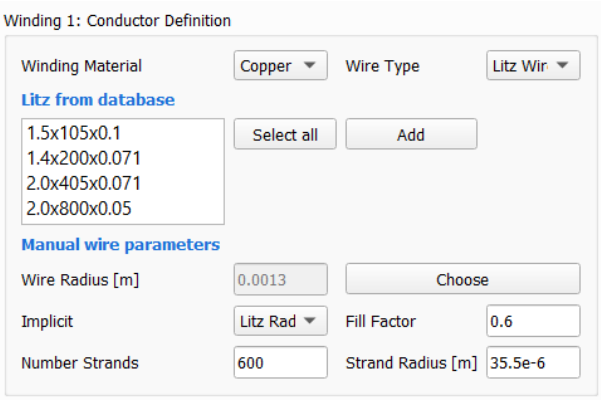


(a) Manual parameters

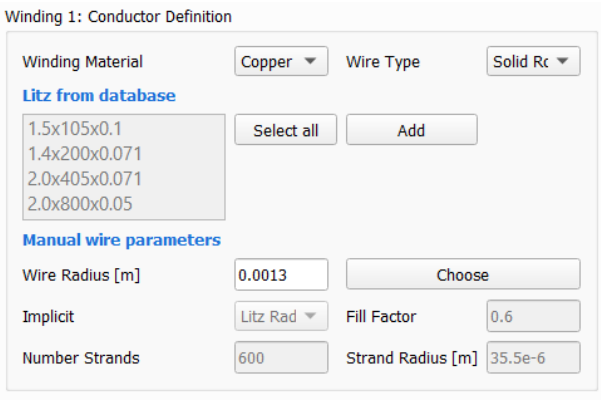


(b) Manual choice basket

Fig. 5.7: Manual selection



(a) Litz wire selection



(b) Solid wire selection

Fig. 5.8: Dynamic parameters

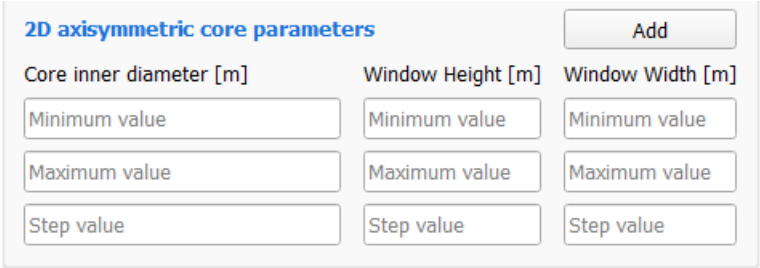


Fig. 5.9: Multiple value sweep

5.3 Database

The database tab, compares the data of different materials, which helps the user to choose between the materials available in the database for their simulations. Multiple plots are presented in a single window facilitating an easier comparison and the plots are made interactive with a click to view values model.
The choice of materials and their respective values are made dynamic depending on the values available in the database for the particular material chosen.
The first tab, datasheet-datasheet, as shown in Fig 5.10, compares relative power loss of a material with temperature, relative power loss of a material with flux, relative power loss of a material with frequency. The second tab, measurement-measurement, as shown in Fig 5.11, compares complex permeability within different materials. The third tab, datasheet-measurement, as shown in Fig 5.12, compares the datasheet and the measurement readings of the particular material selected.

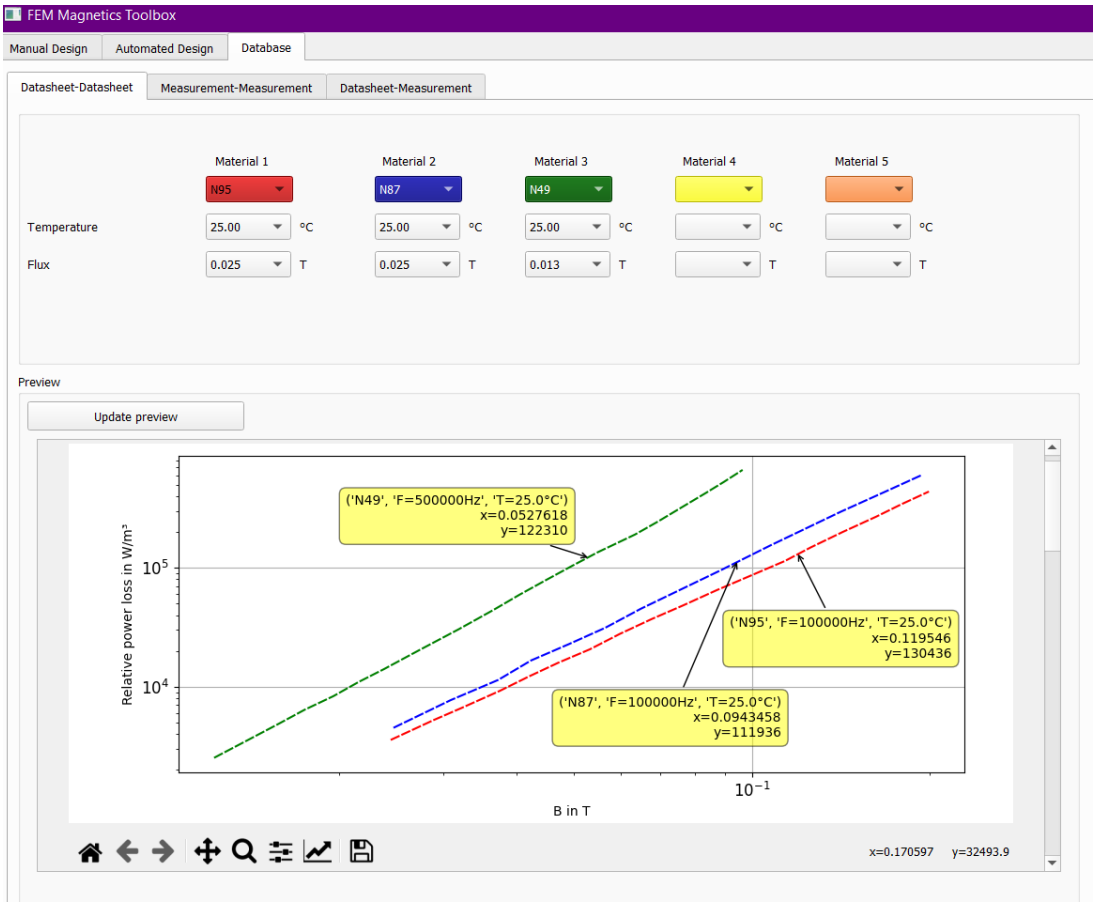


Fig. 5.10: Datasheet vs datasheet plot

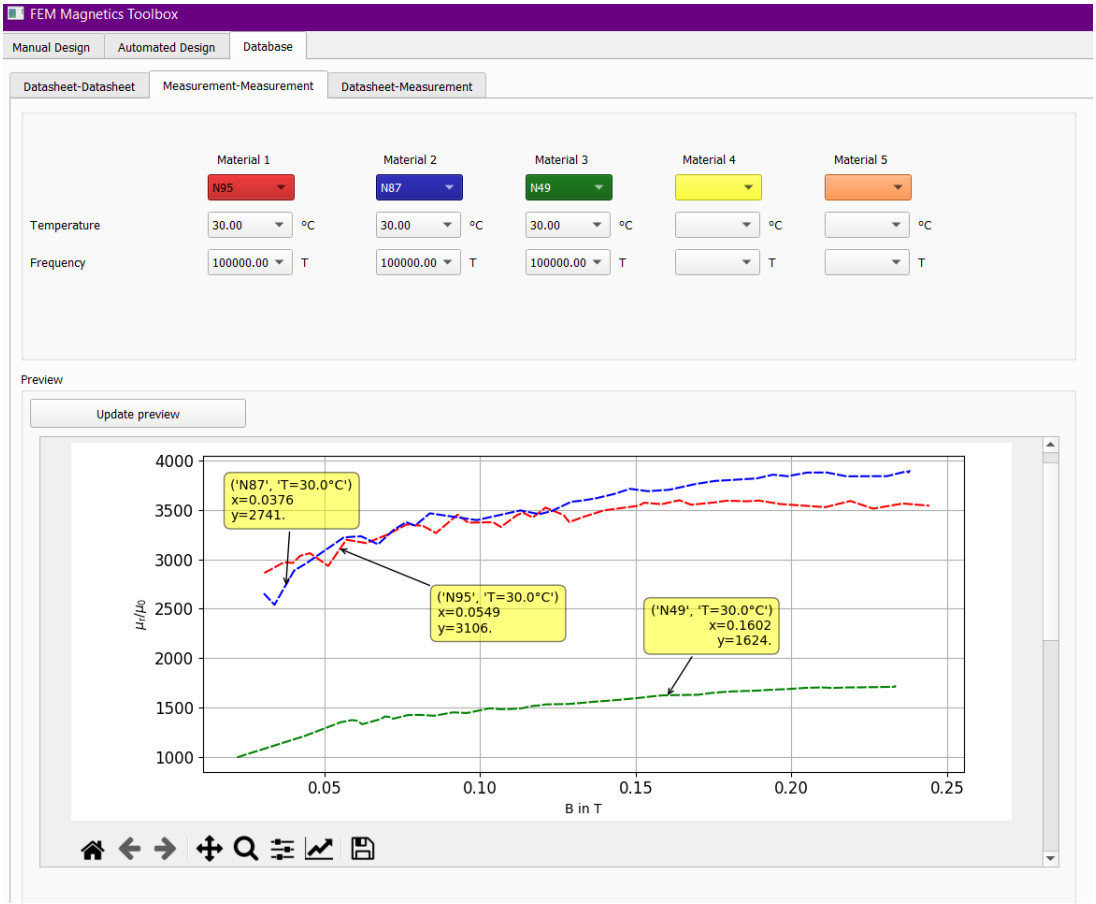


Fig. 5.11: Measurement vs measurement plot

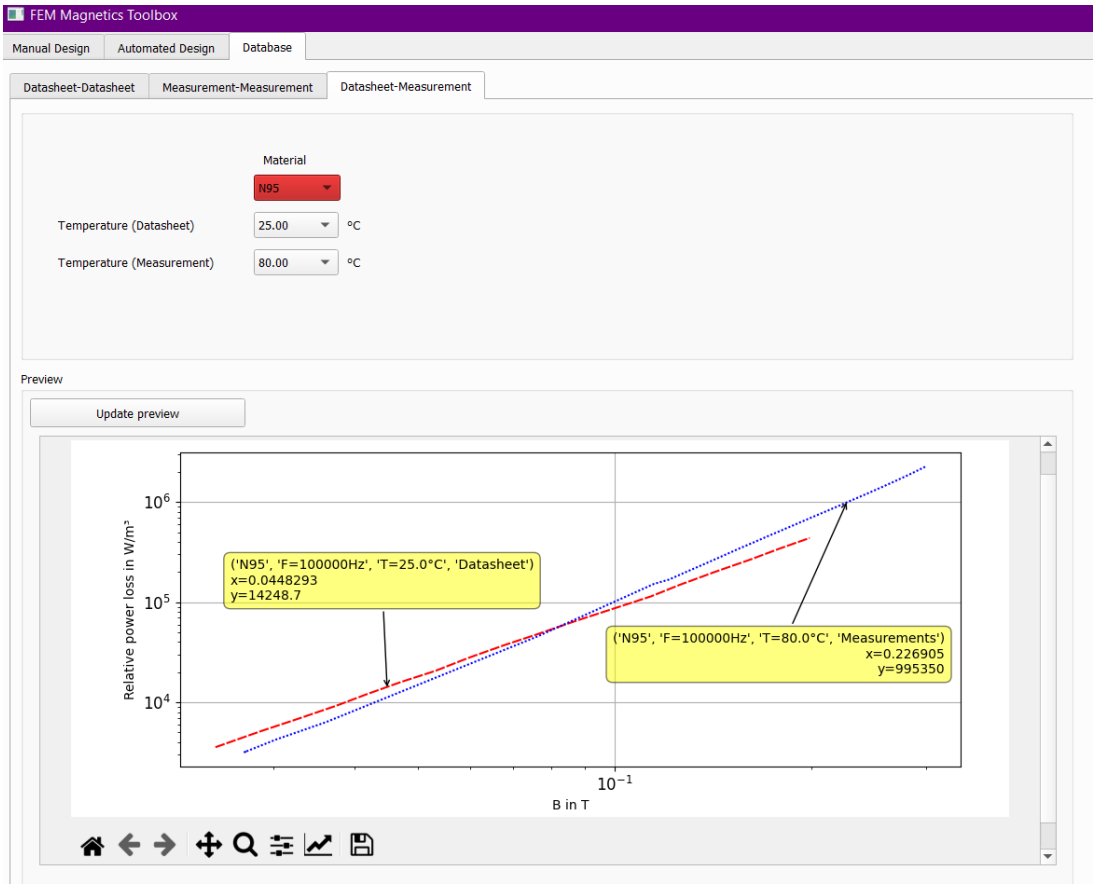


Fig. 5.12: Datasheet vs measurement plot

Appendix

Automated design simulation settings for the example design shown in Section 4.5.

```
ad = AutomatedDesign(working_directory='D:/Personal_data/MS_Paderborn
                     /Sem4/Project_2/2022-11-30_fem_simulation_data',
                      magnetic_component='inductor',
                      goal_inductance=120 * 1e-6,
                      frequency=100000,
                      goal_inductance_percent_tolerance=10,
                      winding_scheme='Square',
                      peak_current=8,
                      percent_of_b_sat=70,
                      percent_of_total_loss=30,
                      database_core_names=[],
                      database_litz_names=['1.5x105x0.1',
                      '1.4x200x0.071'],
                      solid_conductor_r=[],  # 0.0013
                      manual_core_inner_diameter=list
                      (np.linspace(0.005,0.05,10)),
                      manual_window_h=list(np.linspace(0.01,0.08,5)),
                      manual_window_w=list(np.linspace(0.005,0.04,10)),
                      no_of_turns=[2, 3, 4, 5, 6, 7, 8, 9, 10, 11,
                      12, 13, 14, 15, 16, 17, 18, 19, 20],
                      n_air_gaps=[1, 2],
                      air_gap_height=list
                      (np.linspace(0.0001, 0.0005, 5)),
                      air_gap_position=list(np.linspace(20, 80, 2)),
                      core_material=['N95'],
                      mult_air_gap_type=['center_distributed'],
                      top_core_insulation=0.001,
                      bot_core_insulation=0.001,
                      left_core_insulation=0.001,
                      right_core_insulation=0.001,
                      inner_winding_insulation=0.0005,
                      temperature=100.0,
                      manual_litz_conductor_r=[],
                      manual_litz_strand_r=[],
                      manual_litz_strand_n=[],
                      manual_litz_fill_factor=[])
```

Listing A.1: Automated design settings

Lists

List of Tables

List of Figures

References

[1] *Onelab: Getting started*, http://www.onelab.info/, Accessed: 2022-06-10.
[2] E. C. Snelling, *Soft Ferrites, Properties and Applications*. Butterworths, 1988.
[3] J. Mühlethaler, "Modeling and multi-objective optimization of inductive power components," Ph.D. dissertation, ETH Z'urich, 2014.
[4] T. Piepenbrock, *Automated fem transformer design for a dual active bridge*, 2021.
[5] A. Balakrishnan, W. T. Joines, and T. G. Wilson, "Air-gap reluctance and inductance calculations for magnetic circuits using a schwarz-christoffel transformation," in *IEEE Transactions on Power Electronics*, 1997, pp. 654–663.
[6] *Siferrit material n95*, May 2017. [Online]. Available: https://www.tdk-electronics.tdk.com/download/528866/73730346930dfce60e468312e5e3023e/pdf-n95.pdf.
[7] *Siferrit material n87*, Sep. 2017. [Online]. Available: https://www.tdk-electronics.tdk.com/download/528882/71e02c7b9384de1331b3f625ce4b2123/pdf-n87.pdf.
[8] *Siferrit material n49*, May 2017. [Online]. Available: https://www.tdk-electronics.tdk.com/download/528856/cf394eea3fae828c345f46dc297b76ab/pdf-n49.pdf.

[9] L. Keuck, "Entwurf eines einstufigen ladewandlers auf basis eines llc-resonanzwandlers (unveröffentlicht)," in *PhD thesis. Universität Paderborn*, 2021.

[10] J. Böcker, "Analysis of the magnetic skin effekt in motors and inductors," in *2020 International Symposium on Power Electronics Electrical Drives, Automation and Motion*, 2020.

[11] C. W. T. Mclyman, *Transformer and inductor design handbook*. Marcel Dekker, 2004.

[12] *Qt designer: Qt designer manual*, https://doc.qt.io/qt-6/qtdesigner-manual.html, Accessed: 2022-05-30.

[13] *Matplotlib: Visualization with python*, https://matplotlib.org/, Accessed: 2022-05-30.