Optimization Design of Wireless Charging Coil Geometry

Based on Pymoo Multi-Objective Optimization and Finite Element Analysis

Author: Haobo Guo

Date: August 22, 2025

Optimization Design of Wireless Charging Coil Geometry

Based on Pymoo Multi-Objective Optimization and Finite Element Analysis

By
Haobo Guo

In partial fulfilment of the requirements for the degree of:
Master of Science
in Electrical Engineering
Electrical Power Engineering

at the Delft University of Technology,
to be defended publicly on 28 August 2025

Supervisor:	Prof.dr.ir. P. Bauer (Responsible Supervisor)		
	Dr. Jianning Dong (Daily supervisor)		
Thesis committee:	Jianning Dong,	Chair	TU Delft
	Fabio Muñoz Muñoz,	Core Member	TU Delft
	Aditya Shekhar,	Core Member	TU Delft

This thesis is confidential and cannot be made public until 28 August 2025

Acknowledgments

First of all, I would like to express my sincere gratitude to Prof. Dr. Ir. P. Bauer for his willingness to serve as my supervisor and for his invaluable teaching and guidance during my first-year courses. I am also deeply grateful to Dr. Jianning Dong, Dr. Zian Qin, Dr. Gautham Ram Chandra Mouli, Dr. Laura Ramirez Elizondo, Dr. Hani Vahedi, and Dr. Sebastián Rivera Iunnissi, who provided crucial instruction and support in various courses. With their guidance, I was able to comprehensively and thoroughly acquire professional theoretical knowledge in the field of power electronics, laying a solid foundation for my future work and research. Additionally, I would like to thank Harrie Olsthoorn and Hitesh Dialani from the laboratory for their meticulous support in hands-on learning activities, which greatly enhanced my practical skills and enabled me to integrate theoretical knowledge with real-world applications.

Secondly, I would like to extend my special thanks to Dr. Jianning Dong and Joris Koudijs from TILER. Dr. Jianning Dong introduced me to this coil optimization design project from TILER, and Joris Koudijs showed great interest in this research from an open-source perspective, providing the initial design framework. Dr. Jianning Dong offered me a wealth of relevant learning materials and guidance. Together, we defined the project structure and the methods to achieve our objectives, and in each routine meeting, we discussed encountered challenges and the progress of solutions.

Furthermore, the support I received from both the official ElmerFEM resources and the community was crucial in my learning process for electromagnetic simulations using ElmerFEM. The ElmerFEM Webinar video tutorials provided a systematic introduction that helped me become familiar with the solver’s usage and development logic, while the elmer-elmag GitHub repository contained a valuable collection of practical examples. Among these, the Fivecoils example and the TEAM 7 electromagnetic induction coil example provided by Peter were particularly helpful for my project. Additionally, in the Elmer Discussion Forum, Peter offered invaluable advice and assistance regarding how to define closed coil currents, which greatly contributed to overcoming the challenges I faced.

Finally, I would like to express my deepest gratitude to my parents for their unwavering support and financial assistance throughout my studies abroad. They have always been my strongest pillar of support. I am also sincerely thankful to my friends for accompanying me in both my academic and personal life. Without you all, I would not be who I am today.

Delft, March 19, 2025
Haobo Guo

Abstract

This thesis presents an implementation method for optimizing the external geometric dimensions of an existing wireless power transfer (WPT) coil through multi-objective optimization. Wireless charging systems have been widely applied in daily electrical devices, and the trade-off between the geometric dimensions of the charging system, its charging efficiency, and power transfer capability is a key challenge faced by designers and manufacturers. During the design process, the evaluation methods for the power, losses, weight, and size of wireless charging coils significantly influence the product design cycle as well as the labor and time expenses associated with the design process. Based on an existing WPT coil sketch, this thesis designs and verifies the feasibility of implementing a multi-objective optimization method. The proposed optimization method is developed based on given power transfer requirements and external dimension constraints. A 3D geometric model is reconstructed using the SALOME open-source modeling platform, where meshing is performed to prepare the geometry for finite element analysis (FEA). The ElmerFEM open-source finite element solver is then employed to evaluate the coil's performance from multiple perspectives. To achieve large-scale iterative optimization for a single performance evaluation, a multi-objective constrained optimization framework is formulated in a Python environment, where constraint equations are defined and deployed using Pymoo. To address the challenges encountered in the implementation of this design method, this thesis primarily considers two key aspects. The first is the automated performance evaluation of a given coil geometry using Python. Given a set of geometric parameters, SALOME can be automated using Python scripts to generate the geometric model and perform mesh generation through a descriptive approach. The generated mesh files are then processed by the ElmerFEM solver, which, supported by SIF configuration files, computes the required physical quantities for performance evaluation. The obtained physical quantities are refined through a proposed computational method to extract the objective data necessary for multi-objective optimization. The second aspect is the multi-objective optimization of coil dimensions using Pymoo. Pymoo is a well-established open-source multi-objective optimization framework. It enables designers to define problems, establish constraints, and formulate quantitative equations, thereby integrating and binding optimization cases with real-world applications. By leveraging this framework, the desired optimization design is effectively achieved.

Table of Contents

1 Introduction

This chapter primarily introduces the relevant background knowledge related to the study, defines the research problem and scope, and outlines the structure of the paper.

1.1 Background

Since the 2010s, Wireless Power Transfer (WPT) technology has been widely adopted across various aspects of daily life. Among its different implementations, Inductive Power Transfer (IPT) technology is particularly prominent, with applications ranging from high-power wireless charging systems for electric vehicles to low-power charging solutions for implantable medical devices [1], [2], as shown in Figure 1. Driven by the urgent societal demand for low-carbon and emission-reducing transportation, vehicles and bicycles powered by an electric motor instead of internal combustion engines have emerged. As a portable energy storage device, the battery has become an essential core component [3], [4]. This also implies that selecting an appropriate charging system for onboard power batteries has become a meaningful and valuable research focus. In contrast to conventional plug-in wired charging, due to its exceptional flexibility and safety, the IPT system has gained widespread adoption in outdoor charging scenarios [5].

Figure 1 Wireless charger for modern smartphones [6]

The operating principle of an IPT charging system is similar to that of a transformer with multi-turn coils, utilizing electromagnetic coupling between the input and load circuits to eliminate the need for direct conductive connections between them. The absence of physical wiring between the power supply and the load eliminates potential safety hazards associated with plug-and-socket connections, such as short-circuit incidents on rainy days and accidental contact by children or pets.

An IPT system primarily consists of rectification and inverter circuits on both the power supply and load sides, as well as the coupling coils that facilitate energy transfer between them [7]. The transmitter coil (also referred to as the primary coil or TXC) is powered by an AC source and generates an alternating magnetic field within a carefully designed magnetic circuit. This alternating magnetic field induces an electromotive force (EMF) in the receiver coil (also known as the secondary coil or RXC), thereby supplying power to the load connected to the receiving circuit [7].

In IPT system design, three key evaluation metrics are typically employed to determine the optimal design scheme: transmission power, transmission distance, and transmission efficiency. Depending on the specific application scenario of the target product, trade-offs often exist among these metrics.

This study explores the feasibility of employing multi-objective optimization design methodologies in conjunction with finite element analysis (FEA) techniques for the performance enhancement of inductive coupling coils, using a fixed inductive power transfer (IPT) coil for electric bicycle battery charging as a representative case study. Furthermore, to facilitate system modeling, simulation, and control prototyping, open-source platforms are incorporated into the development workflow. The integration of open-source tools not only accelerates the design and validation process but also promotes transparency, reproducibility, and extensibility in IPT system development.

1.2 Problem definition

This study investigates a multi-objective optimization design methodology for existing coil configurations, with a practical application case focusing on an inductive charger targeting 200 W output power at 108.5 kHz operational frequency. These specifications were derived from prior research involving comprehensive LTspice simulation analyses, as detailed in our previous work [8]. The subsequent tables (Tables 1 and 2) systematically present the essential characteristics of both primary and secondary coil assemblies, including but not limited to geometric parameters, electromagnetic properties, and loss quantification metrics.

Table 1 Electromagnetic specification of TXC [9]

TRANSMITTER COIL	
WIRE	110 STRANDS, 0.091MM STRAND
CORE MATERIAL	3C95
NO. OF TURNS	15
DC RESISTANCE	25mOhm +- 5mOhm
SELF-INDUCTANCE	78uH +- 4uH @100kHz

Table 2 Electromagnetic specification of RXC [10]

RECEIVER COIL	
WIRE	110 STRANDS, 0.091MM STRAND
CORE MATERIAL	3C95
NO. OF TURNS	20
DC RESISTANCE	17mOhm +- 3mOhm
SELF-INDUCTANCE	50uH +- 2uH @100kHz

This research focuses on the critical design constraints of volumetric footprint and mass minimization for wireless charging coils applied in electric bicycle systems. Figure 2 delineates the geometric configuration of the existing magnetic coupler assembly, where orange elements denote Litz wire coils and grey components represent ferrite cores. The upper assembly (receiver) is integrated within the e-bicycle frame structure, while the lower assembly (transmitter) constitutes a ground-mounted infrastructure module. The market-driven requirement for highly planar geometries and lightweight coil assemblies from demands for enhanced vehicular portability and flexible charging infrastructure deployment.

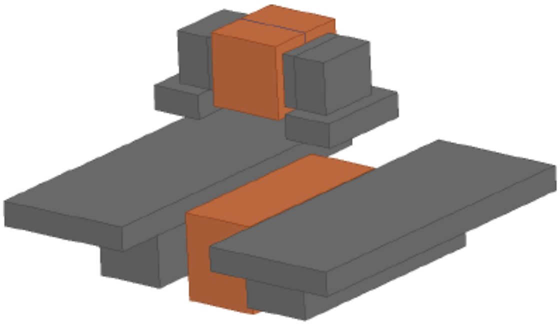


Figure 2 Geometric configuration of existing coils [8]

Conventional design approaches that omit three-dimensional finite element method (3D FEM) simulations demonstrate inherent limitations in accurately evaluating coil performance metrics and loss mechanisms. More specifically, while 2D FEM can simplify problem complexity and accelerate computation when applied to symmetric cases, it is inherently limited in accurately representing the spatial field distribution when the relative positions of the coils vary in three-dimensional space [11]. In contrast, 3D FEM is capable of solving geometrically complex models and enables more accurate inductance calculations by evaluating magnetic field energy over a defined spatial domain. In terms of the solver implementation, employing commercial software packages like ANSYS or COMSOL to support the design process is a well-established practice. Nevertheless, the application of fully open-source platforms for similar purposes remains limited and underexplored. Consequently, the principal technical challenge lies in establishing a computationally efficient 3D FEM-assisted framework that achieves both precision and expediency in the design-validation cycle.

The proposed methodology implements scripted parametric FEM analysis to enable systematic batch simulations of coil dimensions through controlled geometric increments. Integration with multi-objective optimization algorithms facilitate automated exploration of the design space, targeting optimal compromises between competing objectives of size reduction, weight minimization, and efficiency preservation [12].

1.3 Research questions

1) **How can 3D FEM be used for IPT coil design?**
2) **What kind of workflow can be used to obtain the performance parameters of IPT coils?**
3) **How to integrate multi-objective optimization algorithms with 3D FEM-based performance simulations?**

1.4 Scopes and layout

This study focuses on leveraging open-source 3D modeling and FEM simulation tools to validate the design of geometrically constrained IPT coils, thereby acquiring critical optimization parameters such as secondary coil output power and transmission losses. A systematic iterative design workflow is implemented through geometric dimension variation with controlled step sizes, coupled with multi-objective optimization algorithms for dimensional refinement. The thesis structure is organized as follows:

Chapter 2 introduces the IPT system under Series-Series (S-S) compensation topology. Beginning with foundational IPT principles, this chapter details the unique characteristics of S-S topology and their implications for coil design. Analytical derivations of RLC component specifications are rigorously presented.

Chapter 3 investigates analytical methodologies for AC impedance characterization and loss quantification. By conducting comprehensive circuit analysis of the full IPT system encompassing primary/secondary rectifier-inverter stages, this chapter derives essential AC load parameters and current phasor relationships for subsequent design phases. Concurrently, it establishes analytical frameworks for core loss, skin effect loss, and proximity effect loss calculations under the current topology.

Chapter 4 details the implementation of 3D FEM simulations via SALOME platform, ElmerFEM and multi-objective optimization using Pymoo. A standardized simulation protocol is first defined for fixed-dimension coils, requiring triplicate iterations to evaluate self-inductance, mutual inductance, and AC excitation performance. This workflow is subsequently extended into an automated optimization loop integrating dimensional iteration and Pareto-optimal selection.

Chapter 5 demonstrates the strengths and limitations of the proposed design methodology through the practical application of multi-objective optimization design.

Chapter 6 synthesizes research findings while critically analyzing the methodological scalability to alternative compensation topologies and future development pathways.

2 Theoretical Framework of IPT System

This chapter presents the technical foundations and fundamental principles of IPT relevant to the research, while establishing mathematical derivations for key system parameters under investigation.

2.1 IPT Fundamental

2.1.1 Principles of IPT

Inductive Power Transfer (IPT) is an acronym for Inductive Power Transfer, which achieves electrical power transmission through alternating current-generated alternating magnetic field coupling, representing a form of near-field wireless power transfer. As derived from the Ampere-Maxwell law, alternating electric fields can generate alternating magnetic fields, implying that feeding alternating current into conductors of specific geometries excites alternating magnetic fields. According to Faraday's law of electromagnetic induction, conductors exposed to alternating magnetic fields develop induced electromotive forces (EMF). Consequently, conductors with particular configurations subjected to alternating magnetic fields produce alternating EMF. When these conductors form closed circuits, alternating induced currents are generated, thereby enabling power transmission. As shown in the Figure 3Figure 3 , the left half of the system consists of an Ampère's law coil, while the right half comprises a Faraday's law coil; together, they form a near-field wireless power transmission device.

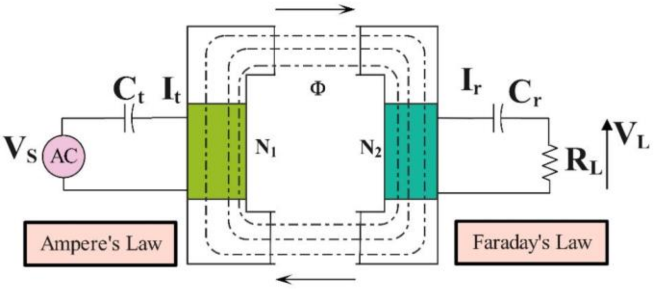


Figure 3 A near-field IPT architecture [13]

V_s represents the modulated voltage source at the transmitter side, typically generating a quasi-sinusoidal waveform with a specified frequency and amplitude. C_t and C_r represents compensation capacitors on the primary and secondary sides. I_t and I_r represent the current in the transmitter coil and the induced current in the receiver coil, respectively.

The transmitted power can be described by Poynting's theorem, where a portion of the energy generated by the source is stored as field energy in the electromagnetic fields, another portion is dissipated as Ohmic losses, and the remaining energy constitutes the transmitted output [7].

$$\iint\limits_{A} (E \times H)\, dA + \iiint\limits_{V} \left(E \cdot \frac{\partial D}{\partial t} + H \cdot \frac{\partial B}{\partial t}\right) dV + \iiint\limits_{V} E \cdot J\, dV = -\iiint\limits_{V} E \cdot J^{ext}\, dV$$

Where $-\iiint_V \; E \cdot J^{ext}\, dV$ denotes the electrical power generated by the power source, $\iiint_V \; E \cdot J\, dV$ represents the Ohmic losses, $\iiint_V \; \left(E \cdot \frac{\partial D}{\partial t} + H \cdot \frac{\partial B}{\partial t}\right) dV$ corresponds to the energy stored in the electromagnetic fields, and $\iint_A \; (E \times H)\, dA$ indicates the instantaneous output power [7].

2.1.2 The transmitter and receiver of IPT.

Transmitter of IPT

The transmitter of IPT typically comprises a set of transmitting coils (with or without a ferromagnetic core) and an associated excitation circuit. This excitation circuit generally consists of an alternating current (AC) power supply with adjustable frequency and amplitude, which can be realized through a combination of AC/DC and DC/AC converters. When a time-varying current is applied to the transmitting coil, it generates a time-varying magnetic field, as governed by the Ampere-Maxwell law:

$$\nabla \times H = \frac{\partial D}{\partial t} + J \qquad (2.1.2\text{-}1)$$

where $\nabla \times H$ denotes the curl of magnetic field intensity describing the circulation characteristics, J represents the current density vector characterizing charge flow per unit area, and D indicates the electric displacement vector [14] [15].

Receiver of IPT

The receiver of IPT typically consists of a set of receiving coil(s) (with or without a ferromagnetic core) and corresponding conversion circuits, which vary depending on the load type at the receiving end. When the load is a battery, the receiver circuit is composed of an AC/DC converter and a DC/DC converter. Exposure of the receiving coil to a time-varying magnetic field induces a time-varying electromotive force (EMF), mathematically expressed by Faraday's law of electromagnetic induction:

$$\nabla \times E = -\frac{\partial B}{\partial t} \qquad (2.1.2\text{-}2)$$

In this context, $\nabla \times E$ represents the curl of the electric field E. If the electric field is generated by static charges, it is a conservative field and its curl is zero [15]. $\frac{\partial B}{\partial t}$ indicating the time rate of change of the magnetic field. The physical implication is that a time-varying magnetic field at a given location induces a circulating electric field in the surrounding space, and the curl of this induced electric field is equal to the rate of change of the magnetic field that generates it [14].

2.1.3 Quality factor and coupling coefficient

Coupling coefficient k

In the design of IPT systems, the geometric configuration of the coils and magnetic cores plays a critical role in determining the efficiency of the power transmission process. This influence is primarily reflected in the self-inductance and mutual inductance of the coils, as well as in the magnetic circuit formed by the core and the air gap.

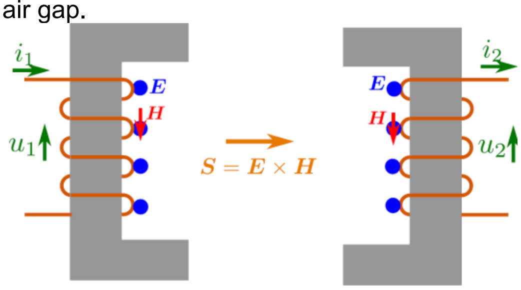


Figure 4 A IPT structure incorporating a magnetic core and an air gap [7]

For a wireless power architecture similar to the one illustrated, the primary and secondary coils can be modeled as inductive components. In circuit schematics, each is typically represented as a series combination of an ideal inductor and an ideal resistor.

Self-inductance characterizes a coil’s ability to EMF in itself in response to a time-varying current, while mutual inductance quantifies the ability of one coil to induce an EMF in another. As a result, both self and mutual inductance directly affect the flux linkage and the induced EMF of the coils during operation.

By introducing the currents and voltages at the primary and secondary sides, the system behavior can be mathematically described using the following expressions based on self and mutual inductance:

$$\begin{bmatrix}\lambda_1\\\lambda_2\end{bmatrix}=\begin{bmatrix}L_1 & -M\\M & -L_2\end{bmatrix}\begin{bmatrix}i_1\\i_2\end{bmatrix};\ \begin{bmatrix}u_1\\u_2\end{bmatrix}=\frac{d}{dt}\begin{bmatrix}\lambda_1\\\lambda_2\end{bmatrix} \qquad \text{(2.1.3-1) [7] [16]}$$

Symbol	Parameter	Unit
$\boldsymbol{\lambda_1}$	Flux linkage of the primary coil (left side)	Wb
$\boldsymbol{\lambda_2}$	Flux linkage of the secondary coil (right side)	Wb
$\boldsymbol{L_1}$	Self-inductance of the primary coil (left side)	H
$\boldsymbol{L_2}$	Self-inductance of the secondary coil (right side)	H
$\boldsymbol{M}$	Mutual inductance between the two coils	H
$\boldsymbol{u_1}$	Excitation voltage of the primary coil	V
$\boldsymbol{u_1}$	Induced EMF of the secondary coil	V

Table 3 List of Symbols

As evident from the mathematical relationships derived above, both self-inductance and mutual inductance play a fundamental role in determining the electromagnetic interaction between the two coupled coils. To characterize the strength of this magnetic coupling, the coupling coefficient k is introduced as follows:

$$k=\frac{M}{\sqrt{L_1L_2}};\ \ where\ 0\le k\le 1 \qquad \text{(2.1.3-1) [7]}$$

When k=1, the system works at perfect coupling (ideal theoretical condition). In this case, all the magnetic flux generated by the primary coil links entirely with the secondary coil. This represents an ideal scenario that is generally unattainable in practical systems.

When k$\ll 1$, Coupling between coils are weak, most of the magnetic flux generated by one coil does not link with the other coil. This indicates inefficient magnetic coupling and poor power transfer efficiency.

When k=0, No coupling (decoupled state). There is no mutual magnetic flux linkage between the two coils. The coils are completely electromagnetically isolated from each other.

Quality factor

The quality factor (Q factor) is a dimensionless parameter that quantifies the energy loss in a resonant system. It represents the ratio of the energy stored in the system to the energy dissipated per cycle. Measuring and calculating the quality factor of an IPT system is useful for quantitatively evaluating its performance during the design process. For example, in the series-compensated circuit configuration considered in this paper, the quality factor Q can be expressed as:

$$Q=\frac{\omega L}{R}$$

In this expression, ω is the electrical angular frequency (i.e., ω=2πf), L denotes the inductance involved in the resonant loop, and R is the equivalent AC resistance, which accounts for conductor losses present in the system.

2.1.4 IPT system with compensation

Since IPT transmits energy through electromagnetic coupling, the large number of coils used as inductive components in the coupling structure significantly increase the reactive power consumption of the system, thereby reducing its energy transfer efficiency. To address this issue, capacitive components are introduced to compensate for the reactive power. In simple terms, this involves adding capacitors to both sides of the IPT coupling circuit. This compensation method gives rise to various circuit topologies, each with distinct effects, depending on factors such as whether the primary and secondary sides are symmetric and whether the components are connected in series or in parallel. The most commonly used compensation topologies include the following four types:

Topology Name	Transmitter-side Compensation	Receiver-side Compensation	Characteristics
Series–Series (S–S)	Series Capacitor	Series Capacitor	Simple structure; easy to design
Series–Parallel (S–P)	Series Capacitor	Parallel Capacitor	Provides constant voltage output
Parallel–Series (P–S)	Parallel Capacitor	Series Capacitor	current-source behavior on receiver side
Parallel–Parallel (P–P)	Parallel Capacitor	Parallel Capacitor	Both sides voltage-source characteristics

Table 4 Compensation Topologies in IPT

From the characteristics of the four compensation topologies described above, it can be concluded that under typical design conditions—where there are no stringent requirements for constant voltage or constant current output—the Series–Series (S–S) topology is the most suitable choice. Schematic of S-S topology shown in Figure 5.

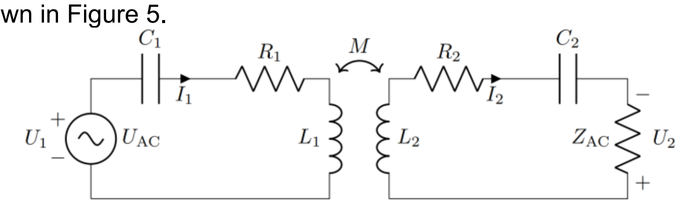


Figure 5 S-S Compensation Topology

U_{AC} denotes the modulated AC voltage source. Components L1 and R1 represent the inductance and internal resistance of the primary coil, respectively, while L2 and R2 denote the inductance and internal resistance of the secondary coil, respectively. The mutual inductance between the two coils is represented by M above. The compensation capacitors, labeled as C1 and C2, are connected in series in the transmitter circuit (containing the primary coil) and the receiver circuit (containing the secondary coil), respectively. Component Z_{AC} represents the equivalent AC impedance of the load. As shown in the Figure 5, the quantitative relationship between the primary and secondary circuits can be derived based on the voltage equations:

$$\begin{bmatrix} U_1 \\ U_2 \end{bmatrix} = \begin{bmatrix} R_1 & 0 \\ 0 & -R_2 \end{bmatrix} \begin{bmatrix} i_1 \\ i_2 \end{bmatrix} + j\omega \begin{bmatrix} L_1 & -M \\ M & -L_2 \end{bmatrix} \begin{bmatrix} i_1 \\ i_2 \end{bmatrix} + \frac{1}{j\omega} \begin{bmatrix} C_1 & \\ & -C_2 \end{bmatrix} \begin{bmatrix} i_1 \\ i_2 \end{bmatrix} \qquad (2.1.4\text{-}1)\ [7]$$

Since $U_2 = i_2 \cdot Z_{AC}$, the expression of the input-side AC impedance can be derived:

$$Z_{in} = \frac{U_1}{i_1} = R_1 + j\omega L_1 + \frac{1}{j\omega C_1} + \frac{\omega^2 M^2}{j\omega L_2 + \frac{1}{j\omega C_2} + R_2 + Z_{AC}} \qquad (2.1.4\text{-}2)$$

From the above expression, it can be seen that when compensation capacitors C1 C2 satisfy specific conditions, the reactive component j in the expression is eliminated, resulting in a purely resistive system and thereby achieving maximum power factor. To achieve this, C1 and C2 must satisfy the following conditions:

$$\omega = \frac{1}{\sqrt{C_1 L_1}} = \frac{1}{\sqrt{C_2 L_2}} \qquad (2.1.4\text{-}3)$$

This given:

$$C_1 = \frac{1}{(2\pi f)^2 L_1};\ C_2 = \frac{1}{(2\pi f)^2 L_2} \quad (2.1.4\text{-}4)$$

Therefore, during the design process, the values of the compensation capacitors can be determined based on the inductance values, which are derived from the coil geometry and the desired output power. The compensated circuit not only improves power transfer efficiency but also reduces the complexity of the design process.

2.2 Initial Design of a 200-Watt-Class IPT Coil

2.2.1 Design Specifications

The initial design discussed in this paper originates from an early-stage technical prototype developed by TILER for one of its product lines. The ultimate goal of this product is to serve as a wireless charging system for electric bicycles, aimed at addressing the challenges associated with the deployment of such systems. One of the product from TILER shows the key idea, see in Figure 6.

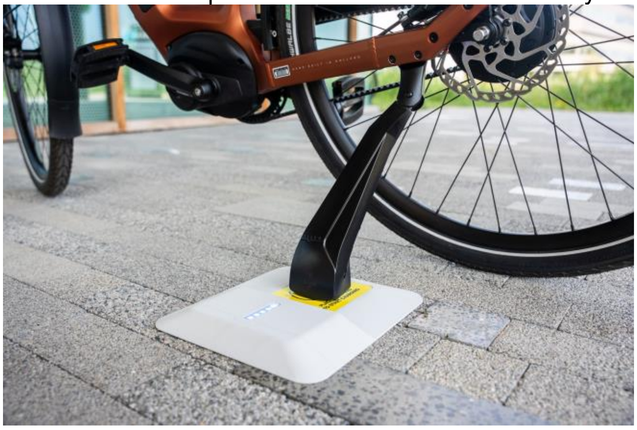


Figure 6 Product of TILER [17]

Conventional wireless charging devices are often constrained by their size and lack of flexibility, making them unsuitable for onboard integration or causing reduced parking convenience due to the large space they occupy during deployment. To minimize the size of the wireless charging system, special attention must be given to the geometric design of both the transmitter and receiver coils. The target system to be optimized is required to meet the following basic design specifications:

Parameter	Value
Output Power Level	150 W
Output DC Voltage	48 V
Switching Frequency	108.5 kHz
Maximum Humidity	95%
Ambient Operating Temperature	-10°C to 85°C
Shock Resistance	Resistant to 300 m/s², 6 ms half-sine shock for 20,000 cycles in X-direction (DIN EN 60068-2-29)
Vibration Resistance	Resistant to Vibration Profile D (LV124)
Winding Wire	110 strands, Ø0.091 mm strand, Grade 1 insulation

Core Material	3C95 Ferrite
Number of Turns	RXC: 20 turns, TXC: 15 turns
Wire Harness Layout	3 wires in parallel

Table 5 Design specifications of IPT parts [9], [10]

2.2.2 Circuit Design and Simulation

Prior to the commencement of this project, Professor Jianning Dong conducted the initial circuit design and simulation of the IPT system based on the specified performance requirements. Using tools such as LTspice and COMSOL, a preliminary design and feasibility analysis were carried out for a series–series (S–S) compensated topology with an output power of 150 W.

By comparing system performance under both aligned and misaligned coil conditions, key characteristics of the charging system were clearly illustrated in terms of system efficiency, VA (volt-ampere) performance, and frequency response analysis. These simulations provided insights into the behavior and limitations of the system under practical deployment scenarios.

The simulation circuit of the IPT system is shown in Figure 7.

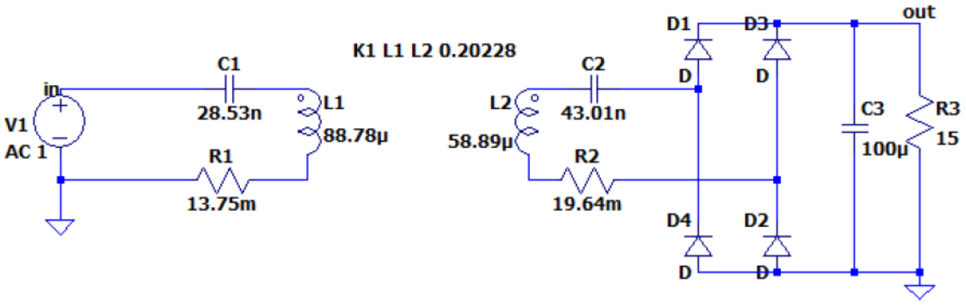


Figure 7 Simulation Circuit of the Initial IPT Design [8]

Under the condition of precise coil alignment, the voltage–current (V–A) characteristics on both the primary and secondary sides, along with the power performance of the system, are shown in Figure 8.

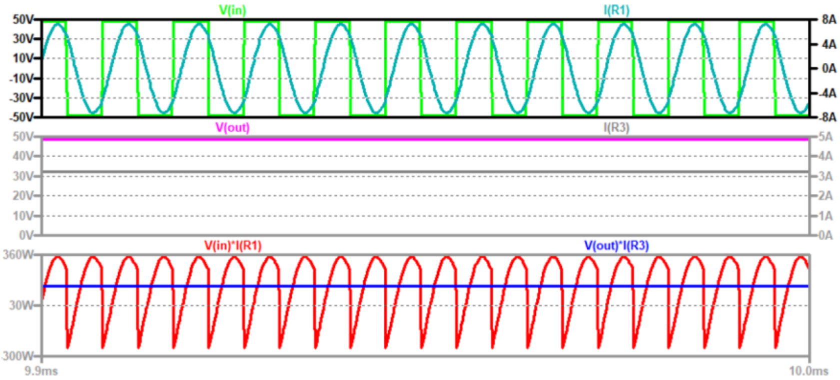


Figure 8 ALIGNED Position Performance of IPT Design 2 [8]

In the absence of core losses (i.e., assuming no magnetic material losses), the system achieves an efficiency of 94.17%, with an input power of 166.35 W and an output power of 156.66 W. The RMS current on the primary side is 5.22 A, while the secondary side exhibits an RMS current of 3.62 A. These simulation results demonstrate that the system operates with high efficiency under ideal alignment conditions, validating the effectiveness of the initial S–S compensated topology at the 150 W power level.

To address the impact of spatial misalignment between the transmitter and receiver coils on the power transfer performance, two design approaches have been proposed, as illustrated in Figure 9 and Figure 10, referred to as Design 1 and Design 2, respectively.

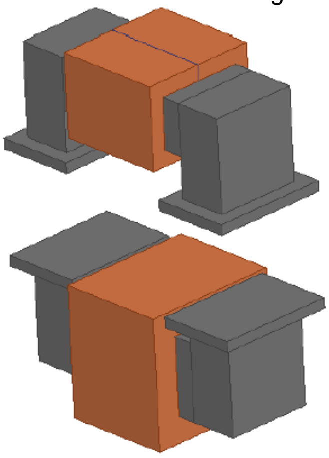


Figure 9 IPT coil Design 1

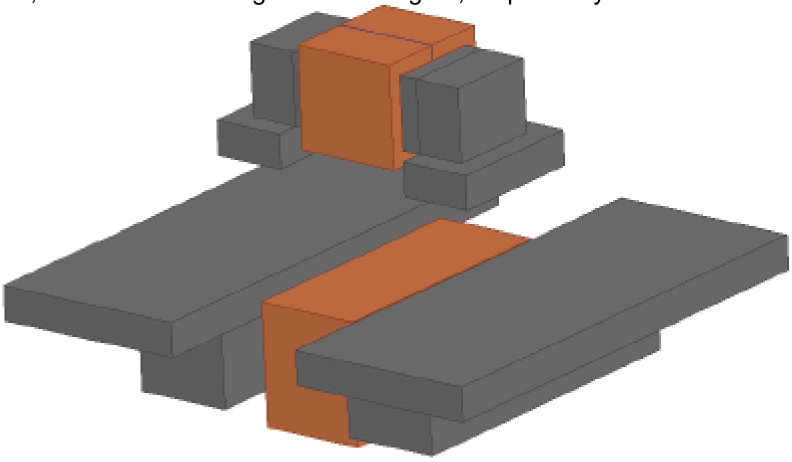


Figure 10 IPT coil Design 2

When the spatial alignment between the transmitter and receiver coils is disturbed, the system characteristics exhibit noticeable changes. Specifically, when the IPT transmitter remains stationary and the receiver is displaced by 25 mm longitudinally and 15 mm laterally, the mutual coupling coefficient decreases from 0.20 to 0.16, as illustrated in Table 7 Coupling Coefficient K with displacement on Design 1

.

K	0	25	forw. Disp (mm)
0	0.15488	0.10571	-31.75%
15	0.1097	0.078609	
side. disp (mm)	-29.17%		-49.25%

Table 6 Coupling Coefficient K with displacement on Design 2 [8]

K	0	25	forw. Disp (mm)
0	0.20228	0.19225	-4.96%
15	0.17043	0.16176	
side. disp (mm)	-15.75%		-20.03%

Table 7 Coupling Coefficient K with displacement on Design 1 [8]

It is worth highlighting that a transmitter-side design with a larger core area demonstrates better tolerance to spatial misalignments compared to design 1(Table 7) where the core areas of the transmitter and receiver are equal. This indicates that a larger transmitter core area provides greater inertia in terms of power transfer efficiency, that is, the efficiency becomes less sensitive to coil displacement.

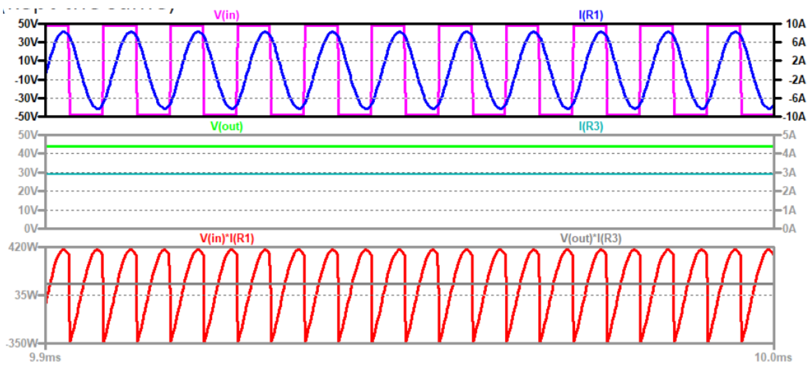


Figure 11 MISALIGNED position Performance of IPT Design 2 [8]

In Design 2, when spatial misalignment occurs between the transmitter and receiver coils, the system experiences a measurable performance degradation, shows at Figure 11. Specifically, under the defined displacement conditions, the following changes were observed:

1) The output voltage decreased by approximately 9.7%, dropping from its original value to 43.99 V. The output voltage of the receiver coil is directly influenced by the mutual inductance M, which decreases with spatial misalignment: $V_{out} \propto M \cdot \omega \cdot {I_t}^2$
2) The power output decreased by approximately 17.63%, reaching 129.04 W. The output power is related to the square of the mutual inductance $P \propto \frac{(M \cdot \omega \cdot I_t)^2}{R}$ Consequently, a reduction in mutual inductance M caused by positional variation leads to a decline in the power delivered to the secondary coil.
3) The power transfer efficiency declined slightly by 1.44%, resulting in an overall efficiency of 92.73%.

3 Coil evaluation method combining FEM and analytical approach

This chapter presents the design and analysis of the IPT system circuit based on the design criteria outlined in Section 2.2.1. The circuit is analytically modeled using fundamental electrical principles, and key performance parameters are calculated from specific design data.
By detailing the calculation methods and solution processes for each performance indicator, this chapter demonstrates the rationale and methodology behind the proposed circuit design.
Following the analytical evaluation, the second part of the chapter describes the simulation and validation of the design using a 3D finite element method (3D-FEM) solver. This simulation process verifies the electromagnetic behavior and performance of the system under realistic physical conditions, thereby validating the analytical results.

3.1 Circuit Design and Analysis

3.1.1 Design and Analysis of the Receiver-Side AC/DC Conversion Circuit

In an IPT system, the coil segment functions solely for power transfer and does not alter the form of energy. As a result, the EMF and induced current on the secondary coil remain in the form of alternating sinusoidal waveforms.
However, in typical applications such as electric bicycles, the onboard energy storage devices require DC for charging. This makes the inclusion of an AC/DC conversion stage essential in the overall system architecture.
The design specifications outlined in Section 2.2.1 are defined in terms of DC charging power and DC output voltage. In contrast, the coil design process primarily involves the AC characteristics of the system. Therefore, a proper translation between AC-side parameters and DC-side requirements is indispensable, and the design of the AC/DC converter must be carefully integrated to ensure that the system meets its intended performance targets.

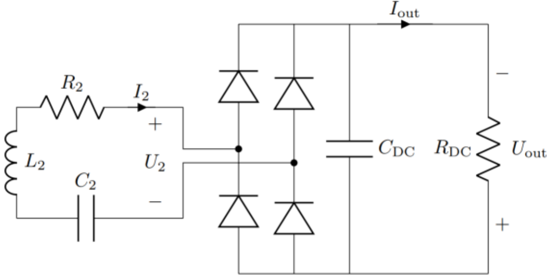


Figure 12 Secondary-side coil circuit with rectification function

In the current application scenario, the IPT system is not required to actively regulate output power or voltage. Therefore, a passive full-bridge rectifier can be adopted to perform the AC/DC conversion efficiently and reliably.
As shown in Figure 12, the complete rectifier circuit is connected to the secondary-side coil, enabling the conversion of the induced AC voltage into a unidirectional DC voltage suitable for charging energy storage devices.

This passive topology offers the advantages of simplicity, low cost, and sufficient performance for systems where output control is not critical. It also minimizes circuit complexity while ensuring adequate conversion efficiency under the given load conditions.

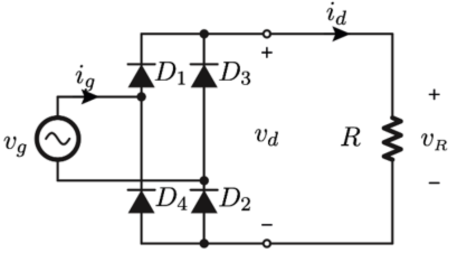


Figure 13 Full-wave diode rectifier with an R load [18]

After introducing compensation capacitors into the coil circuit, the original inductive characteristics can be considered eliminated. The circuit now exhibits purely resistive behavior. Therefore, in the analysis, it can be equivalently represented as a full-wave diode rectifier with an R load, as shown in the Figure 13. v_g represents a sin wave AC voltage input, i_g represents the corresponding AC current. D1 to D4 are four diodes arranged to conduct current unidirectionally during alternating half-cycles. i_d and v_d are the rectified DC current and voltage.

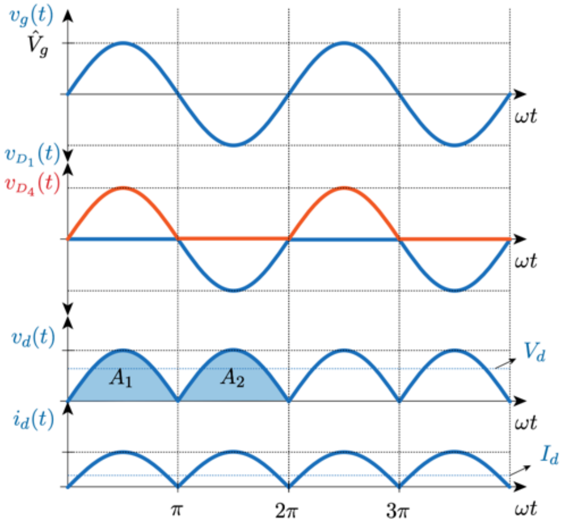


Figure 14 Waveform of Full-wave rectifier with an R load [18]

When the sinusoidal voltage and current pass through the full-bridge section, the unidirectional conduction characteristic of the diodes causes the negative half-cycle of the waveform to be inverted, thereby producing the output waveform shown in the Figure 14.

From the waveform, the output DC voltage and the RMS current can be derived analytically：

$$\mathrm{V_d} = \frac{1}{\mathrm{T}}\int_0^{\mathrm{T}} \mathrm{v_d}(\tau)\,\mathrm{d}\tau = \frac{\mathrm{A1+A2}}{\mathrm{T}} \qquad \text{(3.1.1-1) [18]}$$

From a waveform perspective, this implies averaging the areas A1 and A2 over the 2π interval. The height of the equivalent rectangle formed through this averaging process is the DC output voltage V_d we aim to determine.

$$\mathrm{V_d} = \frac{2}{2\pi}\int_0^{\pi} \widehat{\mathrm{V_g}}\sin(\omega t)\,\mathrm{d}\omega t = -\frac{\widehat{\mathrm{V_g}}}{\pi}\cos(\omega t)\,|_0^{\pi} = \frac{2}{\pi}\widehat{\mathrm{V_g}} \qquad \text{(3.1.1-2)}$$

Similarly, the RMS value of the current can be calculated as follows:

$$\mathrm{I_d} = \frac{2}{2\pi}\int_0^{\pi} \widehat{\mathrm{I_g}}\sin(\omega t)\,\mathrm{d}\omega t = -\frac{\widehat{\mathrm{I_g}}}{\pi}\cos(\omega t)\,|_0^{\pi} = \frac{2}{\pi}\widehat{\mathrm{I_g}} \qquad \text{(3.1.1-3)}$$

However, in our practical application, the voltage across the coil is not a perfect sinusoidal waveform due to modulation techniques. Instead, it resembles a square-like waveform containing multiple harmonic components, as shown in the Figure 15. Therefore, it is necessary to analyze its fundamental component from the perspective of Fourier transformation.

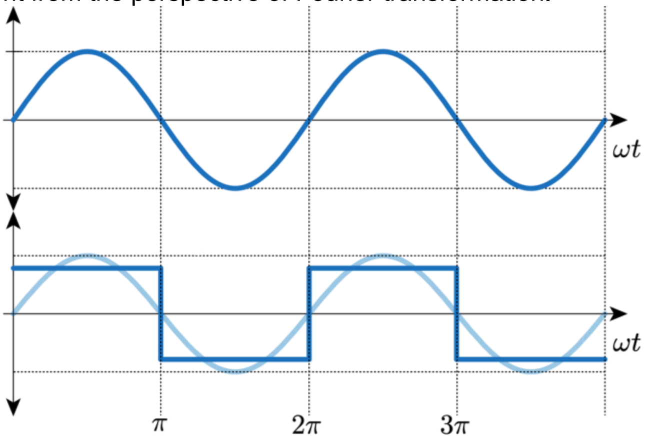


Figure 15 Square-like voltage waveform [18]

Using the Fourier expansion and assuming a square wave with amplitude V_d and a 50% duty cycle, the grid voltage can be represented as:

$$\mathrm{V_g(t)} = \frac{4}{\pi}\mathrm{V_d}\left(\sin(\omega t) + \frac{1}{3}\sin(3\omega t) + \frac{1}{5}\sin(5\omega t) + \cdots\right) \qquad (3.1.1\text{-}4)$$

Thus,

$$\widehat{\mathrm{V_{g1}}} = \frac{4}{\pi}\mathrm{V_d}, \mathrm{V_g} = \frac{2\sqrt{2}}{\pi}\mathrm{V_d} \qquad (3.1.1\text{-}5)$$

Referring to Figure 12,

$$\widehat{\mathrm{U_2}} = \frac{4}{\pi}\mathrm{U_{out}} = \frac{4}{\pi}\cdot 48 = 61.12\ \mathrm{V} \qquad (3.1.1\text{-}6)$$

$$\mathrm{U_2} = \frac{2\sqrt{2}}{\pi}\mathrm{U_{out}} = \frac{2\sqrt{2}}{\pi}\cdot 48 = 43.22\ \mathrm{V} \qquad (3.1.1\text{-}7)$$

$$\widehat{\mathrm{I_2}} = \frac{\pi}{2}\mathrm{I_{out}} = 6.55\ \mathrm{A} \qquad (3.1.1\text{-}8)$$

$$\mathrm{I_2} = \frac{\widehat{\mathrm{I_2}}}{\sqrt{2}} = 4.63\ \mathrm{A} \qquad (3.1.1\text{-}9)$$

3.1.2 Analytical Method for Determining the Load-Side AC Impedance

The DC impedance on the load side is typically straightforward to express. However, when reflected to the coil side as an AC impedance, the analytical calculations discussed in Section 3.1.1 must be taken into account.

Therefore, by referring back to Figure 12, the relationship between the DC and AC impedances can be established, allowing the equivalent AC impedance of the system to be derived. When the output power is 200 W and the output DC voltage is 48 V, the corresponding output current can be calculated as:

$$\mathrm{I_{out}} = \mathrm{I_d} = \frac{\mathrm{P_{out}}}{\mathrm{U_{out}}} = \frac{200}{48} = 4.17\ \mathrm{A} \qquad (3.1.1\text{-}10)$$

Then, R_{AC} present as:

$$\mathrm{R_{AC}} = \frac{\widehat{\mathrm{U_2}}}{\widehat{\mathrm{I_2}}} = \frac{\frac{4}{\pi}\mathrm{U_{out}}}{\frac{\pi}{2}\mathrm{I_{out}}} = \frac{8}{\pi^2}\frac{\mathrm{U_{out}}}{\mathrm{I_{out}}} = \frac{8}{\pi^2}\frac{48}{4.17} = 9.33\ \Omega \qquad (3.1.1\text{-}11)$$

To verify the above calculations, cross-validation can be performed using two different computational algorithms. From the calculated AC impedance, $\mathrm{U_2}$ can be computed as:

$$U_2 = R_{AC} \cdot I_2 = 9.33 \cdot 4.63 = 43.20\,V \qquad (3.1.1\text{-}12)$$

It can be seen that the results in (3.1.1-7) and (3.1.1-12) are match.

3.1.3 Analytical Method for Electromagnetically Coupled Coil Circuits

After reflecting the DC resistance of the AC/DC conversion stage to the coil side as an equivalent AC impedance, the full-bridge rectifier circuit can be temporarily neglected. The magnetically coupled circuit between the primary and secondary coils is shown in the figure.

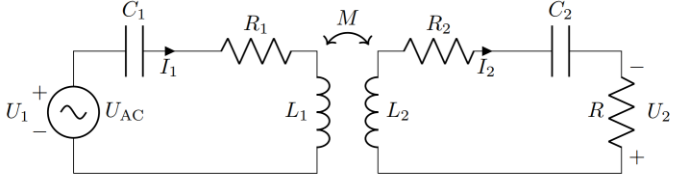


Figure 16 S-S compensated IPT schematic

Based on the voltage equations, the following expressions can be obtained:

$$\begin{cases} U_{AC} = U_{C1} + U_{R1} + U_{L1} - U_{M2} \\ U_{M1} = U_{L2} + U_{R2} + U_{C2} + U_2 \end{cases} \qquad (3.1.3\text{-}1)$$

$$\begin{bmatrix} U_1 \\ U_2 \end{bmatrix} = \begin{bmatrix} R_1 & 0 \\ 0 & -R_2 \end{bmatrix} \begin{bmatrix} i_1 \\ i_2 \end{bmatrix} + j\omega \begin{bmatrix} L_1 & -M \\ M & -L_2 \end{bmatrix} \begin{bmatrix} i_1 \\ i_2 \end{bmatrix} + \frac{1}{j\omega} \begin{bmatrix} C_1 & \\ & -C_2 \end{bmatrix} \begin{bmatrix} i_1 \\ i_2 \end{bmatrix} \qquad (3.1.3\text{-}2)$$

Those two groups of voltage equations lead to:

$$\begin{cases} \underline{U_1} = R_1\underline{I_1} - j\omega M\underline{I_2} + j\omega L_1\underline{I_1} + \frac{1}{j\omega C_1}\underline{I_1} \\ j\omega M\underline{I_1} = R\underline{I_2} + R_2\underline{I_2} + j\omega L_2\underline{I_2} + \frac{1}{j\omega C_2}\underline{I_2} \end{cases} \qquad (3.1.3\text{-}3)$$

According to the S–S compensation capacitor calculation method described in Section 2.1.4 equation (2.1.4-4), when C1 and C2 are selected as their optimal values,

$$C_1 = \frac{1}{\omega^2 L_1};\ C_2 = \frac{1}{\omega^2 L_2} \qquad (3.1.3\text{-}4)$$

This leads to:

$$\begin{cases} j\omega L_1 + \frac{1}{j\omega C_1} = 0 \\ j\omega L_2 + \frac{1}{j\omega C_2} = 0 \end{cases} \qquad (3.1.3\text{-}5)$$

Combine equations (3.1.3-2) and (3.1.3-5), some terms containing the imaginary unit j in the equation can be eliminated, resulting in the following expression:

$$\begin{cases} \underline{U_1} = R_1\underline{I_1} - j\omega M\underline{I_2} \\ j\omega M\underline{I_1} = R\underline{I_2} + R_2\underline{I_2} \end{cases} \quad => \quad \underline{I_1} = \frac{I_2(R+R_2)}{j\omega M} \qquad (3.1.3\text{-}6)$$

Therefore, the transmitter-side voltage can be expressed in terms of the secondary coil current, which in turn can be represented by the equivalent load-side voltage. See in equation (3.1.3-7)

$$\begin{cases} \underline{U_1} = \frac{R_1\underline{I_2}(R+R_2)}{j\omega M} - j\omega M\underline{I_2} \\ \underline{U_2} = R\underline{I_2} \end{cases} \qquad (3.1.3\text{-}7)$$

The relations given in (3.1.3-6) and (3.1.3-7) help clarify the electrical behavior between the two sides of the coil under different design objectives.

3.1.4 Analytical Analysis Based on Different Design Objectives

Regarding the geometric design of IPT coils, different application scenarios entail distinct key performance indicators. This section identifies a set of critical design objectives based on current IPT system applications and provides theoretical analysis and analytical calculations for each.

Misalignment Tolerance of the Coupling Coefficient

The core of this design objective lies in minimizing the reduction of the system's coupling coefficient caused by spatial misalignment between the transmitter and receiver. A design with high resistance to such misalignment can significantly reduce the corrective actions required by users during parking and charging, thereby enhancing user convenience while improving energy transfer efficiency. Based on this, two quantitative evaluation methods can be proposed. The first is the percentage reduction in the coupling coefficient caused by every 10 mm of forward or lateral displacement of the receiver. The second is the ratio of the projected area of the receiver core to that of the transmitter core. The former directly reflects the system's tolerance to spatial misalignment, while the latter indicates the ease of alignment during vehicle parking.

System Input and Output Power

Based on a specified output power, the minimum input power required by the system can be determined through analytical methods. This forms the basis for the design of key coil parameters, including internal resistance, mutual inductance, and turn count.

Power input expressed by primary side as:

$$P_{in} = Re\left\{\underline{U_1 I_1}^*\right\} \quad \text{(3.1.4-1)}$$

Substitute $\underline{U_1}$ and $\underline{I_1}$ from Equation (3.1.3-6) and Equation (3.1.3-7) into Equation (3.1.4-1):

$$\begin{aligned} P_{in} &= Re\left\{\left(\frac{\mathrm{R_1}\underline{\mathrm{I_2}}(\mathrm{R}+\mathrm{R_2})}{\mathrm{j\omega M}} - \mathrm{j\omega M}\underline{\mathrm{I_2}}\right)\cdot\frac{\underline{\mathrm{I_2}}(\mathrm{R}+\mathrm{R_2})}{\mathrm{j\omega M}}\right\} \\ &= Re\left\{\frac{\mathrm{R_1}\underline{\mathrm{I_2}}^2(\mathrm{R}+\mathrm{R_2})^2}{\omega^2\mathrm{M}^2} + \underline{\mathrm{I_2}}^2(\mathrm{R}+\mathrm{R_2})\right\} \\ &= \frac{\mathrm{R_1}}{\omega^2\mathrm{M}^2}(\underline{\mathrm{I_2}}^2\mathrm{R}^2 + 2\underline{\mathrm{I_2}}^2\mathrm{R}\mathrm{R_2} + \underline{\mathrm{I_2}}^2\mathrm{R_2}^2) + \underline{\mathrm{I_2}}^2\mathrm{R} + \underline{\mathrm{I_2}}^2\mathrm{R_2} \quad \text{(3.1.4-2)} \end{aligned}$$

Substitute $P_{out} = \underline{\mathrm{I_2}}^2\mathrm{R}^2$ into (3.1.4-2):

$$\begin{aligned} \mathrm{P_{in}} &= \frac{\mathrm{R_1}}{\omega^2\mathrm{M}^2}(\mathrm{P_{out}R} + 2\mathrm{P_{out}R_2} + \mathrm{P_{out}}\frac{\mathrm{R_2}^2}{\mathrm{R}}) + \mathrm{P_{out}} + \frac{\mathrm{P_{out}R_2}}{\mathrm{R}} \\ &= \frac{2\mathrm{P_{out}R_1R_2}}{\omega^2\mathrm{M}^2} + \frac{\mathrm{R_1}}{\omega^2\mathrm{M}^2}(\mathrm{P_{out}R} + \mathrm{P_{out}}\frac{\mathrm{R_2}^2}{\mathrm{R}} + \frac{\mathrm{P_{out}R_2}\omega^2\mathrm{M}^2}{\mathrm{RR_1}}) \\ &= \frac{2\mathrm{P_{out}R_1R_2}}{\omega^2\mathrm{M}^2} + \frac{\mathrm{R_1P_{out}}}{\omega^2\mathrm{M}^2}(\mathrm{R} + \frac{\mathrm{R_2}^2 + \frac{\mathrm{R_2}\omega^2\mathrm{M}^2}{\mathrm{R_1}}}{\mathrm{R}}) \quad \text{(3.1.4-3)} \end{aligned}$$

It can thus be observed that when the expression inside the parentheses $(\mathrm{R} + \frac{\mathrm{R_2}^2 + \frac{\mathrm{R_2}\omega^2\mathrm{M}^2}{\mathrm{R_1}}}{\mathrm{R}})$ reaches its minimum, the minimum input power required to achieve the given output power can be obtained.

When $R = \sqrt{{R_2}^2 + \frac{R_2\omega^2M^2}{R_1}}$ (3.1.4-4), The minimum value P_{in} can be obtained. Considered that $R_2 \ll R$, The expression in (3.1.4-3) can be rewritten as: $R \approx \omega M\sqrt{\frac{R_2}{R_1}}$.

At this point, a specific analytical relationship among R_1, R_2, and the mutual inductance has been established, which can be used as a constraint for designing the number of turns on both coils.

3.2 Simulation Verification of Coil Design Using 3D FEM

The process of validating the design scheme using a solver begins with the input of core modeling parameters. Based on these parameters, Salome performs parametric modeling by leveraging a series of predefined associative data and subsequently generates the mesh file. This mesh file, together with the SIF (Simulation Input File), serves as the input for ElmerSolver, which then carries out the simulation of the model under specific conditions [19]. The resulting outputs, which include the parameters to be optimized, are passed to the subsequent stages of the workflow. This process is illustrated in the Figure 17.

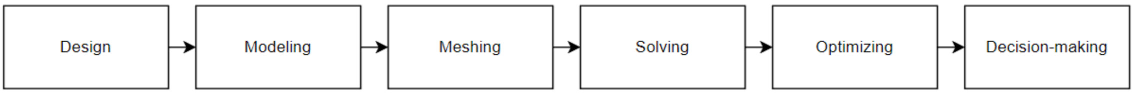


Figure 17 Flowchart of the design

3.2.1 3D Modeling of the Coil Based on SALOME

SALOME (Simulation Architecture for Low-Order Modeling and Engineering) is an open-source, multidisciplinary numerical simulation integration platform jointly developed by the French Alternative Energies and Atomic Energy Commission (CEA), Électricité de France (EDF), and Open CASCADE SAS [20].

Its core objective is to provide a unified framework for modeling, computation, and post-processing of complex physical systems. The platform emphasizes modular design, cross-disciplinary collaboration, and compatibility with high-performance computing (HPC) environments. SALOME has been certified under the ISO 9001 quality management system.

The core functional modules of SALOME include geometry modeling, mesh generation, solver integration, workflow management, post-processing and visualization, and data management. Among these, geometry modeling, mesh generation, and post-processing and visualization played a critical role in the present project [21].

SALOME's key usage in these functionalities can be summarized as follows, shown in Figure 18.

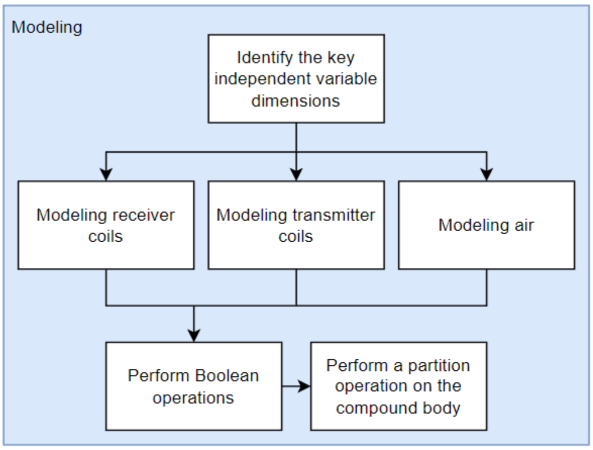
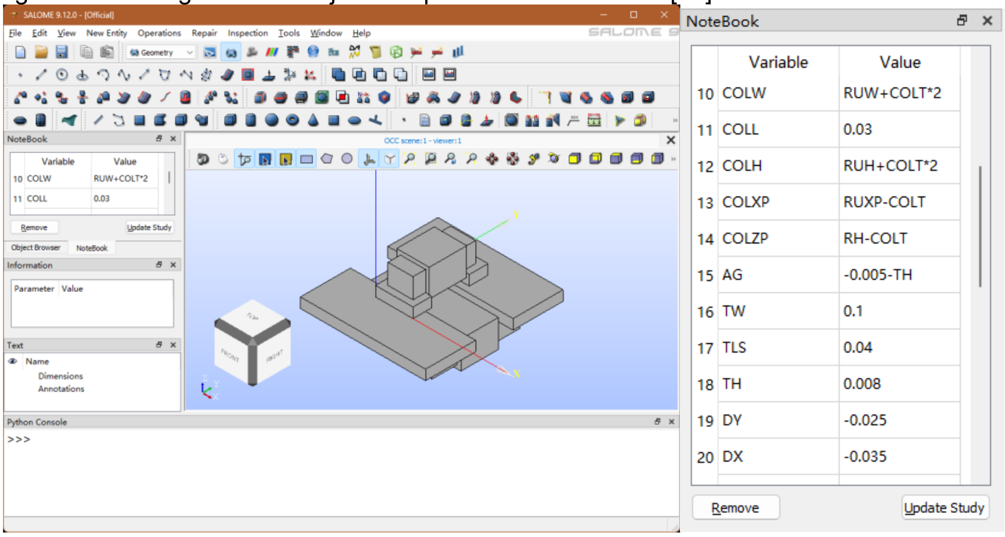

Modeling
Identify the key independent variable dimensions
Modeling receiver coils
Modeling transmitter coils
Modeling air
Perform Boolean operations
Perform a partition operation on the compound body

Figure 18 Flowchart of modeling in SALOME

Geometry modeling

The geometry module of SALOME is built upon the Open CASCADE Technology (OCCT) kernel, with a modeling logic that is similar to that of common commercial CAD software such as SolidWorks and Siemens NX (UG) [22]. It supports parametric modeling through a combination of GUI and TUI interfaces, see in Figure 19 [19]. In particular, the ability to perform parameter-driven modeling via external Python command scripts provides a practical foundation for batch geometry generation, enabling iterative design in multi-objective optimization workflows [22].



NoteBook

	Variable	Value
10	COLW	RUW+COLT*2
11	COLL	0.03
12	COLH	RUH+COLT*2
13	COLXP	RUXP-COLT
14	COLZP	RH-COLT
15	AG	-0.005-TH
16	TW	0.1
17	TLS	0.04
18	TH	0.008
19	DY	-0.025
20	DX	-0.035

Remove　Update Study

Figure 19 SALOME modeling interface (left) and parametric modeling data (right)

Meshing

SALOME's meshing module integrates several engines, including NETGEN, Gmsh, and HOMARD, providing powerful capabilities for mesh generation. It supports adaptive mesh refinement, boundary layer meshing, and the generation of unstructured meshes for complex assemblies. As shown in the Figure 20, the meshing workflow in SALOME is quite intuitive.

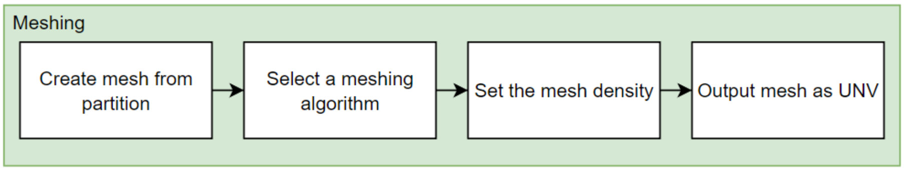


Figure 20 Flowchart of meshing in SALOME

Post-processing

The post-processing module of SALOME is built upon the ParaView/VTK pipeline architecture, enabling compatibility with the output formats of most commonly used solvers.

Thanks to the powerful features of ParaView, the module supports animation generation, quantitative data extraction and computation, as well as identification of various physical characteristics [21].

Modeling in the GUI

The process of geometric modeling using the Graphical User Interface (GUI) in SALOME is similar to that of common CAD software such as SOLIDWORKS and UG NX, allowing users to create models through sketching or operations such as extrusion, translation, and Boolean operations [22].

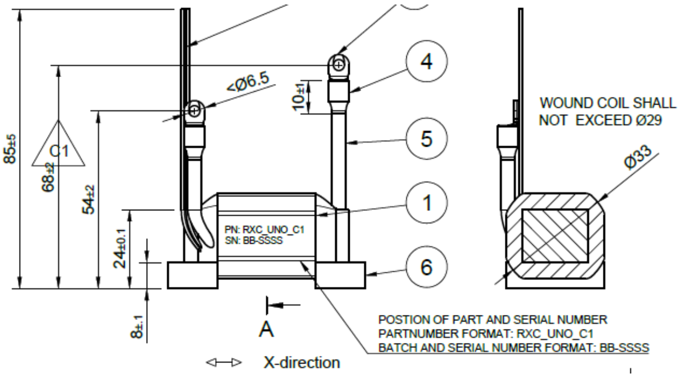


Figure 21 RXC_UNO_C1 Orthographic Projection Drawings [10]

In the optimization design methodology proposed in this work, the initial design is based on the “TXC_uno_C1” and “RXC_uno_C1” models provided by TILER [10], [9],see in Figure 21 . For dimensions not explicitly labeled in certain design figures, references are made to the previous reports by Professor Jianning Dong [23].

The geometry model consists of three major parts: the transmitting coil, the receiving coil, and the air domain. While the first two serve as the physical components of interest and require no further elaboration, the air domain is modeled specifically to capture the near-field magnetic field distribution of the IPT system. Although Elmer supports discontinuous meshing and can internally compensate for missing air regions during simulation, this study opts to explicitly model the air domain to ensure solution accuracy and mesh continuity, spherical part in Figure 22 shows the air modelling.

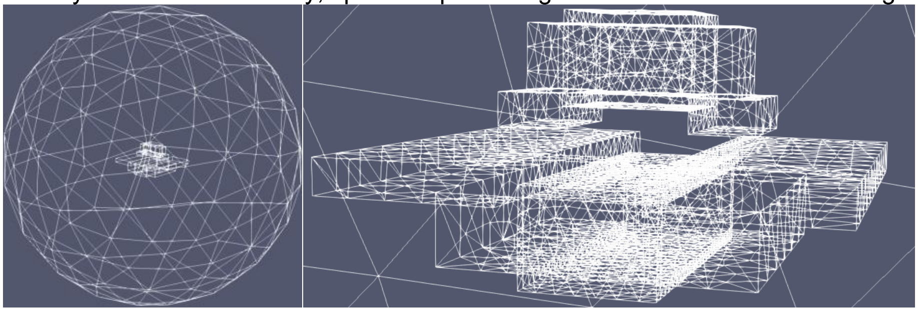


Figure 22 Meshing view to the air and coils

To replicate the aligned coupling condition between the coils, the modeling process reconstructs the working configuration based on the overall external contour. The core components are composed of rectangular solids without curved surfaces, which are created by extruding rectangular sketches and combining them using Boolean operations.

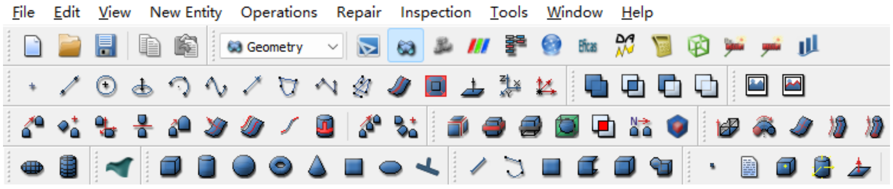


Figure 23 Workspace of SALOME GUI

As shown in Figure 23, when the "Geometry" module is selected in the GUI, the modeling workspace provides functional blocks such as "Basic", "Transformation", and "Boolean Operations", which correspond to sketching, extrusion, and Boolean manipulation tools, respectively [22].

On the left panel, the "NoteBook" tab can be opened to define and manage parametric modeling variables, see in Figure 19. By creating named variables and assigning values, users can build a fully parametric model. Moreover, the values in the NoteBook can be defined as expressions involving other variables, enabling interdependent dimension control within the model [22].

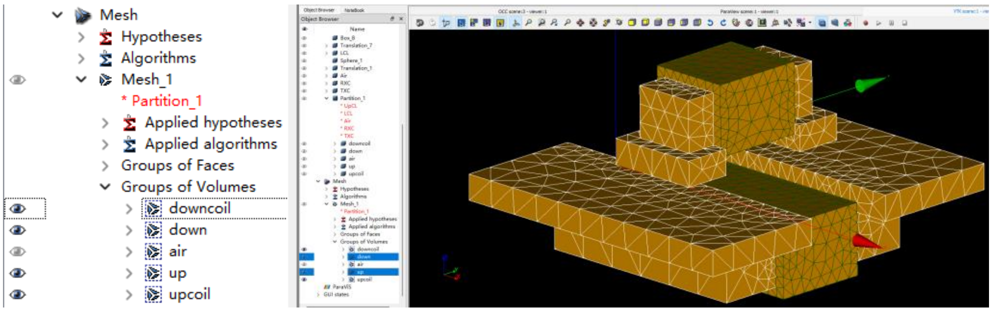


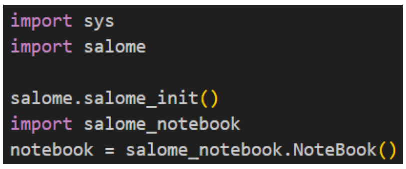
Figure 24 Meshing in SALOME GUI

After completing the modeling process, switch to SALOME's built-in "Mesh" module to generate the mesh based on the geometry. A "Partition" operation must be performed prior to mesh generation. The Partition groups all bodies into a unified structure such that co-planar surfaces in contact are recognized as shared entities, preventing redundant sampling during the meshing process. See Figure 24, By grouping the surfaces and volumes of the mesh model, it becomes more convenient to define materials and assign regions in Elmer during the subsequent simulation process.

Model Reconstruction Using the Python Command Library (TUI)

SALOME's TUI (Text User Interface) operations can be performed by writing Python scripts that are compatible with the software. The execution logic is relatively straightforward.

The first step is to ensure that the system environment is properly set up. This involves importing the necessary system libraries as well as SALOME’s core library, and initializing the SALOME environment, show in Figure 25.

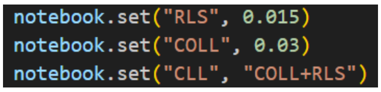


```
import sys
import salome

salome.salome_init()
import salome_notebook
notebook = salome_notebook.NoteBook()
```

Figure 25 SALOME TUI Initialization

After initialization, additional submodules can be imported as needed, depending on the desired functionality. For example, when performing geometry modeling, it is common to import submodules such as GEOM, geomBuilder, and SALOMEDS, which are included in the SALOME library.

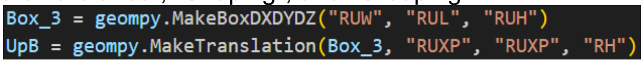
Under the parametric modeling framework adopted in this study, it is recommended to begin by creating a "noteBook" module to ensure the dimension-first principle is followed. The “notebook.set” command is used to define and assign values to dimensional variables required for parametric modeling as shown in Figure 26.



```
notebook.set("RLS", 0.015)
notebook.set("COLL", 0.03)
notebook.set("CLL", "COLL+RLS")
```

Figure 26 Setting Variables in the SALOME Notebook

Once the parameter definitions are complete, the geometry modeling process can be initiated. After initializing the GEOM module, the following steps are carried out in sequence: "Create a new geometry", "Create coordinate axes", "Shaping", and "Grouping".



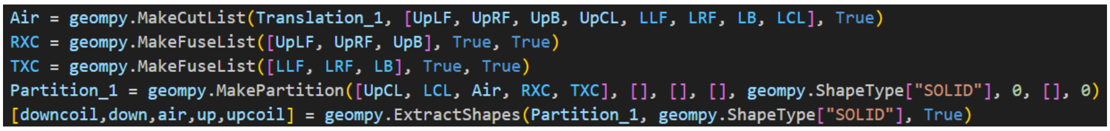
```
Box_3 = geompy.MakeBoxDXDYDZ("RUW", "RUL", "RUH")
UpB = geompy.MakeTranslation(Box_3, "RUXP", "RUXP", "RH")
```

Figure 27 Parametric Modeling

In the GEOM module, operations follow the format: “ObjectName = OperationCommand”. For example, as shown in the Figure 27, the following two commands create a box named Box_3 with length, width, and height assigned from the variables "RUW", "RUL", and "RUH", respectively. The box is then translated along the X, Y, and Z directions by the distances defined by the variables

"RUXP", "RUXP", and "RH". All of these variables were previously defined in the "noteBook" as part of the parametric modeling setup.

After completing the shaping operations, Boolean operations such as union (fuse) and subtraction (cut) are performed to combine or remove specific geometry features. The resulting shapes are then renamed to more intuitive and descriptive names, as illustrated in Figure 28 under the sections "Cutlist" and "Fuselist".



```
Air = geompy.MakeCutList(Translation_1, [UpLF, UpRF, UpB, UpCL, LLF, LRF, LB, LCL], True)
RXC = geompy.MakeFuseList([UpLF, UpRF, UpB], True, True)
TXC = geompy.MakeFuseList([LLF, LRF, LB], True, True)
Partition_1 = geompy.MakePartition([UpCL, LCL, Air, RXC, TXC], [], [], [], geompy.ShapeType["SOLID"], 0, [], 0)
[downcoil,down,air,up,upcoil] = geompy.ExtractShapes(Partition_1, geompy.ShapeType["SOLID"], True)
```

Figure 28 Boolean operation and partition

In the final step, a partition operation is applied to the combined geometry. During this process, the ShapeType is explicitly set to SOLID, as shown in the Figure 28. This ensures that the resulting geometry is interpreted as a solid body, which is essential for subsequent meshing and simulation tasks.

3.2.2 Preprocessing of the 3D Mesh Model Using ElmerFEM

Elmer FEM is an open-source multiphysics finite element analysis software developed by CSC – IT Center for Science in Finland. It supports simulations across various physical domains, including heat transfer, electromagnetics, structural mechanics, and fluid dynamics [24].

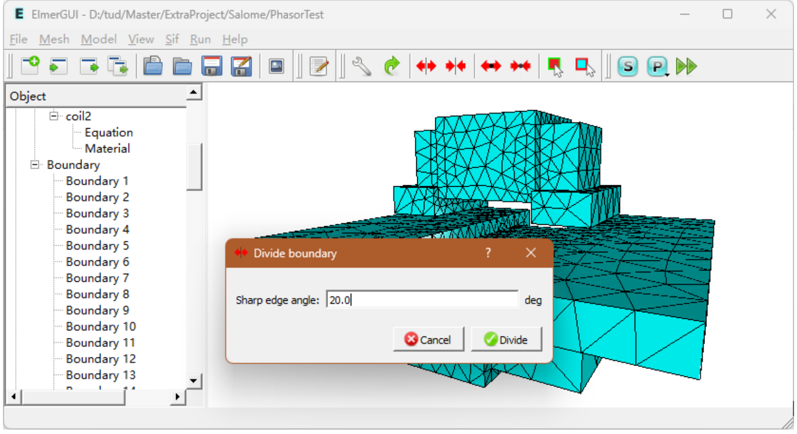
At its core, Elmer is based on the finite element method (FEM) and features a modular architecture. Users can build and solve models either through the graphical user interface (ElmerGUI) or by writing solver input files (SIF) for more advanced control [22].

Elmer supports multiphysics coupling, making it well-suited for the numerical simulation of complex engineering problems. It has been widely adopted in both academic research and engineering applications.

Preprocessing in the GUI

Preprocessing operations are typically performed to ensure compatibility and define boundaries after importing the model and mesh into Elmer. Boundaries can be defined as individual faces or combinations of multiple faces, see in Figure 29. One of the main goals of preprocessing is to clearly identify and assign boundaries, facilitating the subsequent definition of boundary conditions.

In this project, the primary preprocessing task involves identifying the outer surface of the mesh representing air. During the simulation, this outer boundary will be assigned to a far-field condition, representing a zero potential reference.



Sharp edge angle: 20.0 deg

Figure 29 Elmer GUI to divide boundaries

Preprocessing Based on SIF Commands

In Elmer, the SIF (Solver Input File) is the central configuration file that controls the simulation process. It is written in plain text format using a structure similar to INI-style configuration [19].

The SIF file describes essential components of the simulation, including the physical models, material properties, boundary conditions, solver settings, and mesh information.

A SIF file is composed of multiple modular sections, by formalizing these components into a structured process, the resulting flowchart is shown in Figure 30.

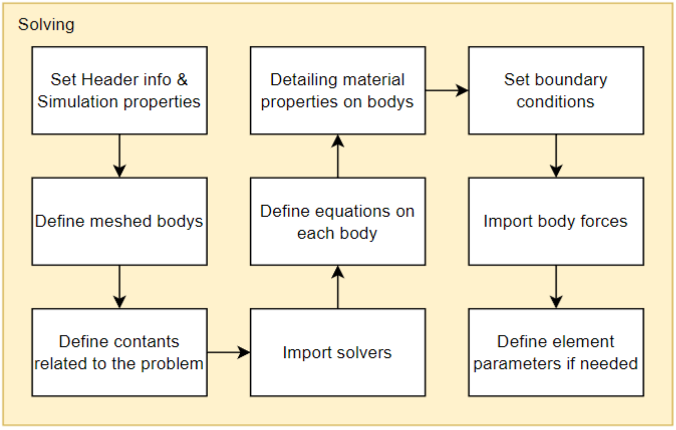

Solving
Set Header info & Simulation properties
Detailing material properties on bodys
Set boundary conditions
Define meshed bodys
Define equations on each body
Import body forces
Define contants related to the problem
Import solvers
Define element parameters if needed

Figure 30 Flowchart of solving a problem in Elmer

Compared to the GUI, the SIF file provides a more detailed and explicit definition of the solver settings and simulation process. In this study, the simulation of the coil using Elmer is primarily carried out by writing and utilizing SIF.

3.2.3 Configuration of the ElmerFEM Solver

Simulation Environment and Basic Physical Parameter Setup

Before configuring any solver or material properties, a global setup for the simulation process must be completed. In the SIF file, this is reflected in the “Header” and “Simulation” sections.

The “Header” section contains key parameters such as Include Path and Results Directory, which allow the solver to specify the directory for importing external files and the directory for storing computed results. If the specified directory does not exist yet, the system will automatically create it.

The Simulation section defines the basic solver settings. The keywords that are directly relevant to the simulations conducted in this study include the following:

Max Output Level: This keyword takes an integer value from 0 to 10 and determines the level of detail in the solver output. The higher the value, the more detailed information the solver will print during the computation process.

Coordinate System: This keyword defines the coordinate system used in the simulation. Three types are supported: Cartesian, Cylindrical, and Spherical coordinate systems.

Simulation Type: This keyword specifies the type of simulation to be performed. It can be set to either Transient (for time-dependent simulations) or Steady State (for time-independent simulations).

Solver Input File: Indicates the name of the SIF file used by the solver. This allows for referencing external or alternative SIF configuration files when needed.

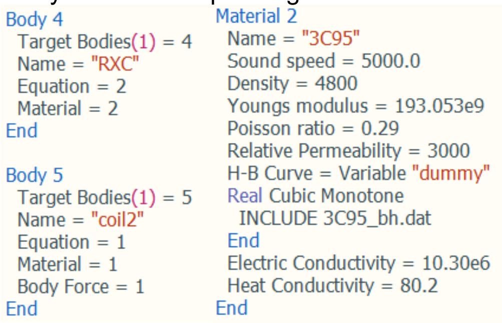
Post File: Defines the name of the output file (usually in “.vtu” format) where the simulation results will be stored for post-processing and visualization.

Mesh level: This field takes a numerical value and determines the mesh density scaling factor. It can be adjusted as needed to refine or coarsen the mesh density.

The “Constants” section is used to define various physical constants involved in the simulation process, such as the vacuum permittivity and vacuum permeability, among others.

Definition of Physical Parameters for Materials

In addition to the global physical constants defined in the environment, individual objects should be assigned to their own material properties. For example, the object representing a coil should be assigned the electromagnetic properties of copper wire, while the core should be assigned the appropriate magnetic permeability of the corresponding material.



```
Body 4
  Target Bodies(1) = 4
  Name = "RXC"
  Equation = 2
  Material = 2
End

Body 5
  Target Bodies(1) = 5
  Name = "coil2"
  Equation = 1
  Material = 1
  Body Force = 1
End
```

```
Material 2
  Name = "3C95"
  Sound speed = 5000.0
  Density = 4800
  Youngs modulus = 193.053e9
  Poisson ratio = 0.29
  Relative Permeability = 3000
  H-B Curve = Variable "dummy"
  Real Cubic Monotone
    INCLUDE 3C95_bh.dat
  End
  Electric Conductivity = 10.30e6
  Heat Conductivity = 80.2
End
```

Figure 31 "Material" section in SIF

As shown in the **Error! Reference source not found.**, in order to distinguish the different parts of a 3D mesh model and their physical meanings, the "Body" section allows you to define the name of each object and assign to them the corresponding "Equation" and "Material".

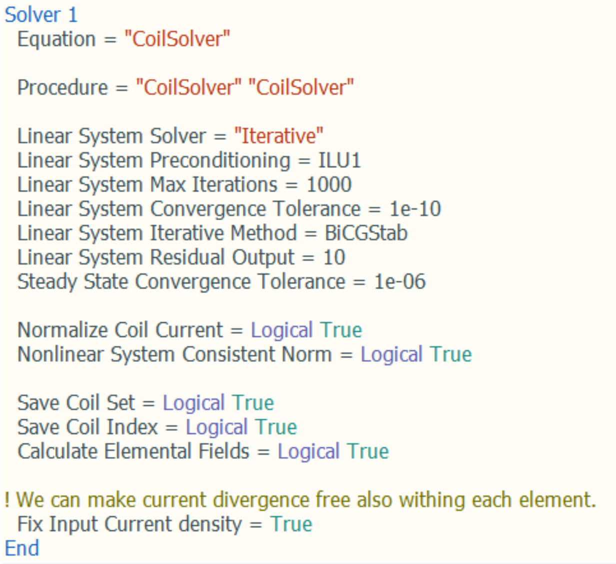
The definition of materials in Elmer is relatively straightforward. The Elmer material library includes several commonly used materials, such as cast iron, copper, and air. However, for more specific materials, the SIF file allows for detailed customization.

As shown in the figure, the core material used in this design is 3C95. Based on its datasheet, we determined several fundamental parameters. However, for electromagnetic simulations, the B-H curve of the material is required. Therefore, we extracted a series of coordinate points from the datasheet and fitted the B-H curve accordingly. These data points were saved in a ".dat" file, which was then imported into the SIF file using the "Include" directive.

Solver Parameter Configuration

The configuration of the Solver section determines the purpose and method of computation during the simulation process. For example, in electromagnetic simulations, solvers such as "WhitneyAVSolver" and "MagnetoDynamicsCalcFields" are used to solve the magnetic field, while "CoilSolver" is employed to define the circular current in coils.

As illustrated in the Figure 31, the Solver section includes the definition of iteration methods, maximum number of iterations, and convergence tolerances. It also contains settings that ensure compatibility between different solvers used in the simulation.



```
Solver 1
  Equation = "CoilSolver"

  Procedure = "CoilSolver" "CoilSolver"

  Linear System Solver = "Iterative"
  Linear System Preconditioning = ILU1
  Linear System Max Iterations = 1000
  Linear System Convergence Tolerance = 1e-10
  Linear System Iterative Method = BiCGStab
  Linear System Residual Output = 10
  Steady State Convergence Tolerance = 1e-06

  Normalize Coil Current = Logical True
  Nonlinear System Consistent Norm = Logical True

  Save Coil Set = Logical True
  Save Coil Index = Logical True
  Calculate Elemental Fields = Logical True

! We can make current divergence free also withing each element.
  Fix Input Current density = True
End
```

Figure 32 Solver section in SIF

Introduction of BodyForce

In Elmer FEM, the Body Force section is used to define volume-distributed source terms acting within the interior regions of geometries. In electromagnetic simulations, the Body Force is primarily used to describe the current density within coils or conductive bodies — a key parameter in establishing the source of the electromagnetic field.

In the simulation of wireless charging coils, excitation is typically applied by assigning an alternating current density within the coil. This current density can be defined in the Body Force section. Elmer supports current density definitions as constants, or as functions of time and space, enabling the modeling of excitations with varying frequencies, waveforms, or spatial dependencies.

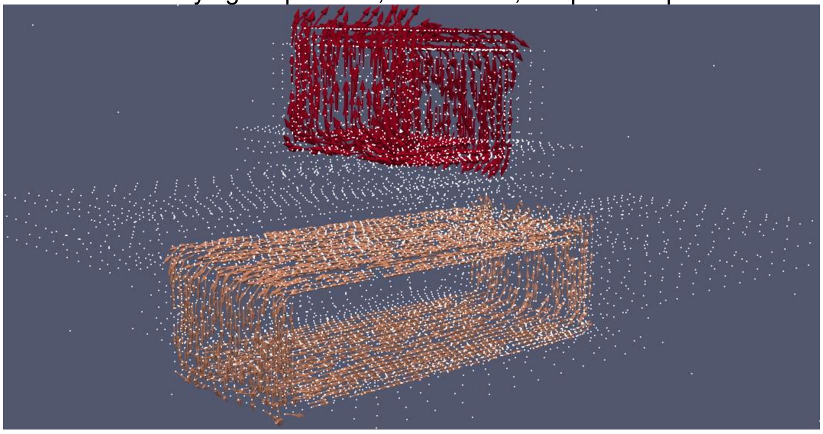


Figure 33 Define coil currents as Body Force

By associating a Body Force with a specific Body section, the current density can be applied precisely to the corresponding coil region in the geometry, as shown in Figure 33. Screenshot in Figure 33 illustrates an example where the excitation current within a closed coil is defined as a Body Force. The color gradient represents the magnitude of the current, with darker shades indicating higher current density. From the solver's perspective, the key parameter is not the total current itself, but rather the current density (i.e., current per unit area). Therefore, special attention must be paid to the type of coil, its cross-sectional area, and the number of turns, as these factors directly affect the resulting current density and, consequently, the behavior of the electromagnetic field. This generates a time-varying magnetic field distribution, which is essential for analyzing key performance indicators such as: Wireless power transfer efficiency, Magnetic coupling strength, Electromagnetic losses Therefore, the Body Force plays a critical role in bridging the physical excitations and field variables in electromagnetic finite element modeling. It provides a powerful and flexible way to accurately define coil excitation conditions in wireless charging systems (i.e., Figure 34).

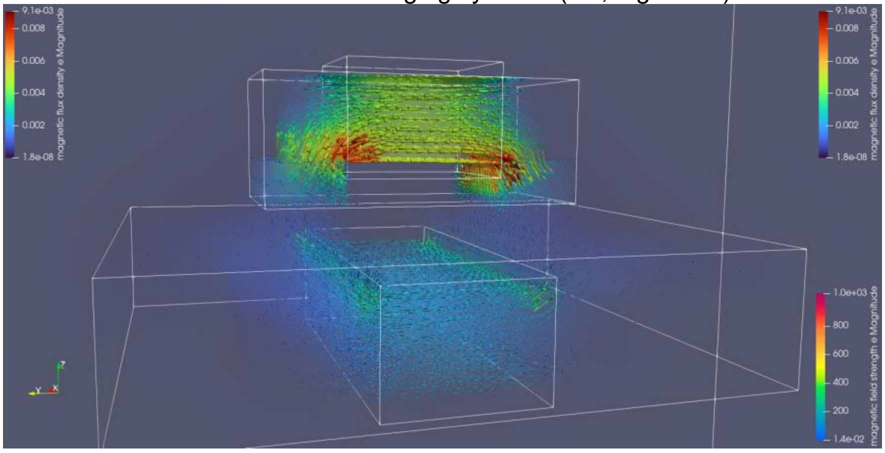

9.1e-03
0.008
0.006
0.004
0.002
1.8e-08
magnetic flux density e Magnitude
9.1e-03
0.008
0.006
0.004
0.002
1.8e-08
magnetic flux density e Magnitude
1.0e+03
800
600
400
200
1.4e-02
magnetic field strength e Magnitude
Z
Y
X

Figure 34 Visualization of magnetic flux density induced by coil current excitation

Figures 30 and 31 clearly show the magnetic flux density field induced by the coil current, which is applied as a body force in the IPT model.

To summarize, a complete workflow has been established, encompassing modeling, meshing, and solving, as shown in Figure 35.

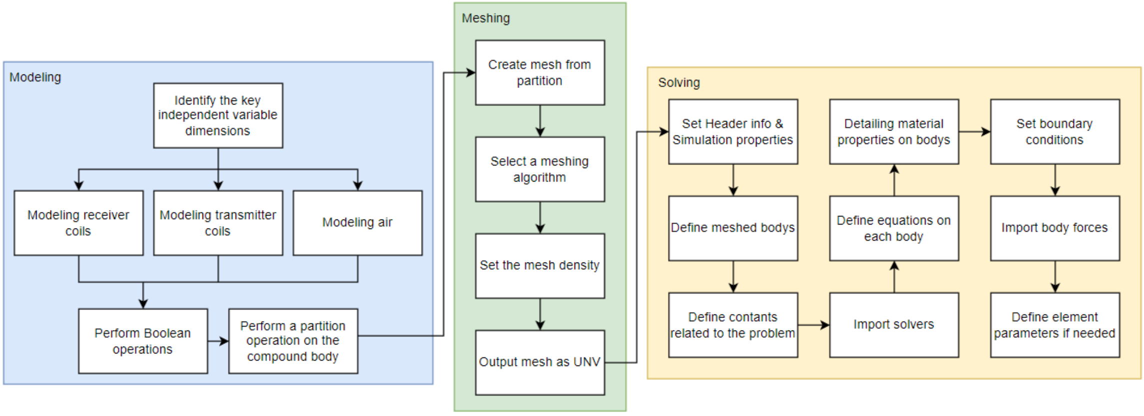

Modeling
Identify the key independent variable dimensions
Modeling receiver coils
Modeling transmitter coils
Modeling air
Perform Boolean operations
Perform a partition operation on the compound body
Meshing
Create mesh from partition
Select a meshing algorithm
Set the mesh density
Output mesh as UNV
Solving
Set Header info & Simulation properties
Detailing material properties on bodys
Set boundary conditions
Define meshed bodys
Define equations on each body
Import body forces
Define contants related to the problem
Import solvers
Define element parameters if needed

Figure 35 Full workflow combines modeling, meshing and solving

3.2.4 Simulation and Analysis of Key Parameters Using ElmerFEM

With the modeling and meshing in Section 3.2.1, as well as the solver configurations in Sections 3.2.2 and 3.2.3, all necessary conditions for performing electromagnetic analysis using 3D finite element methods (FEM) have been established.

Referring back to the analytical approach in Section 3.1, it can be observed that the mutual inductance values in equations (3.1.3–6), which are affected by the coil geometry and typically require empirical measurement, can instead be obtained through a series of calculations or simulations.

While analytical methods are well-suited for calculating the inductance of conventional circular coils, they become significantly more complex and error-prone when applied to non-standard coil shapes, such as the quasi-rectangular coils used in this study.

Computation of Coil Self-Inductance

There are several commonly used methods for calculating the self-inductance of a coil, including the following.

From the perspective of magnetic field and magnetic flux, the self-inductance of a coil can be determined by evaluating the magnetic flux linkage generated by the current flowing through the coil itself. Specifically, self-inductance is defined as the ratio of the total magnetic flux linked with the coil to the current producing it:

$$L = \frac{\phi}{I} = \frac{1}{I^2}\int_V \; B \cdot H \, dV \qquad (3.2.4–1)$$

More generally:

$$L = \frac{\mu_0}{I^2}\iint \frac{J(r)\cdot J(r')}{4\pi|r-r'|} dV \, dV' \qquad (3.2.4–2)$$

This implies that the self-inductance of the coil can be obtained by solving for the magnetic field within the region using Elmer and subsequently performing the required integration over the flux to compute the inductance.

However, this approach is highly sensitive to the mesh density and may introduce considerable numerical errors. Moreover, it requires additional post-processing steps, including the definition of new integration regions. Therefore, an alternative method is to approach self-inductance calculation from the perspective of magnetic energy.

It is well known that an inductor, as an energy storage element, stores energy in the form of a magnetic field. The magnetic energy can be expressed as:

$$W = \int_V \; w \, dV = \int_V \; \frac{1}{2} B \cdot H \, dV \qquad (3.2.4–3)$$

Combine with (3.2.4–1):

$$W = \frac{1}{2} L I^2 \qquad (3.2.4–4)$$

From this, the expression for the self-inductance based on magnetic energy can be derived as:

$$L = \frac{2W}{I^2} \qquad (3.2.4–5)$$

It is important to note, however, that in the modeling process, each individual turn of the coil is not explicitly represented. Instead, the current applied to the "coil" entity corresponds to the total current summed over all turns.

As a result, when calculating the inductance based on magnetic energy, the number of turns must be taken into account, and the formula needs to be modified accordingly:

$$L = \frac{2WN^2}{I^2} \qquad (3.2.4–6)$$

Based on the above considerations, a test plan can be introduced for evaluating the self-inductance within the Elmer simulation environment.

Variable	Value	Sources
TXC current	5.22 Arms	Aligned position data from Design 2 [23]
RXC current	0 Arms	experimental control factor
Relative position	aligned	experimental control factor
Expected L Value	88.78 uH @108.5kHz	Aligned position data from Design 2 [23]

Table 8 Test plan on transmitter-side inductance

Variable	Value	Sources
TXC current	0 Arms	experimental control factor
RXC current	3.62 Arms	Aligned position data from Design 2 [23]

Relative position	aligned	experimental control factor
Expected L Value	58.89 uH @108.5kHz	Aligned position data from Design 2 [23]

Table 9 Test plan on receiver-side inductance

By substituting the above parameters into the inductance formula, the self-inductance of the transmitting coil can be calculated as follows:

Test Conditions				**Results**	**Calculation**	
TXC (A)	RXC (A)	Position	f	ElectroMagnetic Field Energy	Inductance	Error
5.22	0	aligned	DC	5.25E-6	86.7 uH	2.3%
0	3.62	aligned	DC	1.02E-6	62.3 uH	5.4%

Table 10 Test results from Elmer

Based on the results of the two tests, it can be observed that the inductance testing and calculation method based on magnetic field energy has been validated through simulation. The measured self-inductance values of the coils on both sides deviate from the known reference values by only 2.3% and 5.4%, respectively, which can be considered within an acceptable accuracy range.

A further analysis suggests that the potential sources of error may include:

1. Insufficient mesh density, which may have led to limited sampling in the surrounding air region, affecting the accuracy of the magnetic energy calculation.
2. The use of the “Stranded” coil type in Elmer, which might result in a current definition that is not fully consistent with the coil configuration used in Design2 [23].

Computation of Mutual Inductance Between Coils

Similar to the previous section where the self-inductance of a coil was calculated using magnetic field energy, a comparable approach can be applied to determine the mutual inductance between two coils. This method is particularly suitable for coil systems with complex geometries where analytical solutions are not feasible. The fundamental idea is as follows: when two coils carry currents I1 and I2, respectively, the total magnetic energy W stored in the system can be expressed as:

$$W(t) = \frac{1}{2}L_1 I_1(t)^2 + \frac{1}{2}L_1 I_2(t)^2 + M I_1(t) I_2(t) \qquad (3.2.4\text{–}7)$$

By averaging the instantaneous expressions in Equations (3.2.4–7) over one cycle, we obtain:

$$\overline{W} = \frac{1}{2}L_1 {I_{1,rms}}^2 + \frac{1}{2}L_1 {I_{2,rms}}^2 + M I_{1,rms} I_{2,rms} \cos{(\phi)} \qquad (3.2.4.8)$$

After introducing the number of turns in each coil, the magnetic energy expression is modified as follows:

$$\overline{W} = \frac{1}{2}L_1 \frac{{I_{1,rms}}^2}{N_1} + \frac{1}{2}L_2 \frac{{I_{2,rms}}^2}{N_2} + M \frac{I_{1,rms}}{N_1} \frac{I_{2,rms}}{N_2} \cos{(\phi)} \qquad (3.2.4\text{–}9)$$

After transformation, the mutual inductance between the coils can be expressed as below when phase difference set 0:

$$M = \frac{W - W_1 - W_2}{I_1 I_2} N_1 N_2 \qquad (3.2.4\text{–}10)$$

Here, W represents the total magnetic energy when both coils are energized simultaneously, while cases W_1 and W_2 correspond to the magnetic energy when each coil is energized individually, respectively. To verify the accuracy of this calculation method, a series of tests are conducted using Elmer and compared against known reference results. The following test plan is proposed in this study:

Variable	**Value**	**Sources**
TXC current	5.22 Arms	Aligned position data from Design 2 [23]
RXC current	3.62 Arms	experimental control factor

Relative position	aligned	experimental control factor
Expected K Value	Higher than 0.20228	Aligned position data from Design 2 [23]

Table 11 Test plan of mutual inductance

Test Conditions				Results	Calculation	
TXC (A)	RXC (A)	Position	f	ElectroMagnetic Field Energy	Inductance	Judge
5.22	-3.62	aligned	DC	7.97E-6	27.99 uH	Positive

Table 12 Test results of mutual inductance

The calculated coupling coefficient K=0.38, which is higher than the original value of 0.202, aligns well with the expected physical behavior and experimental predictions. The key difference between the two configurations lies in the spacing between the two coils. In the replicated design, the coil-to-coil distance is reduced from 20 mm to 10 mm. While self-inductance remains nearly unchanged, the magnetic flux linkage between the coils increases due to the reduced spacing, resulting in a higher mutual inductance and, consequently, an increased coupling coefficient (according to equation 2.1.3-1).

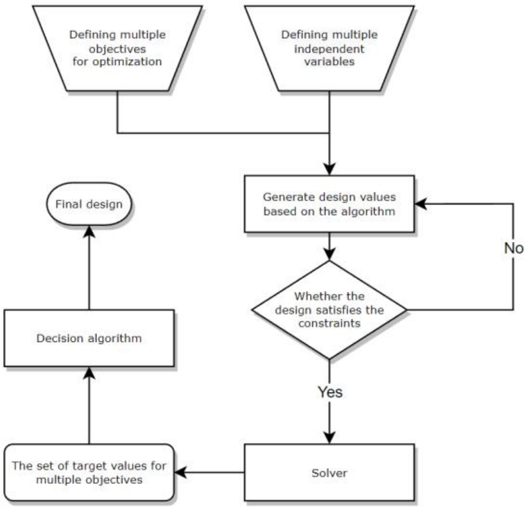
4 Multi-Objective Optimization Algorithms in Coil Design Optimization

The core concept of multi-objective optimization lies in seeking an optimal trade-off among multiple conflicting objectives, aiming to obtain a set of Pareto optimal solutions rather than a single optimal result [25]. These solutions form the so-called Pareto Front, which provides designers with multiple feasible options, enabling them to adjust weights and make decisions based on specific practical requirements.

This chapter provides a detailed discussion of the fundamental principles of multi-objective optimization, commonly used algorithms, and their specific application to the design problem addressed in this study. Practical case studies are also included to demonstrate the value of these methods in enhancing system performance and supporting design decision-making.

4.1 Implementation of Multi-Objective Optimization Using PYMOO

In recent years, with the advancement of computational capabilities and the development of intelligent algorithms, multi-objective optimization methods have been widely applied in various fields such as engineering, electromagnetic design, energy systems, and traffic scheduling [25]. A complete multi-objective optimization process is typically illustrated as shown in the Figure 36.



Defining multiple objectives for optimization
Defining multiple independent variables
Final design
Generate design values based on the algorithm
No
Whether the design satisfies the constraints
Decision algorithm
Yes
The set of target values for multiple objectives
Solver

Figure 36 Flowchart of MOO design method

Representative techniques include evolutionary strategy-based algorithms such as NSGA-II (Non-dominated Sorting Genetic Algorithm II), MOPSO (Multi-Objective Particle Swarm Optimization), and MODE (Multi-Objective Differential Evolution). These algorithms have demonstrated strong global search capabilities and robustness in solving high-dimensional, nonlinear, and non-convex optimization problems [26].

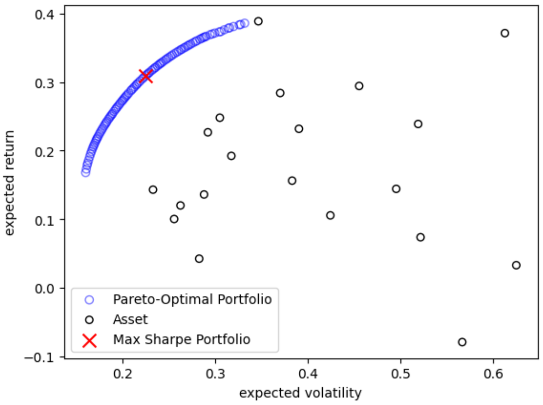


Figure 37 Example of pareto front

Pymoo (Python Multi-Objective Optimization Framework) is an open-source, Python-based library designed for solving complex single-objective and multi-objective optimization problems [26]. The framework integrates a wide range of both classical and modern evolutionary algorithms, such as NSGA-II, NSGA-III, MOEA/D, and CMA-ES, and supports flexible customization of problem definitions, constraints, and performance indicators. It offers excellent extensibility and built-in visualization capabilities.

One of the key advantages of Pymoo lies in its modular architecture, which allows users to easily define optimization problems, select appropriate optimizers, and perform visual analysis of the optimization process. Built-in tools support the plotting of Pareto fronts, convergence trends, and more. Additionally, Pymoo supports continuous, discrete, constrained, and combinatorial optimization problems, making it suitable for a wide range of applications including engineering design, hyperparameter tuning in machine learning, and multi-objective decision making.

4.1.1 Types and Selection of Multi-Objective Optimization Iteration Methods

The mainstream types of multi-objective optimization iterations can be generally categorized into the following three classes:

Population-Based Iteration

Representative algorithms such as NSGA-II, NSGA-III, and MOEA/D employ a population-based evolutionary approach that enables simultaneous exploration of multiple optimal directions within the solution space [27]. This method is particularly well-suited for solving multi-variable, nonlinear, and non-convex optimization problems, such as those encountered in wireless charging coil design. In this study, this class of algorithms is primarily adopted to obtain a more comprehensive set of Pareto-optimal solutions.

Scalarization-Based Iteration

This approach transforms a multi-objective problem into a sequence of single-objective problems using techniques such as weighted sum or the ε-constraint method. While effective for problems with low dimensionality and clear design preferences, it may risk missing solutions when facing non-convex Pareto fronts [28].

Preference-Based Iteration

This method incorporates decision-maker preferences into the optimization process, guiding the algorithm toward regions of interest in the objective space [29]. It is particularly useful during the fine-tuning stage, where quickly identifying solutions that meet engineering-specific requirements is essential [30].

To effectively address the conflicting objectives involved in wireless charging coil design, this study adopts the Non-dominated Sorting Genetic Algorithm II (NSGA-II) as the primary optimization method. NSGA-II is a classical multi-objective evolutionary algorithm known for its strong global search capability and diversity preservation mechanism. It has been widely applied in fields such as engineering optimization, structural design, and electromagnetic systems [31] [32].

The core idea of NSGA-II is to maintain a set of Pareto-optimal solutions across multiple objectives during the evolutionary process by employing non-dominated sorting, crowding distance calculation, and an elitism strategy [31]. The basic workflow of NSGA-II includes:

a) *Population Initialization*: Randomly generate a set of candidate solutions.

b) *Non-dominated Sorting*: Classify solutions into multiple Pareto fronts based on dominance relationships, giving priority to individuals with higher dominance.

c) *Crowding Distance Calculation*: Within the same front, compute the "crowding distance" between solutions to maintain diversity.

d) *Selection, Crossover, and Mutation*: Apply genetic operations to generate the next generation.

e) *Elitism Strategy*: Combine the parent and offspring populations and select the best individuals for the next generation.

In this study, the design objectives for the wireless charging coil include maximizing transmission efficiency, minimizing coil size, and minimizing magnetic field leakage, which are often conflicting. The design variables—such as number of turns, spacing, wire diameter, and layout configuration—exhibit strong nonlinearity. NSGA-II enables global exploration of complex, non-convex search spaces without relying on gradient information, and produces a well-distributed Pareto front, facilitating trade-off analysis among competing objectives [31] [32]

4.1.2 Algorithm Deployment Based on the Pymoo Framework

Pymoo (Python Multi-Objective Optimization Toolbox) is an open-source optimization framework developed in Python, specifically designed for solving multi-objective optimization problems (MOO). Developed by J. Blank and collaborators, Pymoo supports a wide range of both classical and advanced multi-objective evolutionary algorithms, including NSGA-II, NSGA-III, MOEA/D, and SMPSO. It also provides support for single-objective and constrained optimization problems.

Pymoo offers the following key features:

a) Flexible Algorithm Interfaces

b) Strong Customization Capabilities

c) Seamless Simulation Integration

d) Comprehensive Visualization Tools

e) Parallel Computing and History Tracking

Algorithm deployment is carried out from four main dimensions as outlined below:

Problem Definition and Variable Encoding

Multi-objective optimization problems typically involve the simultaneous optimization of two or more conflicting objectives. These objectives are expressed as a vector-valued function, with decision variables forming a vector {x} = {x_1, x_2, ..., x_n}, each constrained within a feasible range defined by problem-specific or engineering limitations.

In Pymoo, problem formulation is achieved by subclassing the “Problem” class, wherein the user specifies the number of decision variables, objectives, and constraints, as well as the logic for evaluating the objective functions. The objective functions may be derived from analytical models, empirical correlations, or numerical simulations, making the framework highly adaptable for both theoretical investigations and engineering applications.

Algorithm Configuration

To effectively solve multi-objective optimization problems, it is essential to select and configure appropriate evolutionary algorithms. Pymoo offers a wide range of established and state-of-the-art algorithms, including NSGA-II, NSGA-III, MOEA/D, and SMPSO, which are well-suited for complex, nonlinear, non-convex, and black-box problems that do not require gradient information.

The parameter settings are determined based on trade-offs between convergence performance, solution diversity, and computational efficiency.

Optimization Procedure and Execution

The optimization process in Pymoo is executed using the “minimize()” function, which governs the iterative evolution of the population. The general workflow consists of the following steps:

1. Generate an initial population of candidate solutions
2. Evaluate the objective values of all individuals
3. Perform non-dominated sorting and compute crowding distances
4. Apply selection, crossover, and mutation to generate offspring
5. Update the population and repeat the process until termination conditions are satisfied

Result Analysis and Visualization

Upon completion of the optimization process, Pymoo provides a suite of tools for analyzing and visualizing the results. These include:

Pareto front plots, which illustrate the trade-off relationships among competing objectives.

Decision space visualizations, which reveal patterns and trends in the distribution of decision variables.

Statistical analysis of the non-dominated solution set to identify representative or preferred designs.

Preference-based filtering, which assists in selecting final solutions according to user-defined criteria.

These tools facilitate a comprehensive understanding of the optimization outcomes and support informed decision-making in multi-objective design processes.

4.2 Application of Multi-Objective Optimization Algorithms in Coil Design

This section demonstrates the applicability and advantages of the proposed method through a multi-objective optimization (MOO) design case study of the transmitter coil.

4.2.1 Selection of Design Objectives

The selection of optimization objectives is typically based on the functional requirements and usage scenarios of the product, or on specific demands from clients. These demands often reflect expectations regarding the product’s competitiveness in the market or the need to address critical pain points in specific industries. For instance, in the case of a wireless charging system designed for outdoor use on 200W-class electric bicycles, key challenges lie in the footprint and deployability of the charging equipment. This necessitates the inclusion of design objectives related to the physical size or weight of the product. Furthermore, in WPT systems, the coil design directly influences both the coupling coefficient and the power transfer efficiency. Based on these considerations, three primary objectives are defined for the optimization process: physical size, coupling coefficient, and transmission efficiency.

Area as a Measure of Size

Design metrics used to represent physical size may include volume, area, or length, depending on the specific characteristics of the object. The selection of such a metric should emphasize the most representative features of the design. In cases where the geometry is highly flexible—i.e., both shape and size can vary freely—the overall dimensions (length, width, and height) change with the design variables. Therefore, volume should be adopted as the size-related objective, as using only area or length may result in the loss of critical geometric information.

Conversely, when certain dimensions of the object are fixed or act merely as constant coefficients—such as a fixed height or width—volume becomes a derivative of area (e.g., volume = area × fixed height), making area a more concise and effective metric to capture the core physical characteristics. Similarly, if two dimensions are constrained, a single key length may suffice to represent size in the optimization process.

In the coil design case presented in this study, the height of the transmitting coil is effectively fixed in order to maintain a flat structure. As a result, the projected area facing the ground becomes the primary factor influencing the overall size, and is therefore selected as the design objective.

Coupling coefficient

The coupling coefficient (K) is a critical parameter in coil design, as it reflects the degree of electromagnetic coupling between the primary and secondary coils. A higher K value can effectively reduce system losses and decrease the system's dependence on the quality factor (Q) of the coils. Moreover, under multiple operating frequencies, a higher coupling coefficient contributes to improved system stability.

Transmission efficiency

Power transfer efficiency directly reflects the system's ability to convert and deliver energy, making it a fundamental performance indicator. For charging applications, minimizing energy loss is essential. In the current system, which employs series–series (S–S) compensation topology, the efficiency can be expressed as follows:

$$\eta = \frac{\omega^2 M^2}{\omega^2 M^2 + 2R_1 R_2 + R_1 R + \frac{R_1 {R_2}^2 + \omega^2 M^2 R_2}{R}} \qquad (4.2.1\text{-}1)$$

Here, R_1 denotes the impedance of the transmitting coil, while R_2 represents the impedance of the receiving coil. These values can be obtained through both direct measurement and analytical calculation. In the initial design documentation, the measured DC resistance of R_1 is 25 ± 5 mΩ, and that of R_2 is 17 ± 3 mΩ.

For the analytical calculation of the excitation coil impedance, the winding loss model for Litz wire as presented in "Modeling and Multi-Objective Optimization of Inductive Power Components" by Jonas Mühlethaler is adopted, leading to the following expression:

$$R_{dc} = \frac{4}{\rho \pi {d_i}^2} \qquad (4.2.1\text{-}2)\ [12]$$

By applying Equations (4.2.1–2) to calculate the DC resistance in the initial design and comparing the results with the measured values, the accuracy of the formulas can be validated. The results are summarized in the following Table 13:

Table 13 DC resistance compute of Litz-Wire Windings

	R1 (mΩ)	R2 (mΩ)
Measured values	25 +- 5	17 +- 3
Calculate values	24.1	18.4

The values obtained from the formulas are consistent with the measured data, thereby confirming the validity of the analytical model.

4.2.2 Identification of Design Schemes and Independent Variables

The coil design consists of two parts: the transmitting coil and the receiving coil. The receiving coil is installed within the frame of the electric bicycle, and therefore its geometry is subject to strict spatial constraints. As a result, this study focuses primarily on the design of the transmitting coil.

The original transmitting coil adopted a "retro telephone" shape similar to that of the receiving coil. However, due to its relatively large height, it could not be laid flat on the ground, limiting its deployability in outdoor environments. To address this issue, the new design decomposes the single cylindrical coil into two disk-shaped coils, significantly reducing the overall height of the component. A magnetic core is placed between the two coils to fill the gap and maintain magnetic continuity.

As shown in the figures below, the coil geometry has been modified from the original design in Figure 38 to the new configuration in Figure 38 original design
Figure 39.

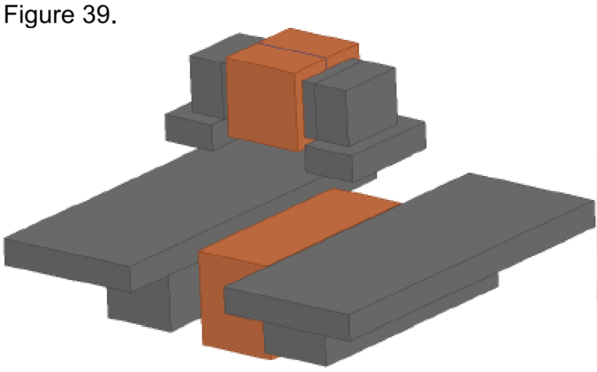


Figure 38 original design

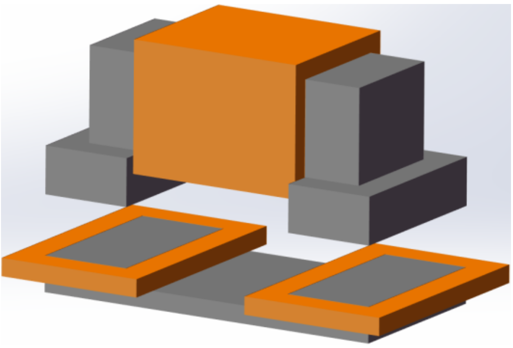


Figure 39 new design

Thanks to the new coil layout, the height of the transmitting coil is now determined solely by the coil thickness and the height of the magnetic core. Under the constraint of constant coil volume, the disk-type configuration significantly reduces the coil thickness, which in turn allows for a reduction in core material usage and contributes to a lighter component weight. This offers a positive impact on the overall system design.

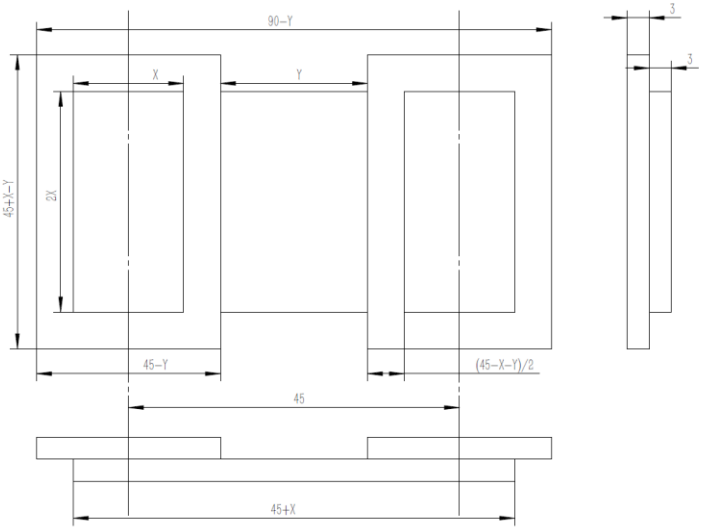


Figure 40 Parametric modeling-based design of transmitting coil geometry

When introducing multi-objective optimization, the core geometric dimensions must be treated as design variables and iteratively updated. Therefore, it is essential to ensure full parameterization of the core geometry during the modeling process. Due to the thickness limitations of the component, the coil thickness and core height are fixed at 3 mm and 6 mm, respectively, and the core's outer profile is set to fill the inner diameter of the coil. The resulting bottom-half design is illustrated in Figure 38 original design Figure 39.

For parametric modeling, this study selects two design variables to define the size and shape of the coil: X, representing the inner diameter of the coil's short edge, and Y, representing the spacing between the two coils. All other geometric parameters can be derived from these two variables.

The design constraint requires that the geometric centers of the terminals at both ends of the transmitting core align with those of the receiving core. This constraint is illustrated in Figure 35, and all lengths can be mathematically expressed.

4.2.3 Determination of Constraints on Design Parameters

In the multi-objective optimization process discussed in the previous sections, constraints constitute an indispensable component. Constraints derived from design criteria or physical limitations serve to define a closed feasible region for the design variables. All sample selection and iterative procedures must be carried out within this constrained domain.

Moreover, the boundaries of the design variable space determined by these constraints may significantly influence the shape and location of the decision boundaries. In this design case study, a set of relevant constraints will be enumerated and analyzed from several perspectives, and these constraints will be quantitatively formulated.

Geometry Constraints

1. Alignment of Coil Centers, the geometric center of the transmitting coil core must align precisely with that of the receiving coil. Therefore, the center-to-center distance between the two coils is constrained to a fixed value.
2. Non-negative constraints on geometric dimensions, all geometric dimensions defined through parametric modeling must be non-negative; that is, each parameter should be strictly greater than zero or, in specific cases, greater than or equal to zero. This gives:

$$\begin{cases} x > 0 \\ y > 0 \\ \frac{45-x-y}{2} > 0 \end{cases} \qquad (4.2.3\text{-}1)$$

3. Incompressibility of Copper Wire, due to the incompressible nature of copper material, the volumetric space allocated for the coil must be greater than or equal to the minimum compressible volume of the copper wire under a given number of turns. In this case, the receiving coil consists of 20 turns, while the transmitting coil consists of 15 turns. Accordingly, when the receiving coil is taken as the reference, the volume of the transmitting coil must be greater than or equal to 75% of the receiving coil's volume. This leads to the following constraint condition:

$$12x^2 + 6xy - 270x + 540y - 6y^2 < 6480 \qquad (4.2.3\text{-}2)$$

Based on the aforementioned constraints, an intuitive visualization of the feasible design variable region can be obtained, as illustrated in Figure 36.

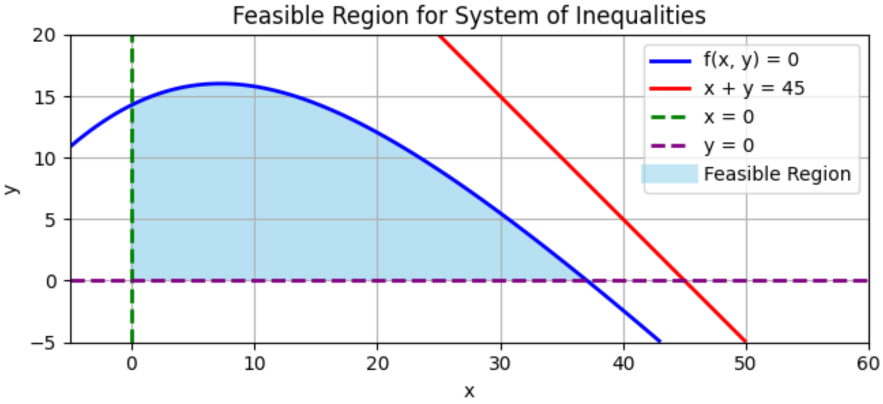


Figure 41 Feasible region for coils' geometry

In the figure, the blue line represents Constraint 3, while the red, green, and purple lines correspond to Constraint 2. The shaded region indicates the feasible design variable domain that satisfies all the imposed constraints.

Physical constraints

From a physical perspective, the imposed constraints are reflected in both the simulation process and the evaluation of the results, as detailed below:

1. Ensuring that the direction of current injected into the coils complies with the magnetic coupling requirements between the transmitting and receiving coils.
2. Ensuring that the outer air boundary satisfies far-field conditions, i.e., the magnetic scalar potential approaches zero at a sufficiently far distance.
3. Ensuring that the magnetic core does not reach magnetic saturation under normal operating conditions.

When the coil current directions are inconsistent, the magnetic fields generated at the transmitting and receiving ends fail to couple effectively. This phenomenon can be clearly observed through field vector visualizations.

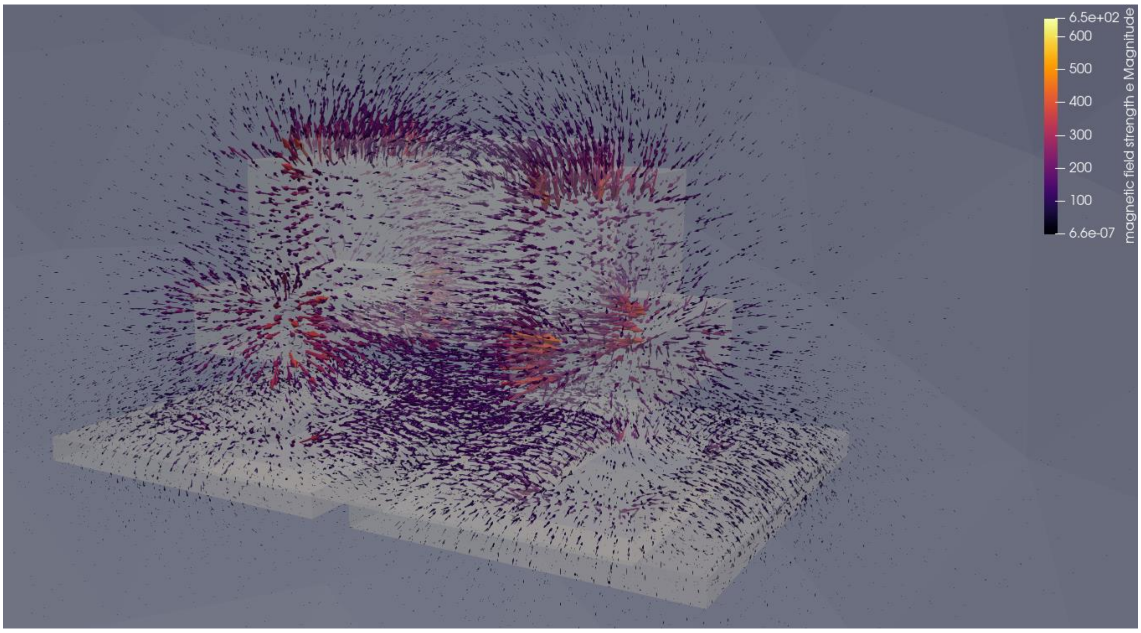


Figure 42 Magnetic field strength vectors under instability

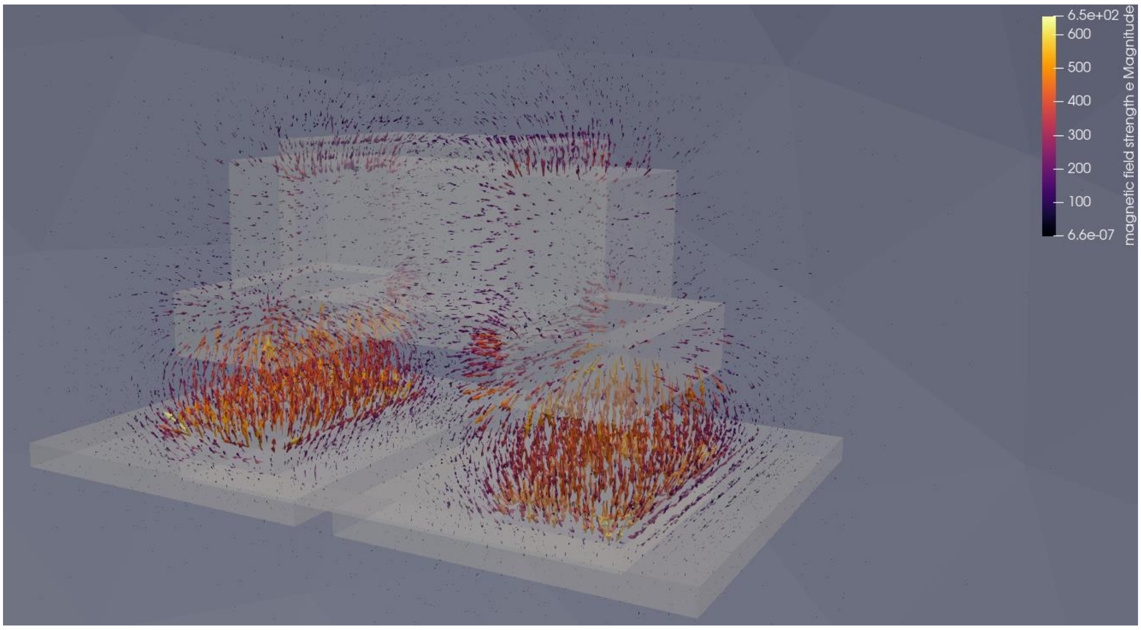


Figure 43 Magnetic field strength vectors under stable coupling conditions

As shown in Figure 42, the magnetic field vectors exhibit a disordered spatial distribution with significant field leakage. In contrast, Figure 43 demonstrates a well-organized field pattern, where the field vectors are concentrated around the air gap near the end faces of the magnetic cores.

4.2.4 Design Iteration and Data Processing

After completing the preparatory work for optimization, we proceed to the formal optimization phase. For this design problem, we adopt the NSGA-II algorithm as previously discussed. A population size of 100 (pop_size=100) is selected, with only 10 offspring generated per generation (n_offsprings=10). This implementation represents a greedier variant of the standard NSGA-II, which can enhance convergence for relatively simple optimization problems without encountering significant difficulties, such as the existence of local Pareto fronts. In addition, duplicate checking is enabled (eliminate_duplicates=True) to ensure that the offspring generated through mating are distinct from both the existing population and each other in terms of their design space values.

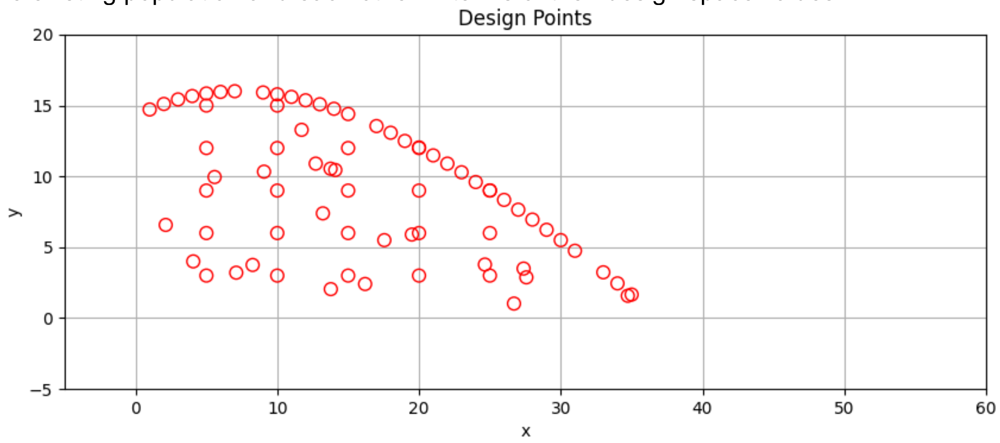


Figure 44 Design Points

As shown in Figure 44, the outline of the evolved population after several iterations resembles the previously defined feasible region based on the design variable constraints. By overlaying the two, we obtain the visualized design space presented in Figure 45. It should be noted that the selected design points shown are not the complete set of initial candidates. During the iterative process, design points that did not satisfy the constraints were eliminated. As a result, the visualization does not include any points outside the feasible region.

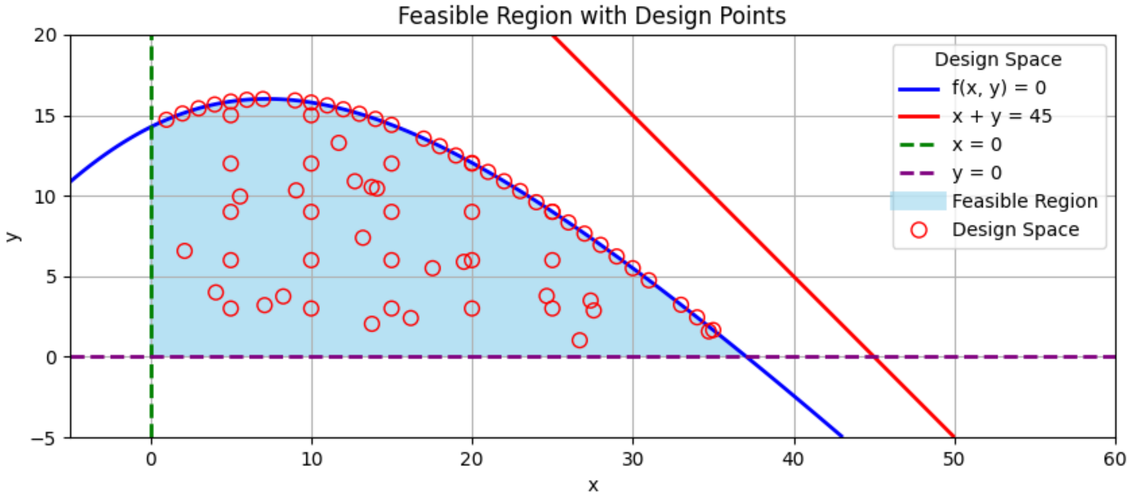
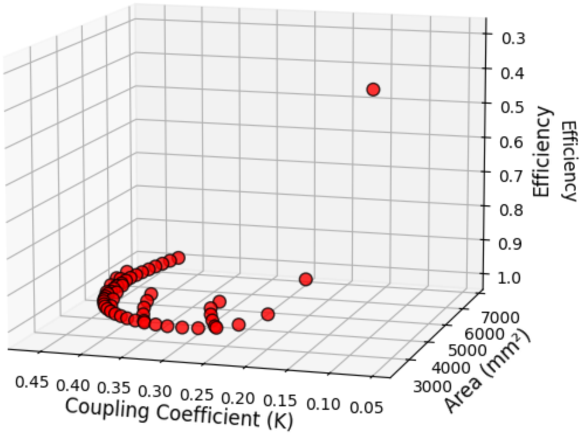


Figure 45 Feasible Region with Design Points

In this design case, three optimization objectives have been previously defined: area, coupling coefficient, and efficiency. Therefore, a three-objective optimization can be performed and visualized. The results can be represented as a spatial scatter plot, forming a surface that reflects the Pareto front. For simplicity, the data set can also be projected pairwise to generate three two-dimensional visualizations, effectively decomposing the problem into three bi-objective subproblems, each yielding a corresponding Pareto front.



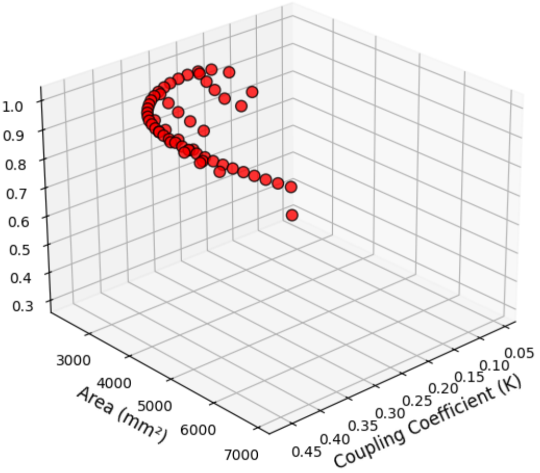


Figure 46 3D Pareto-front another view

Figure 47 3D Pareto-front

Based on the obtained set of results for the three optimization objectives, scatter plots were first generated as shown in Figure 47 and Figure 46 for feature analysis. It can be clearly observed that a well-defined Pareto front emerges in the plots, forming a distinct convex curve. By mapping these result points back to their corresponding design variable configurations, it becomes evident the dominant limiting factor in the optimization process is Geometry Constraint 3, as discussed in Section 4.2.3.

Based on the trend observed in the 3D scatter plot, the design solutions exhibit limited differentiation when efficiency is used as a characteristic. Therefore, it can be anticipated that when efficiency is selected as one of the objectives in a bi-objective optimization, the resulting Pareto front will likely have an irregular shape and low distinguishability. This observation is confirmed by the results shown in Figure 48.

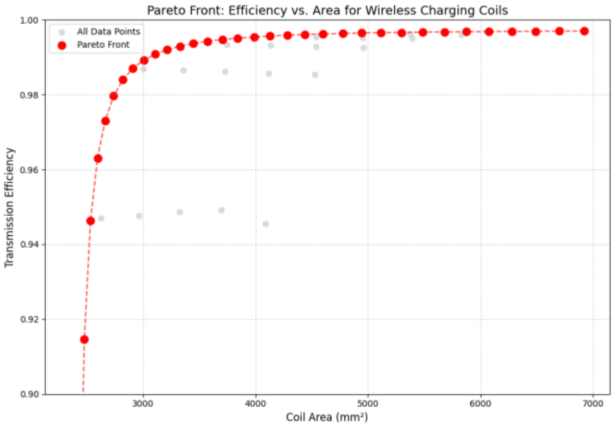


Figure 48 Pareto front of Area vs Efficiency

When the coupling coefficient and area are used as optimization objectives, the resulting Pareto front is smooth and well-defined, as shown in Figure 49.

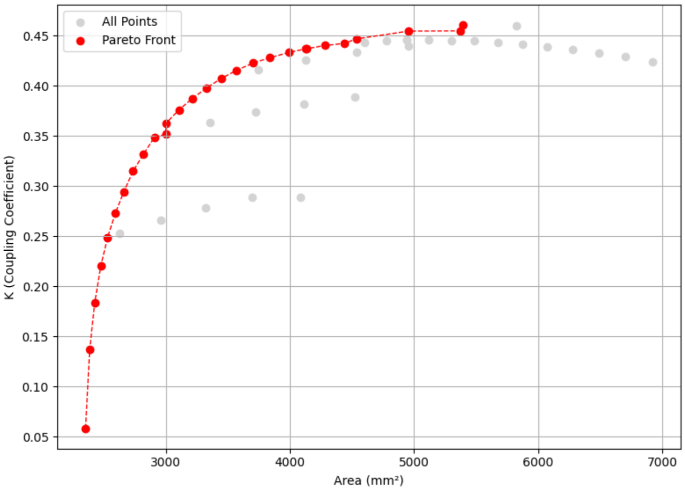


Figure 49 Pareto front of K vs Area

It is worth noting that in the Pareto front shown in Figure 49, although most of the samples on the front appear to lie along the boundary of the design space, this is not entirely the case. In the region where the area exceeds 5000 mm², the Pareto front deviates from the boundary samples of the design space and is instead more significantly influenced by the values of x.

In a standard MOO workflow, the dataset should be normalized to facilitate feature selection and support final weighted decision-making. However, in this section, the original (unnormalized) data is used to provide a more intuitive understanding of the trend relationships among the three objectives.

4.2.5 Objective-Oriented Decision-Making

In the context of the optimization design process, once a dataset representing the Pareto front has been obtained, the primary objective is essentially achieved. Subsequent decision-making often depends on the relative importance of the three design objectives, or even market-driven factors in practical applications. However, since this study focuses specifically on technical considerations, non-technical factors affecting weight assignments are excluded from the current analysis.

Weighted Scoring Decision

The weighted scoring decision method enables the transformation of a three-dimensional decision problem into a one-dimensional one. By assigning weights to the three design objectives and computing a weighted sum, a design score is obtained, and the solution with the highest score is selected as the optimal design.

To ensure the reliability of the decision, it is essential to normalize the dataset prior to scoring. Based on design requirements, weight coefficients are assigned to the three objectives, satisfying the condition:

$$W_A + W_K + W_\eta = 1$$

Based on our prioritization of minimizing physical size, the weights assigned to the three objectives can be specified as follows, $W_A = 0.5$, $W_K = 0.3$, $W_\eta = 0.2$.

After completing the weighted decision-making process described above, the design with parameters x=7 mm and y=16.01 mm achieved the highest overall score. The corresponding design structure is illustrated in the figure below.

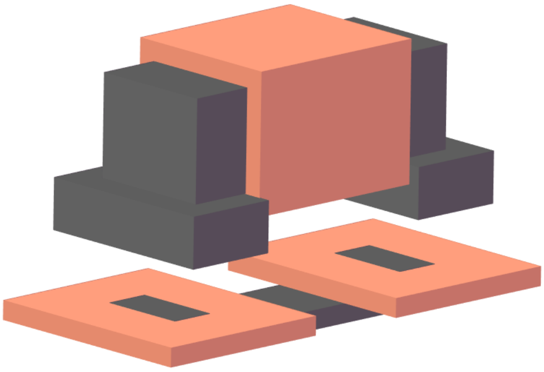


Figure 50 Weighted Decision-Based Optimization Design Scheme

Optimal Approximate Decision

In this study, the optimal approximate solution is identified by evaluating the proximity of each normalized design point to a predefined ideal point, which represents a perfect design scenario (e.g., maximum performance and minimum cost). The Euclidean distance in the normalized objective space is computed for all candidates, and the design with the minimal distance to the ideal point is selected as the most optimal approximation, see Figure 51.

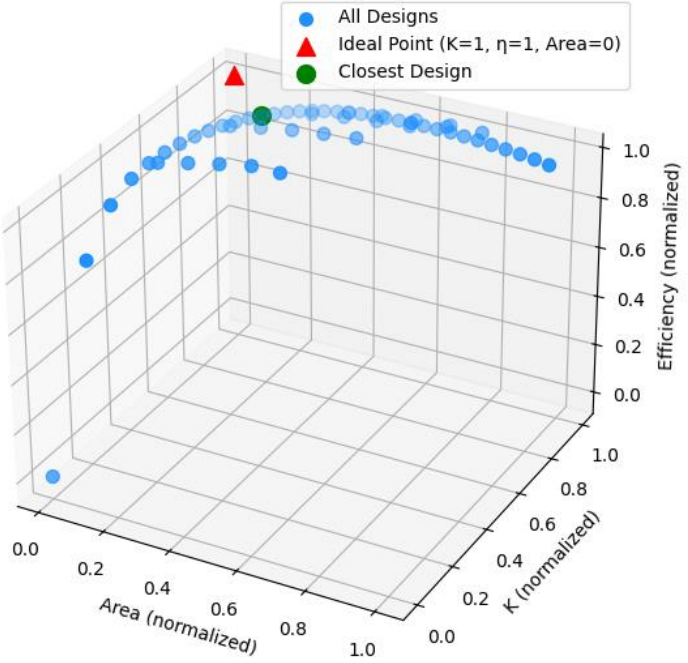


Figure 51 Design Scheme Based on the Optimal Approximate Solution

Based on the visual decision-making approach conducted directly within the three-dimensional normalized objective space, the optimal solution identified is x = 7 mm, y = 16.01 mm. Notably, this result is consistent with the solution obtained through the weighted decision-making method. However, it is important to note that, in theory, these two decision-making approaches may yield different outcomes, particularly when the optimization objectives exhibit substantial differences in scale, sensitivity, or physical meaning. In such cases, the choice of decision-making strategy can significantly influence the final selection, leading to potential deviations in the identified optimal design.

4.2.6 Comparison with original design

After obtaining the optimized design, a quantitative comparison can be conducted between the optimized and original designs in the following aspects. This enables a clearer demonstration of the advantages or limitations of the new design.

Aligned Performance

Under perfectly aligned conditions, both designs exhibit the performance summarized in the table below.

Table 14 Comparison between designs

	Original design	Optimized design	Differences
L_1 (uH)	88.78	2.71	-96.94%
L_2 (uH)	58.89	67.11	+13.96%
M (uH)	14.63	3.97	-72.86%
K	0.202	0.294	+45.54%
Footprint (mm^2)	11000	2662	-75.8%
P_{out} (W)	156.66	200.07	+27.71%
Efficiency	94.17%	97.3%	+2.73%

These results indicate that, after the coil section was optimized, the new design exhibits significant improvements in coupling coefficient, output power, and overall system efficiency under perfectly aligned conditions. In addition, the footprint of the transmitting coil was substantially reduced—by more than 75% compared to the original design.

Misalignment Tolerance

To evaluate the misalignment tolerance of the coils, tests were conducted under the same conditions as used for the original design. Three types of displacements were applied to simulate different misalignment scenarios, and the resulting variations in coupling coefficient were recorded. The conclusions are summarized in the following table (Table 15).

Table 15 Coupling coefficient change by misalignment

K	**0**	**25**	**forw. Disp (mm)**
0	0.294	0.105	-64.28%
15	0.149	0.040	
side. disp (mm)	-49.32%		-86.39%

Compared to the original design, the new design suffers from a much more pronounced performance degradation under misalignment conditions. For the same displacement magnitude, the percentage drop in coupling coefficient (K) is significantly greater in the optimized design than in the original one. By analyzing the structural differences between the two designs, it is hypothesized that this phenomenon is primarily attributed to the magnetic core rather than the geometry of the coils themselves.

To verify this assumption, the magnetic core can be extended upward and laterally, forming a planar structure that covers the footprint of the coil, as shown in Figure 52, followed by a second round of testing to evaluate its impact on misalignment tolerance.

Table 16 Coupling coefficient change under improved design

K	0	25	forw. Disp (mm)
0	0.203	0.120	-40.89%
15	0.153	0.080	
side. disp (mm)	-24.63%		-60.59%

Test results from the improved design show that the coil equipped with a larger-area magnetic core exhibits a smaller percentage decrease in coupling coefficient when subjected to the same spatial displacement.

Although the degradation remains noticeable compared to the original design, this finding supports the previous hypothesis—namely, that the geometry of the magnetic core plays a dominant role in determining the misalignment tolerance of the wireless power transfer system.

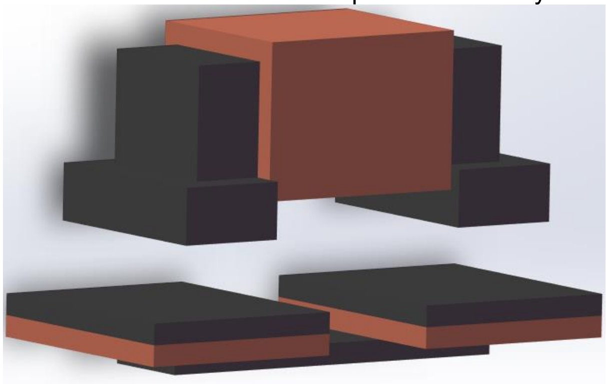


Figure 52 Improved design based on optimized parameter

4.2.7 Conclusion

For a WPT system with a target output of 48 V and 200 W, and under the condition of a fixed receiver coil geometry, a multi-objective optimization was conducted for the transmitter coil. By comprehensively considering the coupling coefficient, component size, and energy efficiency, a locally optimal design was obtained. This design corresponds to the parameterized model with X = 7 mm and Y = 16.01 mm. To enhance misalignment tolerance, the top magnetic core area above the transmitting coil can be expanded as needed.

The convergence of different decision-making strategies to the same optimal result can be primarily attributed to the following factors:

1. In this study, the efficiency objective did not effectively differentiate the data points, thereby limiting its impact as a distinguishing feature in the multi-objective optimization process.
2. The multi-objective optimization (MOO) process employed a fixed number of iterations rather than relying on analytically defined convergence criteria, which may have constrained the exploration of the solution space.
3. Depending on the system's requirements for spatial misalignment tolerance under different operating conditions, the upper structure of the transmitting coil's magnetic core can be incorporated into the optimization process.

5 Conclusion and Future Work

This chapter provides a summary based on the three research questions outlined in Section 1.2, and offers a forward-looking perspective on the future development of the proposed optimization-based design methodology, as demonstrated through the application cases presented in this work.

5.1 How can 3D FEM be used for IPT coil design?

3D FEM can be employed to achieve more accurate analysis and validation of electromagnetic coupling behavior between coils in spatial configurations. This includes, for example, calculating coil self-inductance and mutual inductance using energy-based approaches, or analyzing the EMF on the secondary side.

From a system-level design perspective, 3D FEM offers a clearer and more detailed visualization of vector field distributions compared to 2D FEM or analytical methods. It provides a high density of information, improved data richness, and excellent visualization capabilities.

In terms of design accuracy, 3D FEM outperforms both 2D FEM and analytical models by delivering higher precision and greater convenience in simulation workflows. For non-standard component geometries or irregular coil structures, analytical methods often suffer from significant modeling errors and require cumbersome mathematical derivations, which reduce design efficiency relative to FEM-based methods.

Moreover, 2D FEM has inherent limitations when dealing with asymmetric components—particularly under misaligned coil conditions, as encountered in the present case study. In such scenarios, 2D FEM typically yields lower accuracy compared to 3D FEM, further reinforcing the latter's advantages in precision-critical and spatially complex WPT system designs.

5.2 What Kind of Workflow Can Be Used to Obtain the Performance Parameters of IPT Coils?

For WPT coils with non-typical geometric features in spatial arrangements, key electromagnetic parameters can be extracted through three sequential 3D FEM simulations.

In the first simulation, a predefined current is applied to the transmitter coil, and its self-inductance is calculated using the energy-based method.

In the second simulation, the same approach is applied to the receiver coil, where a specific current is introduced to determine its self-inductance.

Finally, in the third simulation, both coils are simultaneously energized, allowing for the computation of their mutual inductance and the EMF on the secondary coil.

This stepwise simulation procedure enables accurate characterization of the coupling behavior in geometrically complex coil configurations.

5.3 How to Integrate Multi-Objective Optimization Algorithms with 3D FEM-Based Performance Simulations?

The multi-objective optimization (MOO) methodology itself features a relatively complete and closed-loop design process. The integration of 3D FEM-based solvers into the optimization framework primarily relies on the modularization of functional components and the standardization of the simulation workflow.

To ensure proper control of variables during each iteration, it is essential that all configurations—other than the design variables—remain consistent across simulations. This consistency guarantees the generality and comparability of the obtained results.

Specifically, this requirement is reflected in the use of homologous SIF and parameterized geometric models, which together ensure reproducibility and coherence throughout the optimization process.

5.4 Future Development Directions of This Technology

In summary, the MOO-based design methodology for wireless charging coils proposed in this paper—implemented on an open-source platform—has been clearly demonstrated. Nevertheless, the method still holds significant potential for further development and extension in several key directions:

1. Refinement of coil and core geometry: More specific design strategies can be applied to the shape of the coil and magnetic core. For example, raising the end face of the transmitter-side core above the coil plane may enhance the system’s tolerance to lateral misalignment.
2. Expansion of the design space: Introducing additional design variables during parametric modeling can broaden the design space and increase the diversity and flexibility of coil configurations.
3. Incorporation of spatial misalignment factors: Including spatial displacement variables between the two coils in the optimization process can help evaluate and improve the system’s robustness to misalignment.
4. Material property optimization: Material-related variables can be introduced to the optimization framework to identify optimal material selections based on performance and cost trade-offs.
5. Decision-making under high-dimensional objectives: When facing a large number of design objectives, advanced decision-making techniques are required to balance trade-offs effectively.
6. Global design space exploration: Setting convergence thresholds for the MOO algorithm can facilitate broader coverage of the design space, enabling more comprehensive exploration of potential design configurations.
7. Integration of machine learning techniques: In high-dimensional optimization scenarios, machine learning can be employed for feature extraction, pattern recognition, and decision support to further enhance the efficiency and intelligence of the design process.

6 References

[1] X. J. ,. C. W. ,. J. F. C. Z. M. I. a. M. I. M. Guo Wei, "An Automatic Coil Design Method With Modified AC Resistance Evaluation for Achieving Maximum Coil–Coil Efficiency in WPT Systems," *IEEE TRANSACTIONS ON POWER ELECTRONICS,* pp. VOL. 35, NO. 6, 6 2020.

[2] H.Dialani, "Highly Efficient Dual-Side Wireless Power Transfer," Delft University of Technology, Delft, 2023.

[3] P. W. D. N. D. I. K. a. Z. H. X. Lu, "Wireless Charging Technologies: Fundamentals, Standards, and Network Applications,," *IEEE Communications Surveys & Tutorials,* pp. 1413-1452, Secondquarter 2016.

[4] S. M. M. A. S. M. L. M. B. a. W. Y. C. G. Colombo, "Literature Review on Wireless Charging Technologies: Future Trend for Electric Vehicle?," *2022 Second International Conference on Sustainable Mobility Applications, Renewables and Technology (SMART),* pp. 1-5, 2022.

[5] V. Ramakrishnan, "A Comprehensive Review on Efficiency Enhancement of Wireless Charging System for the Electric Vehicles Applications," *IEEE Access,* pp. 46967-46994, 2024.

[6] "INIU-Qi-gecertificeerd-Oplaadstation-Laadstandaard-Laadstation," 31 3 2025. [Online]. Available: https://www.amazon.nl/s?k=wireless+charger&crid=F7Y30PXAN38Z&sprefix=wireless+charger%2Caps%2C75&ref=nb_sb_noss_1.

[7] J. Dong, "EE4C13 Wireless Systems for Electrical Engineering ApplicationsWireless Power Transfer," 2025.

[8] ADVANCED ELECTROMAGNETICS GROUP, "TILER: wireless charging for bikes," 2020.

[9] JORIS（TILER）, "TXC_uno_C1.pdf".

[10] JORIS（TILER）, "RXC_uno_C1.pdf".

[11] R. T. a. S. G. O.C. Zienkiewicz, The Finite Element Method Its Basis and Fundamentals, 2025.

[12] J. MÜHLETHALER, "MODELING AND MULTI-OBJECTIVE OPTIMIZATION OF INDUCTIVE POWER COMPONENTS," ETH ZURICH, Lucerne, Switzerland, 2012.

[13] A. K. M. A. N. S. P. M. B. A. K. A. F. B. AMRITANSH SAGAR, "A Comprehensive Review of the Recent Development of Wireless Power Transfer Technologies for Electric Vehicle Charging Systems," IEEE, 2023.

[14] D. D. Cavallo, "EE4C05 Electromagnetics," Delft University of Technology, Delft, 2023.

[15] I. E. Lager, "Maxwell's equations & electromagnetic waves EE1P21," Delft University of Technology, Delft, 2020.

[16] J. Dong, "ET4121 AC Machines Lecture Note 1: Windings, Flux Linkage and Inductances," Delft University of Technology, Delft, 2023.

[17] Tiler, "TILER Compact: The easiest way to charge your ride," TILER, 2025. [Online]. Available: https://www.tilercharge.com/product. [Accessed 6 2025].

[18] D. S. Rivera, "ET4119 / SET3095 Electronic Power Conversion (EPC)," 2024.

[19] M. M. P. R. Juha Ruokolainen, "ElmerSolver Manual," CSC – IT Center for Science, 2023.

[20] "salome-platform/about," SALOME, 2025. [Online]. Available: https://www.salome-platform.org/?page_id=19. [Accessed 8 7 2025].

[21] R. Bayless, "Get Started with Elmer," ElmerFEM, 2023.

[22] H. Guo, "Electromagnetic Modeling & Simulation of a Coil based on ElmerFEM 2D/3D," 2025.

[23] J. Dong, "TILER: wireless charging for bikes," 2021.

[24] "elmerfem.org/blog/," ELMERFEM, [Online]. Available: https://www.elmerfem.org/blog/. [Accessed 15 6 2025].

[25] S. T. I. R. Mehdi Toloo, Multi-Objective Combinatorial Optimization Problems and Solution Methods, Academic Press, 2022.

[26] J. Blank, "pymoo: Multi-objective Optimization in Python / Algorithms," 2022. [Online]. Available: https://pymoo.org/algorithms/index.html.

[27] S. (. Ventura, Genetic Algorithms, IntechOpen, 2022.

[28] UIUC, "SE 413 Introduction to Scalarization Methods for Multi-objective Optimization," UIUC, 2021.

[29] J. F. E. H. &. S.-H. P. Weiwei Cheng, "Preference-Based Policy Iteration: Leveraging Preference Learning for Reinforcement Learning," *Machine Learning and Knowledge Discovery in Databases,* p. 312–327, 2011.

[30] M. L. H. O. Dexuan Zou, "A non-dominated sorting genetic algorithm III using competition crossover and opposition-based learning for the optimal dispatch of the combined cooling, heating, and power system with photovoltaic thermal collector," *Engineering Applications of Artificial Intelligence,* vol. 128, p. 107607, 2024.

[31] J. P. S. B. N. Rao, "Non-dominated Sorting Genetic Algorithm II and Particle Swarm Optimization for design optimization of Shell and Tube Heat Exchanger," *International Communications in Heat and Mass Transfer,* vol. 132, 2022.

[32] J. C. X. W. F. G. Xue Deng, "Non-dominated sorting genetic algorithm-II for possibilistic mean-semiabsolute deviation-Yager entropy portfolio model with complex real-world constraints," *Mathematics and Computers in Simulation,* vol. 202, pp. 59-78, 2022.

Appendix A: additional info

Function for parametric modeling using SALOME

```
#!/usr/bin/env python
import sys
import salome
salome.salome_init()
import salome_notebook
notebook = salome_notebook.NoteBook()
sys.path.insert(0, r'D:/tud/Master/thesis/Part2')
####################################################
##       Begin of NoteBook variables section       ##
####################################################
notebook.set("x", 7)
notebook.set("y", 16.01)
notebook.set("bcl", "x+45")
notebook.set("bcw", "2*x")
notebook.set("bch", 6)
notebook.set("bccl", "45-x")
notebook.set("bccw", "bcw")
notebook.set("bcch", 3)
notebook.set("ww", "(45-x-y)/2")
notebook.set("dybcc", "ww")
notebook.set("dxbcc", "ww+x")
notebook.set("dxbc", "ww")
notebook.set("dybc", "ww")
notebook.set("cll", "x+2*ww")
notebook.set("clw", "bcw+2*ww")
notebook.set("dxcl2", "2*ww+x+y")
notebook.set("clh", "bcch")
notebook.set("dzbc", "-bch")
notebook.set("dzbcc", "-bcch")
notebook.set("dzcl", "-bcch")
notebook.set("rcl", 60)
notebook.set("rcw", 30)
notebook.set("rch", 8)
notebook.set("ag", 10)
notebook.set("rccl", 30)
notebook.set("rccw", "rcw")
notebook.set("rcch", "rch")
notebook.set("dxrc", "ww+0.5*x-0.5*15")
notebook.set("dyrc", "ww+x-0.5*30")
notebook.set("dxrcc", "dxrc+15")
notebook.set("dyrcc", "dyrc")
notebook.set("rcul", 50)
notebook.set("rcuw", 20)
notebook.set("rcuh", 16)
notebook.set("dxrcu", "dxrc+5")
notebook.set("dyrcu", "dyrc+5")
notebook.set("dzrcu", "ag+rch")
notebook.set("ucll", "uclcl")
notebook.set("uclw", "uclcw+2*ucww")
notebook.set("uclh", "uclch+2*ucww")
notebook.set("ucww", 3)
notebook.set("uclcl", 30)
notebook.set("uclcw", 20)
notebook.set("uclch", 16)
notebook.set("dxuclc", "dxrcu+10")
notebook.set("dyuclc", "dyrcu")
notebook.set("dzuclc", "dzrcu")
notebook.set("dxucl", "dxuclc")
notebook.set("dyucl", "dyuclc-ucww")
notebook.set("dzucl", "dzuclc-ucww")
notebook.set("air", 500)
notebook.set("dxair", "0.5*(4*ww+2*x+y)")
notebook.set("dyair", "ww+x")
######################################################
##        End of NoteBook variables section        ##
######################################################
###
### GEOM component
###

import GEOM
from salome.geom import geomBuilder
import math
import SALOMEDS

geompy = geomBuilder.New()

O = geompy.MakeVertex(0, 0, 0)
OX = geompy.MakeVectorDXDYDZ(1, 0, 0)
OY = geompy.MakeVectorDXDYDZ(0, 1, 0)
OZ = geompy.MakeVectorDXDYDZ(0, 0, 1)
Box_1 = geompy.MakeBoxDXDYDZ("bcl", "bcw", "bch")
Box_2 = geompy.MakeBoxDXDYDZ("bccl", "bccw", "bcch")
geomObj_1 = geompy.MakeTranslation(Box_1, 8.5, 8.5, 0)
geomObj_2 = geompy.MakeTranslation(Box_2, 13.5, 8.5, 0)
geomObj_3 = geompy.MakeCutList(geomObj_1, [geomObj_2], True)
Translation_1 = geompy.MakeTranslation(Box_1, "dxbc", "dybc", "dzbc")
Translation_2 = geompy.MakeTranslation(Box_2, "dxbcc", "dybcc", "dzbcc")
basecore = geompy.MakeCutList(Translation_1, [Translation_2], True)
```

```
Box_3 = geompy.MakeBoxDXDYDZ("cll", "clw", "clh")
Translation_3 = geompy.MakeTranslation(Box_3, 0, 0, "dzcl")
Translation_4 = geompy.MakeTranslation(Box_3, "dxcl2", 0, "dzcl")
coil1 = geompy.MakeCutList(Translation_3, [basecore], True)
coil2 = geompy.MakeCutList(Translation_4, [basecore], True)
Box_4 = geompy.MakeBoxDXDYDZ("rcl", "rcw", "rch")
Translation_5 = geompy.MakeTranslation(Box_4, "dxrc", "dyrc", "ag")
Box_5 = geompy.MakeBoxDXDYDZ("rccl", "rccw", "rcch")
Translation_6 = geompy.MakeTranslation(Box_5, "dxrcc", "dyrcc", "ag")
Cut_1 = geompy.MakeCutList(Translation_5, [Translation_6], True)
Box_6 = geompy.MakeBoxDXDYDZ("rcul", "rcuw", "rcuh")
Translation_7 = geompy.MakeTranslation(Box_6, "dxrcu", "dyrcu", "dzrcu")
upcore = geompy.MakeFuseList([Cut_1, Translation_7], True, True)
Box_7 = geompy.MakeBoxDXDYDZ("ucll", "uclw", "uclh")
Box_8 = geompy.MakeBoxDXDYDZ("uclcl", "uclcw", "uclch")
Translation_8 = geompy.MakeTranslation(Box_7, "dxucl", "dyucl", "dzucl")
Translation_9 = geompy.MakeTranslation(Box_8, "dxuclc", "dyuclc", "dzuclc")
upcoil = geompy.MakeCutList(Translation_8, [Translation_9], True)
Sphere_1 = geompy.MakeSphereR("air")
Translation_10 = geompy.MakeTranslation(Sphere_1, "dxair", "dyair", 0)
air = geompy.MakeCutList(Translation_10, [basecore, coil1, coil2, upcore, upcoil], True)
Partition_1 = geompy.MakePartition([basecore, coil1, coil2, upcore, upcoil, air], [], [], [],
geompy.ShapeType["SOLID"], 0, [], 0)
[Solid_1,Solid_2,Solid_3,Solid_4,Solid_5,Solid_6] = geompy.ExtractShapes(Partition_1,
geompy.ShapeType["SOLID"], True)
[Solid_1, Solid_2, Solid_3, Solid_4, Solid_5, Solid_6] =
geompy.GetExistingSubObjects(Partition_1, False)
[Solid_1, Solid_2, Solid_3, Solid_4, Solid_5, Solid_6] =
geompy.GetExistingSubObjects(Partition_1, False)
[Solid_1, Solid_2, Solid_3, Solid_4, Solid_5, Solid_6] =
geompy.GetExistingSubObjects(Partition_1, False)
geompy.addToStudy( O, 'O' )
geompy.addToStudy( OX, 'OX' )
geompy.addToStudy( OY, 'OY' )
geompy.addToStudy( OZ, 'OZ' )
geompy.addToStudy( Box_1, 'Box_1' )
geompy.addToStudy( Box_2, 'Box_2' )
geompy.addToStudy( Translation_1, 'Translation_1' )
geompy.addToStudy( Translation_2, 'Translation_2' )
geompy.addToStudy( basecore, 'basecore' )
geompy.addToStudy( Box_3, 'Box_3' )
geompy.addToStudy( Translation_3, 'Translation_3' )
geompy.addToStudy( Translation_4, 'Translation_4' )
geompy.addToStudy( coil1, 'coil1' )
geompy.addToStudy( coil2, 'coil2' )
geompy.addToStudy( Box_4, 'Box_4' )
geompy.addToStudy( Translation_5, 'Translation_5' )
geompy.addToStudy( Box_5, 'Box_5' )
geompy.addToStudy( Translation_6, 'Translation_6' )
geompy.addToStudy( Cut_1, 'Cut_1' )
geompy.addToStudy( Box_6, 'Box_6' )
geompy.addToStudy( Translation_7, 'Translation_7' )
geompy.addToStudy( upcore, 'upcore' )
geompy.addToStudy( Box_7, 'Box_7' )
geompy.addToStudy( Box_8, 'Box_8' )
geompy.addToStudy( Translation_8, 'Translation_8' )
geompy.addToStudy( Translation_9, 'Translation_9' )
geompy.addToStudy( upcoil, 'upcoil' )
geompy.addToStudy( Sphere_1, 'Sphere_1' )
geompy.addToStudy( Translation_10, 'Translation_10' )
geompy.addToStudy( air, 'air' )
geompy.addToStudy( Partition_1, 'Partition_1' )
geompy.addToStudyInFather( Partition_1, Solid_1, 'Solid_1' )
geompy.addToStudyInFather( Partition_1, Solid_2, 'Solid_2' )
geompy.addToStudyInFather( Partition_1, Solid_3, 'Solid_3' )
geompy.addToStudyInFather( Partition_1, Solid_4, 'Solid_4' )
geompy.addToStudyInFather( Partition_1, Solid_5, 'Solid_5' )
geompy.addToStudyInFather( Partition_1, Solid_6, 'Solid_6' )
###
### SMESH component
###
import  SMESH, SALOMEDS
from salome.smesh import smeshBuilder
smesh = smeshBuilder.New()
#smesh.SetEnablePublish( False ) # Set to False to avoid publish in study if not needed or in some particular situations:
# multiples meshes built in parallel, complex and numerous mesh edition (performance)
try:
  pass
except:
  print('ExportUNV() failed. Invalid file name?')
#smeshObj_1.SetName( 'airbound' ) ### not created Object
try:
  pass
except:
  print('ExportUNV() failed. Invalid file name?')
try:
  pass
except:
  print('ExportUNV() failed. Invalid file name?')
Mesh_1 = smesh.Mesh(Partition_1,'Mesh_1')
NETGEN_1D_2D_3D = Mesh_1.Tetrahedron(algo=smeshBuilder.NETGEN_1D2D3D)
NETGEN_3D_Parameters_1 = NETGEN_1D_2D_3D.Parameters()
NETGEN_3D_Parameters_1.SetMaxSize( 100 )
```

```
NETGEN_3D_Parameters_1.SetMinSize( 1 )
NETGEN_3D_Parameters_1.SetSecondOrder( 0 )
NETGEN_3D_Parameters_1.SetOptimize( 1 )
NETGEN_3D_Parameters_1.SetFineness( 2 )
NETGEN_3D_Parameters_1.SetChordalError( -1 )
NETGEN_3D_Parameters_1.SetChordalErrorEnabled( 0 )
NETGEN_3D_Parameters_1.SetUseSurfaceCurvature( 1 )
NETGEN_3D_Parameters_1.SetFuseEdges( 1 )
NETGEN_3D_Parameters_1.SetQuadAllowed( 0 )
NETGEN_3D_Parameters_1.SetCheckChartBoundary( 16 )
Solid_1_1 = Mesh_1.GroupOnGeom(Solid_1,'Solid_1',SMESH.VOLUME)
Solid_2_1 = Mesh_1.GroupOnGeom(Solid_2,'Solid_2',SMESH.VOLUME)
Solid_3_1 = Mesh_1.GroupOnGeom(Solid_3,'Solid_3',SMESH.VOLUME)
Solid_4_1 = Mesh_1.GroupOnGeom(Solid_4,'Solid_4',SMESH.VOLUME)
Solid_5_1 = Mesh_1.GroupOnGeom(Solid_5,'Solid_5',SMESH.VOLUME)
Solid_6_1 = Mesh_1.GroupOnGeom(Solid_6,'Solid_6',SMESH.VOLUME)
isDone = Mesh_1.Compute()
[ Solid_1_1, Solid_2_1, Solid_3_1, Solid_4_1, Solid_5_1, Solid_6_1 ] = Mesh_1.GetGroups()
[ Group, Group_0, Group_1, Group_2, Group_3, Group_4, Group_5, Group_6, Group_7,
Group_8, Group_9, Group_10,
 Group_11, Group_12, Group_13, Group_14, Group_15, Group_16, Group_17, Group_18,
Group_19, Group_20, Group_21,
 Group_22, Group_23, Group_24, Group_25, Group_26, Group_27, Group_28, Group_29,
Group_30, Group_31, Group_32,
 Group_33, Group_34, Group_35, Group_36, Group_37, Group_38, Group_39, Group_40,
Group_41, Group_42, Group_43,
 Group_44, Group_45, Group_46, Group_47, Group_48, Group_49, Group_50, Group_51,
Group_52, Group_53, Group_54,
 Group_55, Group_56, Group_57, Group_58, Group_59, Group_60, Group_61 ] =
Mesh_1.GetMesh().FaceGroupsSeparatedByEdges( 30, 0, 1 )
try:
  Mesh_1.ExportUNV( r'D:/tud/Master/thesis/Part2/result1.unv', 0 )
  pass
except:
  print('ExportUNV() failed. Invalid file name?')

## Set names of Mesh objects
smesh.SetName(Group_58, 'Group_58')
smesh.SetName(Group_59, 'Group_59')
smesh.SetName(Group_60, 'Group_60')
smesh.SetName(Group_61, 'Group_61')
smesh.SetName(NETGEN_1D_2D_3D.GetAlgorithm(), 'NETGEN 1D-2D-3D')
smesh.SetName(Group_37, 'Group_37')
smesh.SetName(Group_36, 'Group_36')
smesh.SetName(Group_33, 'Group_33')
smesh.SetName(NETGEN_3D_Parameters_1, 'NETGEN 3D Parameters_1')
smesh.SetName(Group_32, 'Group_32')
smesh.SetName(Group_26, 'Group_26')
smesh.SetName(Group_35, 'Group_35')
smesh.SetName(Group_27, 'Group_27')
smesh.SetName(Group_34, 'Group_34')
smesh.SetName(Group_29, 'Group_29')
smesh.SetName(Group_28, 'Group_28')
smesh.SetName(Group_22, 'Group_22')
smesh.SetName(Group_31, 'Group_31')
smesh.SetName(Group_23, 'Group_23')
smesh.SetName(Group_30, 'Group_30')
smesh.SetName(Group_24, 'Group_24')
smesh.SetName(Group_25, 'Group_25')
smesh.SetName(Group_18, 'Group_18')
smesh.SetName(Group_19, 'Group_19')
smesh.SetName(Group_20, 'Group_20')
smesh.SetName(Group_21, 'Group_21')
smesh.SetName(Mesh_1.GetMesh(), 'Mesh_1')
smesh.SetName(Group_57, 'Group_57')
smesh.SetName(Group_56, 'Group_56')
smesh.SetName(Group_46, 'Group_46')
smesh.SetName(Group_51, 'Group_51')
smesh.SetName(Group_47, 'Group_47')
smesh.SetName(Group_50, 'Group_50')
smesh.SetName(Group_49, 'Group_49')
smesh.SetName(Solid_1_1, 'Solid_1')
smesh.SetName(Group_48, 'Group_48')
smesh.SetName(Group_55, 'Group_55')
smesh.SetName(Solid_3_1, 'Solid_3')
smesh.SetName(Group_54, 'Group_54')
smesh.SetName(Group_40, 'Group_40')
smesh.SetName(Solid_2_1, 'Solid_2')
smesh.SetName(Group_53, 'Group_53')
smesh.SetName(Group_41, 'Group_41')
smesh.SetName(Solid_5_1, 'Solid_5')
smesh.SetName(Group_52, 'Group_52')
smesh.SetName(Group_38, 'Group_38')
smesh.SetName(Solid_4_1, 'Solid_4')
smesh.SetName(Group_39, 'Group_39')
smesh.SetName(Group_44, 'Group_44')
smesh.SetName(Solid_6_1, 'Solid_6')
smesh.SetName(Group_45, 'Group_45')
smesh.SetName(Group_42, 'Group_42')
smesh.SetName(Group_43, 'Group_43')
smesh.SetName(Group_6, 'Group_6')
smesh.SetName(Group_7, 'Group_7')
smesh.SetName(Group_4, 'Group_4')
smesh.SetName(Group_5, 'Group_5')
smesh.SetName(Group_2, 'Group_2')
smesh.SetName(Group_3, 'Group_3')
smesh.SetName(Group_0, 'Group_0')
smesh.SetName(Group_1, 'Group_1')
smesh.SetName(Group, 'Group')
smesh.SetName(Group_17, 'Group_17')
smesh.SetName(Group_16, 'Group_16')
smesh.SetName(Group_15, 'Group_15')
smesh.SetName(Group_14, 'Group_14')
smesh.SetName(Group_13, 'Group_13')
smesh.SetName(Group_12, 'Group_12')
smesh.SetName(Group_11, 'Group_11')
smesh.SetName(Group_10, 'Group_10')
smesh.SetName(Group_9, 'Group_9')
smesh.SetName(Group_8, 'Group_8')

if salome.sg.hasDesktop():
  salome.sg.updateObjBrowser()
```

Example SIF code

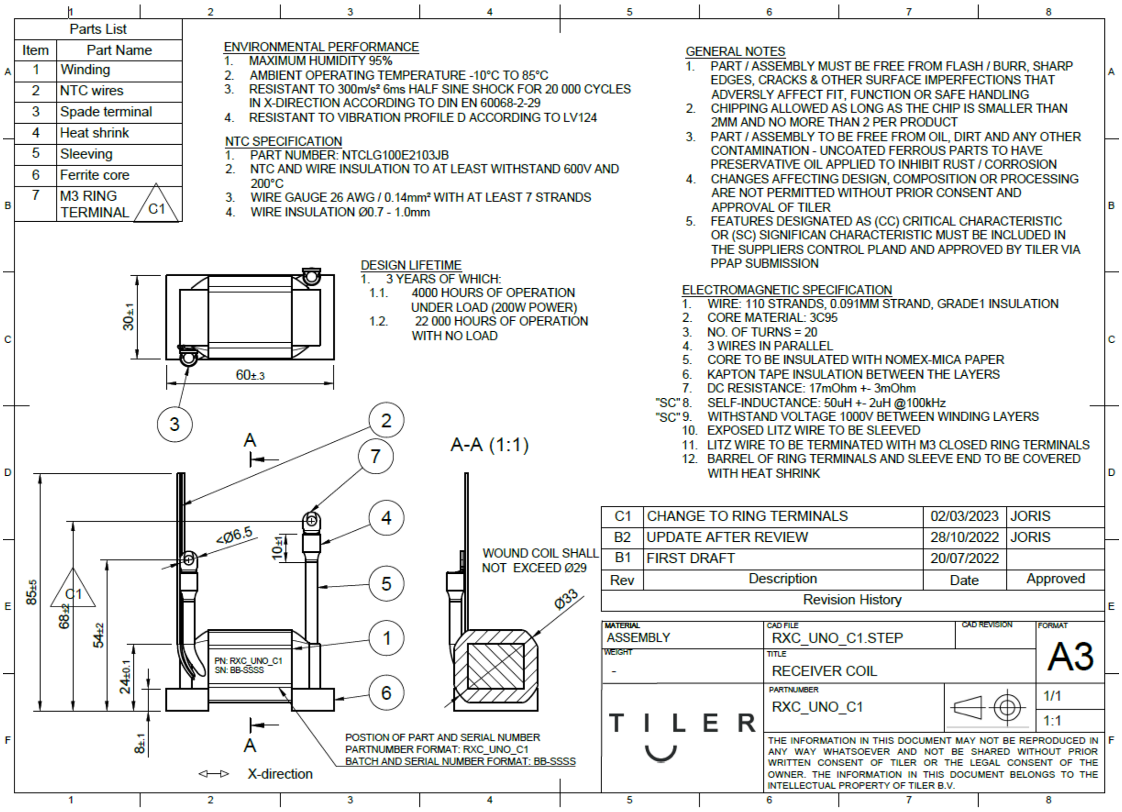
```
Header
CHECK KEYWORDS Warn
Mesh DB "." "."
Include Path ""
Results Directory "res"
End

Simulation
Max Output Level = 4
Coordinate System = Cartesian
Coordinate Scaling = 0.001
Simulation Type = Steady state
Steady State Max Iterations = 1
Output Intervals(1) = 1
Solver Input File = case.sif
Post File = case.vtu
! Nice for visualization of parts.
vtu: vtu part collection = Logical True
vtu: save bulk only = logical True
vtu: Discontinuous Bodies = Logical True

! If you have a crappy paraview version.
! Ascii Output = True

! Double the mesh density.
Mesh Levels = 2
End

Constants
Permittivity of Vacuum = 8.85418781e-12
Permeability of Vacuum = 1.25663706e-6
End

Body 1
Target Bodies(1) = 1
Name = "coil1"
Equation = 1
Material = 1
Body Force = 1
End

Body 2
Target Bodies(1) = 2
Name = "TXC"
Equation = 2
Material = 2
```

```
End

Body 3
Target Bodies(1) = 3
Name = "air"
Equation = 2
Material = 1
End

Body 4
Target Bodies(1) = 4
Name = "RXC"
Equation = 2
Material = 2
End

Body 5
Target Bodies(1) = 5
Name = "coilup"
Equation = 1
Material = 1
Body Force = 1
End

Body 6
Target Bodies(1) = 6
Name = "coil2"
Equation = 1
Material = 1
Body Force = 1
End

Solver 1
Equation = "CoilSolver"
Procedure = "CoilSolver" "CoilSolver"
Linear System Solver = "Iterative"
Linear System Preconditioning = ILU1
Linear System Max Iterations = 1000
Linear System Convergence Tolerance = 1e-10
Linear System Iterative Method = BiCGStab
Linear System Residual Output = 10
Steady State Convergence Tolerance = 1e-06
Normalize Coil Current = Logical True
Nonlinear System Consistent Norm = Logical True
Save Coil Set = Logical True
Save Coil Index = Logical True
Calculate Elemental Fields = Logical True
```

```
! We can make current divergence free also withing each element.
Fix Input Current density = True
End


Solver 2
Equation = MGDynamics
Procedure = "MagnetoDynamics" "WhitneyAVSolver"
Exec Solver = Always
! If we fixed it already no use doing it again the hard way.
Fix Input Current density = False
Use Elemental CoilCurrent = True
Steady State Convergence Tolerance = 1.0e-5
Nonlinear System Max Iterations = 1
Linear System Solver = iterative
Linear System Iterative Method = BiCGStabl ! GCR / Idrs
Linear System GCR Restart = 100
BiCGStabl Polynomial Degree = 6
Idrs Parameter = Integer 6
Linear System Robust = Logical True
Linear System Max Iterations = 1000
Linear System Convergence Tolerance = 1.0e-8
Linear System Preconditioning = none
Linear System Residual Output = 20
Linear System Abort Not Converged = False
! Optionally one can use different scaling. Sometimes gives better results.
! Linear System Row Equilibration = Logical True
Nonlinear System Consistent Norm = Logical True
End


Solver 3
Equation = MgDynPost
Procedure = "MagnetoDynamics" "MagnetoDynamicsCalcFields"
Exec Solver = Before Saving
Calculate Current Density = True
Calculate Magnetic Field Strength = True
Calculate Elemental Fields = True
Calculate Nodal Fields = False
Average Within Materials = True
End


Equation 1
Name = "coil"
```

```
Active Solvers(3) = 1 2 3
End

Equation 2
Name = "others"
Active Solvers(2) = 2 3
End

Material 1
Name = "Air (room temperature)"
Relative Permittivity = 1.00059
Viscosity = 1.983e-5
Heat Capacity = 1005.0
Relative Permeability = 1.00000037
Sound speed = 343.0
Heat expansion Coefficient = 3.43e-3
Heat Conductivity = 0.0257
Density = 1.205
End

Material 2
Name = "3C95"
Sound speed = 5000.0
Density = 4800
Youngs modulus = 193.053e9
Poisson ratio = 0.29
Relative Permeability = 3000
H-B Curve = Variable "dummy"
Real Cubic Monotone
INCLUDE 3C95_bh.dat
End
Electric Conductivity = 10.30e6
Heat Conductivity = 80.2
End

Component 1
Name = String "Coil1"
Master Bodies(1) = Integer 1
Desired Coil Current = Variable Time
Real MATC "-2.5"
Coil Normal(3) = 0 0 1
Coil Closed = Logical True
! Coil Closed = True

Coil type = "stranded"
End
```

```
Component 2
Name = String "Coilup"
Master Bodies(1) = Integer 5
Desired Coil Current = Variable Time
Real MATC "6"
Coil Normal(3) = 1 0 0
Coil Closed = Logical True
! Coil Closed = True
Coil type = "stranded"
End

Component 3
Name = String "Coil2"
Master Bodies(1) = Integer 6
Desired Coil Current = Variable Time
Real MATC "2.5"
Coil Normal(3) = 0 0 1
Coil Closed = Logical True
! Coil Closed = True
Coil type = "stranded"
End

Body Force 1
Name = "source"
! Current Density 1 = Equals "CoilCurrent e 1"
! Current Density 2 = Equals "CoilCurrent e 2"
! Current Density 3 = Equals "CoilCurrent e 3"
End

Boundary Condition 1
Target Boundaries(1) = 63
Name = "BoundaryCondition 1"
P {e} = real 0
End
```


30±1
60±.3
3
2
7
4
5
1
6
A
A
A-A (1:1)
<Ø6.5
10±1
85±5
68±2
C1
54±2
24±0.1
8±1
PN: RXC_UNO_C1
SN: BB-SSSS
WOUND COIL SHALL NOT EXCEED Ø29
Ø33
X-direction
POSTION OF PART AND SERIAL NUMBER
PARTNUMBER FORMAT: RXC_UNO_C1
BATCH AND SERIAL NUMBER FORMAT: BB-SSSS

Parts List	
Item	Part Name
1	Winding
2	NTC wires
3	Spade terminal
4	Heat shrink
5	Sleeving
6	Ferrite core
7	M3 RING TERMINAL C1

ENVIRONMENTAL PERFORMANCE

1. MAXIMUM HUMIDITY 95%
2. AMBIENT OPERATING TEMPERATURE -10°C TO 85°C
3. RESISTANT TO 300m/s² 6ms HALF SINE SHOCK FOR 20 000 CYCLES IN X-DIRECTION ACCORDING TO DIN EN 60068-2-29
4. RESISTANT TO VIBRATION PROFILE D ACCORDING TO LV124

NTC SPECIFICATION

1. PART NUMBER: NTCLG100E2103JB
2. NTC AND WIRE INSULATION TO AT LEAST WITHSTAND 600V AND 200°C
3. WIRE GAUGE 26 AWG / 0.14mm² WITH AT LEAST 7 STRANDS
4. WIRE INSULATION Ø0.7 - 1.0mm

DESIGN LIFETIME

1. 3 YEARS OF WHICH:
  1.1. 4000 HOURS OF OPERATION UNDER LOAD (200W POWER)
  1.2. 22 000 HOURS OF OPERATION WITH NO LOAD

GENERAL NOTES

1. PART / ASSEMBLY MUST BE FREE FROM FLASH / BURR, SHARP EDGES, CRACKS & OTHER SURFACE IMPERFECTIONS THAT ADVERSLY AFFECT FIT, FUNCTION OR SAFE HANDLING
2. CHIPPING ALLOWED AS LONG AS THE CHIP IS SMALLER THAN 2MM AND NO MORE THAN 2 PER PRODUCT
3. PART / ASSEMBLY TO BE FREE FROM OIL, DIRT AND ANY OTHER CONTAMINATION - UNCOATED FERROUS PARTS TO HAVE PRESERVATIVE OIL APPLIED TO INHIBIT RUST / CORROSION
4. CHANGES AFFECTING DESIGN, COMPOSITION OR PROCESSING ARE NOT PERMITTED WITHOUT PRIOR CONSENT AND APPROVAL OF TILER
5. FEATURES DESIGNATED AS (CC) CRITICAL CHARACTERISTIC OR (SC) SIGNIFICAN CHARACTERISTIC MUST BE INCLUDED IN THE SUPPLIERS CONTROL PLAND AND APPROVED BY TILER VIA PPAP SUBMISSION

ELECTROMAGNETIC SPECIFICATION

1. WIRE: 110 STRANDS, 0.091MM STRAND, GRADE1 INSULATION
2. CORE MATERIAL: 3C95
3. NO. OF TURNS = 20
4. 3 WIRES IN PARALLEL
5. CORE TO BE INSULATED WITH NOMEX-MICA PAPER
6. KAPTON TAPE INSULATION BETWEEN THE LAYERS
7. DC RESISTANCE: 17mOhm +- 3mOhm
"SC" 8. SELF-INDUCTANCE: 50uH +- 2uH @100kHz
"SC" 9. WITHSTAND VOLTAGE 1000V BETWEEN WINDING LAYERS
10. EXPOSED LITZ WIRE TO BE SLEEVED
11. LITZ WIRE TO BE TERMINATED WITH M3 CLOSED RING TERMINALS
12. BARREL OF RING TERMINALS AND SLEEVE END TO BE COVERED WITH HEAT SHRINK

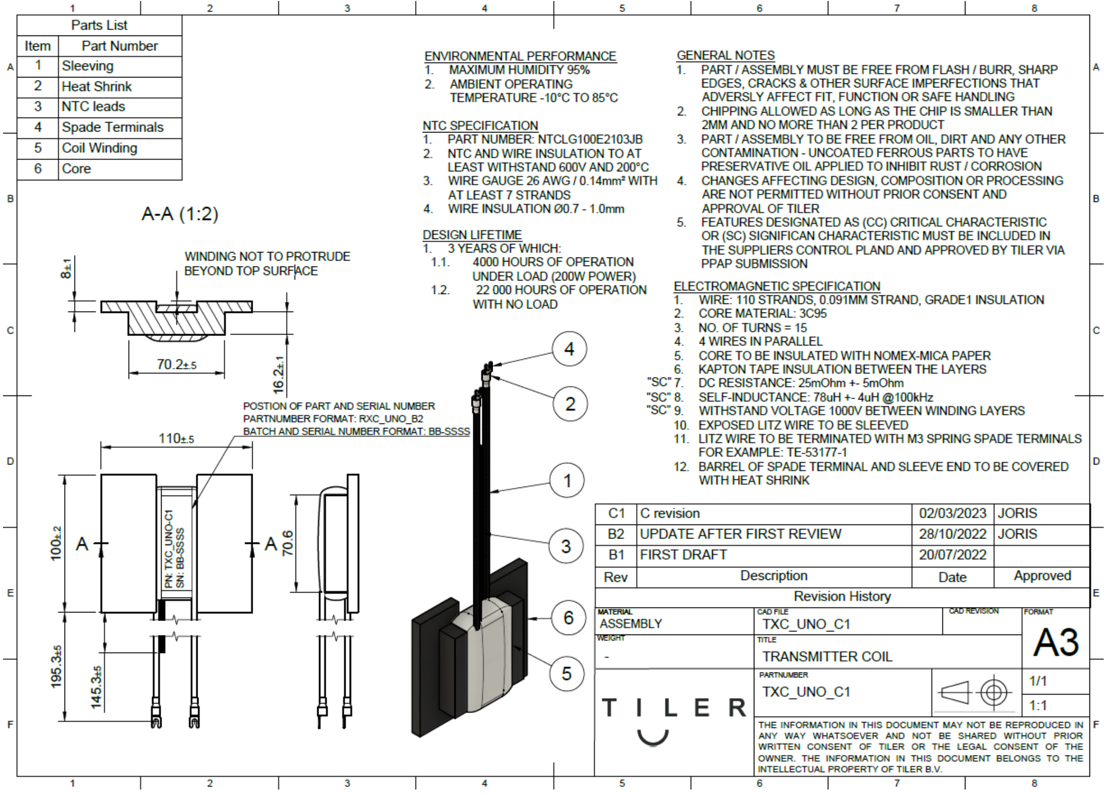
Rev	Description	Date	Approved
C1	CHANGE TO RING TERMINALS	02/03/2023	JORIS
B2	UPDATE AFTER REVIEW	28/10/2022	JORIS
B1	FIRST DRAFT	20/07/2022	

Revision History

MATERIAL	CAD FILE	CAD REVISION	FORMAT
ASSEMBLY	RXC_UNO_C1.STEP		A3
WEIGHT	TITLE		
-	RECEIVER COIL		
TILER	PARTNUMBER RXC_UNO_C1		1/1 1:1

THE INFORMATION IN THIS DOCUMENT MAY NOT BE REPRODUCED IN ANY WAY WHATSOEVER AND NOT BE SHARED WITHOUT PRIOR WRITTEN CONSENT OF TILER OR THE LEGAL CONSENT OF THE OWNER. THE INFORMATION IN THIS DOCUMENT BELONGS TO THE INTELLECTUAL PROPERTY OF TILER B.V.

Figure 53 Original design of RXC


A-A (1:2)
8±1
WINDING NOT TO PROTRUDE BEYOND TOP SURFACE
70.2±.5
16.2±.1
POSTION OF PART AND SERIAL NUMBER
PARTNUMBER FORMAT: RXC_UNO_B2
BATCH AND SERIAL NUMBER FORMAT: BB-SSSS
110±.5
100±.2
A
A
70.6
PN: TXC_UNO-C1
SN: BB-SSSS
195.3±5
145.3±5
4
2
1
3
6
5

Parts List	
Item	Part Number
1	Sleeving
2	Heat Shrink
3	NTC leads
4	Spade Terminals
5	Coil Winding
6	Core

ENVIRONMENTAL PERFORMANCE

1. MAXIMUM HUMIDITY 95%
2. AMBIENT OPERATING TEMPERATURE -10°C TO 85°C

NTC SPECIFICATION

1. PART NUMBER: NTCLG100E2103JB
2. NTC AND WIRE INSULATION TO AT LEAST WITHSTAND 600V AND 200°C
3. WIRE GAUGE 26 AWG / 0.14mm² WITH AT LEAST 7 STRANDS
4. WIRE INSULATION Ø0.7 - 1.0mm

DESIGN LIFETIME

1. 3 YEARS OF WHICH:
  1.1. 4000 HOURS OF OPERATION UNDER LOAD (200W POWER)
  1.2. 22 000 HOURS OF OPERATION WITH NO LOAD

GENERAL NOTES

1. PART / ASSEMBLY MUST BE FREE FROM FLASH / BURR, SHARP EDGES, CRACKS & OTHER SURFACE IMPERFECTIONS THAT ADVERSLY AFFECT FIT, FUNCTION OR SAFE HANDLING
2. CHIPPING ALLOWED AS LONG AS THE CHIP IS SMALLER THAN 2MM AND NO MORE THAN 2 PER PRODUCT
3. PART / ASSEMBLY TO BE FREE FROM OIL, DIRT AND ANY OTHER CONTAMINATION - UNCOATED FERROUS PARTS TO HAVE PRESERVATIVE OIL APPLIED TO INHIBIT RUST / CORROSION
4. CHANGES AFFECTING DESIGN, COMPOSITION OR PROCESSING ARE NOT PERMITTED WITHOUT PRIOR CONSENT AND APPROVAL OF TILER
5. FEATURES DESIGNATED AS (CC) CRITICAL CHARACTERISTIC OR (SC) SIGNIFICAN CHARACTERISTIC MUST BE INCLUDED IN THE SUPPLIERS CONTROL PLAND AND APPROVED BY TILER VIA PPAP SUBMISSION

ELECTROMAGNETIC SPECIFICATION

1. WIRE: 110 STRANDS, 0.091MM STRAND, GRADE1 INSULATION
2. CORE MATERIAL: 3C95
3. NO. OF TURNS = 15
4. 4 WIRES IN PARALLEL
5. CORE TO BE INSULATED WITH NOMEX-MICA PAPER
6. KAPTON TAPE INSULATION BETWEEN THE LAYERS
"SC" 7. DC RESISTANCE: 25mOhm +- 5mOhm
"SC" 8. SELF-INDUCTANCE: 78uH +- 4uH @100kHz
"SC" 9. WITHSTAND VOLTAGE 1000V BETWEEN WINDING LAYERS
10. EXPOSED LITZ WIRE TO BE SLEEVED
11. LITZ WIRE TO BE TERMINATED WITH M3 SPRING SPADE TERMINALS FOR EXAMPLE: TE-53177-1
12. BARREL OF SPADE TERMINAL AND SLEEVE END TO BE COVERED WITH HEAT SHRINK

Rev	Description	Date	Approved
C1	C revision	02/03/2023	JORIS
B2	UPDATE AFTER FIRST REVIEW	28/10/2022	JORIS
B1	FIRST DRAFT	20/07/2022	

Revision History

MATERIAL	CAD FILE	CAD REVISION	FORMAT
ASSEMBLY	TXC_UNO_C1		A3
WEIGHT	TITLE		
-	TRANSMITTER COIL		
TILER	PARTNUMBER TXC_UNO_C1		1/1 1:1

THE INFORMATION IN THIS DOCUMENT MAY NOT BE REPRODUCED IN ANY WAY WHATSOEVER AND NOT BE SHARED WITHOUT PRIOR WRITTEN CONSENT OF TILER OR THE LEGAL CONSENT OF THE OWNER. THE INFORMATION IN THIS DOCUMENT BELONGS TO THE INTELLECTUAL PROPERTY OF TILER B.V.

Figure 54 Original design of TXC