

Andreas Svejgaard Jensen, 334022

Semester Project

DESIGN OPTIMIZATION OF THE PRIMARY COIL FOR INDUCTION HEATING

Professor:

Yves Perriard

Supervisor:

Adrien Thabuis

Integrated Actuators Laboratory

Neuchâtel

11/6 2021

Laboratoire d'actionneurs intégrés (LAI)



SEMESTER PROJECT, SECTION D'ELECTRICITE – SEL, SPRING 2020-2021

Mr Andreas Svejgaard JENSEN

TOPOLOGY OPTIMIZATION OF THE PRIMARY COIL FOR INDUCTION HEATING

Well-known for its application in cooking, induction heating is also widely used in the industry for welding, surface processing, manufacturing and even as the heating system for new types of smart actuators. An important feature of this process is that the heat is generated inside the object itself, instead of by an external heat source via heat conduction. Thus, objects can be heated very rapidly and contactless. Regarding the design of such system, the shape of the primary coil is probably the most challenging one. For a known induced current inside the workpiece to be heated, what is the shape of the primary coil that creates the magnetic field responsible for this induced current? In engineering, it is called an inverse problem. We propose here to use Topology Optimization (TO) to solve such problem. The goal of this project is to adapt an already existing TO framework written in Python to the design of coils for induction heating systems. The student will gain a basic understanding of algorithmic design and finite element analysis, which are highly desired skills for engineers in industry or academia.

Objectives of the work

The objectives of the project are to:

- Understand the inner working of topology optimization
- Adapt the already developed framework to the design of a primary coil for induction heating
- Validate the code with some examples

Report

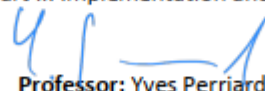
The final report will contain the following points:

- The description of the chosen approach and core components of the library
- Proposed extension to model induction heating problems
- Implementation of proposed method
- Examples of design problems with numerical results

References

- [1] A. J. Mäkinen et al "Magnetic-field modeling with surface currents. Part I. Physical and computational principles of bfieldtools", Journal of Applied Physics, 2020
- [2] R. Zetter et al "Magnetic-field modeling with surface currents. Part II. Implementation and usage of bfieldtools", Journal of Applied Physics, 2020

Supervisor: Adrien THABUIS


Professor: Yves Perriard

During the first week, a weekly meeting will be set with the supervisor(s) and the Professor. Furthermore, an intermediate assessment meeting will be set at the mid-semester.

*Instructions about the report and the presentation can be found on [the LAI website](#). The report and all documents related to the project will be sent by email no later than **June 11th, 2021 at 16:00**. No printed copies are requested. The final presentation will be held during the last week of the semester.*

Neuchâtel, February 22th, 2021.

Figure 0.1: Signed project description

Contents

Contents	i
List of Figures	ii
INTRODUCTION	1
1 THEORY	2
1.1 Induction Heating	2
1.2 Magnetostatic field calculations	3
2 Program	5
2.1 Installation	5
2.1.1 Bfieldtools	5
2.1.2 CoilOptimization	5
2.2 Input	6
2.2.1 Coil Geometries	6
2.2.2 Target Currents	6
2.3 Output	7
2.4 Overview of methods	8
3 RESULTS	9
4 CONCLUSION	12
4.1 Recommendation	12
Time Schedule	13
Bibliography	14

List of Figures

0.1	Signed project description	2
2.1	(A) Screenshot from FreeCAD when generating target current files to input in CoilOptimization. (B) Shows how the exported points for the target current "current1"	7
3.1	(A) Target points, (B) Target fields, (C) Stream function, (D) Stream function when target currents approach each other, (E) Discretization of (D), (F) Volumetric loss density in the workpiece, when unit currents are running in the coils of (E)	9
3.2	(A) Design space, (B) Target current, (C) Target points, (D) Target field, (E) Stream function, (F) Coil design on one of the 2 planes in (A), (G) Coil design on the other of the 2 planes in (A), (H) Comsol verification of volumetric loss density	10
4.1	Actual time schedule of the project	13

INTRODUCTION

This report is the product of the semester project "Design Optimization of Primary Coil Design for Induction Heating". The project was conducted at the Integrated Actuators Laboratory (LAI) at EPFL in the spring semester 2021.

In this project we investigate a new design method for an optimized primary coil design for induction heating.

Induction heating is used in many different applications in the industry to heat up metallic materials. There could be many different reasons for the desired heating, including applications of surface treatment, preheating/postheating workpieces in a welding process or as in our specific case, to heat Shape Memory Alloys (SMA) to drive a phase change.

In section 1 induction heating is covered, followed by a basic introduction to stream functions. In section 2 there is a description on how to use the proposed solution. Section 3 describes the results from 2 different problems that have been tried solved using the proposed solution.

Alongside the report, the python library CoilOptimization is part of the outcome of this project.

Chapter 1

THEORY

1.1 Induction Heating

Heat is an energy form and is related to the kinetic energy of the atom. An increase in heat translates into more intense vibration of the particles of the atom. Heat are following the laws of thermodynamics, and it is propagating due to conduction, convection and radiation.

Heat can be generated from chemical processes, from friction and joule heating. It is the latter which will be important for this project.

Joule heating is heating as a result of resistive losses. When a current is running through any conducting material which are lossy, then parts of the power are converted into heat following the equation:

$$P = \frac{V^2}{I}$$

This assumes a DC current and that the material acts as a perfect resistor.

The resistive loss occur because the charge carriers (the electrons) are randomly scattered by the ions of the matter in which the current is flowing. When the electron are scattered by the ion, kinetic energy is moved from the electron to the ion, resulting in more intense vibrations of the ions.

With higher kinetic energy in the ions, the probability for them to collide with flowing charge carriers increases. This means an increase in temperature of a conductor leads to higher resistance of the material[1].

$$\rho(T) = \rho_0[1 + \alpha(T - T_0)]$$

Where ρ is resistivity, ρ_0 is resistivity at ambient temperature T_0 , α is the temperature coefficient.

Joule heating can be obtained simply by connecting the current carrying wires to the workpiece. The current will flow in the path of least impedance, which is why this method is not capable of heating an arbitrary pattern of the workpiece. An important remark though is, that the least impedance does not necessarily means the shortest path. The impedance are dependent on the frequency of the current, and when the frequency increases it tend to be the inductance that dominates the impedance. Thus when the frequency

goes up, the current tend to follow the path of least inductance, which might not be the shortest path in terms of distance.[2]

Induction, as governed by the Maxwell-Faraday Law, gives the opportunity to generate a current inside a workpiece. The alternating magnetic field gives rise to an electromotive force which in hand accelerates the charge carriers to form an electric current.

From the induction law it is seen that the induced current is proportional to the rate of change of the magnetic field. At the same time eddy currents are induced in such a way, their magnetic field will oppose the original magnetic field. Thus the eddy currents are working to avoid the change in magnetic field.

Due to the skin effect, an alternating current tend to flow closer to the surface of the conductor. The effect becomes more prominent with higher frequency, and the current density falls exponentially from the surface towards the inner core. The skin depth δ is the distance from the surface where the current density has fallen to $\frac{1}{e}$.

1.2 Magnetostatic field calculations

In the articles Magnetic-field modeling with surface currents Part I and II [3, 4], R. Zetter et al. presents a novel methodology to model magnetic field using surface currents.

In their method, current densities are represented by stream functions ψ .

$$\vec{j} = \nabla\psi(\vec{r}') \times \hat{n}(\vec{r}')$$

Where \vec{r}' is the position vector and \hat{n} is the unit surface normal vector.

The stream function is treated as piece wise linear, and so it the value of the stream function is equal to a sum of the contributions from all of the vertices

$$\psi(\vec{r}') = \sum_i s_i h_i(\vec{r}')$$

Where s_i is the weight and h_i is the basis function.

The stream function is defined on all vertices in the mesh, and is then interpolated in between the vertices.

Since there is a scalar weight s_i for all vertices, they can be formed into a column vector \mathbf{s} . This vectorized format allows the usage of linear algebra, which results in fast calculations. The basis function is by default the hat function, but when the surface current density is needed, it is simply a matter of applying another basis function in the summation. The surface current density is equal to the rotated gradient of the stream function, and so the calculation is

$$\vec{j} = \sum_i s_i \nabla h_i(\vec{r}') \times \hat{n}$$

In the R. Zetter et al. paper they argue using potential theory, that within magnetostatics, any magnetic field inside a volume, can be expressed by a surface current on the boundary of the volume. The theory also holds for open surfaces, though the stream function must be constant on the boundary of the wholes and on the boundary of the edges of the surface.

The methodology assumes we are in the magnetostatic region, and from this assumption follows, that any high frequency phenomena is not modelled using Bfieldtools.

Chapter 2

Program

2.1 Installation

CoilOptimization depends on a range of standard scientific python libraries, and it also depends on Bfieldtools. Due to the requirements of Bfieldtools the installation is quite specific, therefore it is recommended to follow the installation guide given below.

2.1.1 Bfieldtools

Bfieldtools is distributed using pip, but in order to have a successful installation it was discovered that the following installation procedure must be followed.

STEP 1: Setup a new python environment using miniconda with python version 3.7

STEP 2: conda install numpy, scipy, matplotlib, sympy, vtk, mayavi, geos

STEP 3: conda install spyder

STEP 4: At this point verify Spyder can open without issues

STEP 5: pip install PyQt5==5.12

STEP 6: pip install --upgrade PyQtWebEngine==5.12

STEP 7: pip install bfieldtools

STEP 8: pip install -U numpy

STEP 9: conda install -c mosek mosek

STEP 10: Get MOSEK license: <https://www.mosek.com/products/academic-licenses/>

When the MOSEK license has been placed correctly, bfieldtools should be fully working

2.1.2 CoilOptimization

The library CoilOptimization requires Bfieldtools to be installed. When Bfieldtools has been installed using the procedure described in section 2.1.1, the installation of CoilOptimization library is simply 2 steps:

STEP 1: Copy the folder CoilOptimization into any path COIL_OPTIMIZATION_PATH

STEP 2: Unless COIL_OPTIMIZATION_PATH is already in python's path, then update the variable in *main_local.py* and *main_server.py*.

2.2 Input

The procedure to create the input files will be based on the program FreeCAD, but the concepts should be applicable to any CAD program. FreeCAD was chosen because it is open source, runs on Windows, MAC OS and Linux, and because it is a simple process to perform the desired tasks.

It is recommended to use version 0.19 or higher, since a bug in the previous version prevents import of picture, which is needed in the section below.

2.2.1 Coil Geometries

The coil geometries are where the primary coils can be designed. For this project we have taken starting point in a sandwiched design space, with 2 identical planes placed parallel to each other. The coil geometries can basically take any desired shapes, which is relevant for designing coil for specific workpieces or environments.

The coil geometries must be transformed into a triangular mesh.

- STEP 1:** Select Arch from the workbench dropdown
- STEP 2:** Choose the polyline tool (Draft -> Creation -> Polyline)
- STEP 3:** Draw a closed loop
- STEP 4:** Select Surface from the workbench dropdown
- STEP 5:** From Surface dropdown select Fill Boundary Curves
- STEP 6:** Click Add Edge, and click all edges of the polyline
- STEP 7:** Click OK
- STEP 8:** From the workbench dropdown select Mesh Design
- STEP 9:** Mark the surface in the left pane
- STEP 10:** Click Meshes -> Create Mesh from Shape
- STEP 11:** Using Mefisto put max edge size and click OK
- STEP 12:** If needed, right click the mesh in the left pane and click Transform in order to transform into another plane
- STEP 13:** Mark mesh in left pane
- STEP 14:** Click Meshes -> Export Mesh (.obj)

2.2.2 Target Currents

The target currents are where it is desired to have a current inside the workpiece, thus also where it is desired to have heating. Target currents are inputted to CoilOptimization in points files, here .asc will be demonstrated. The points forms a loop, which is one target current. The direction of the points defines the direction of the current (which can be reversed by Bfieldtools, if needed). Each target current carries a unit current, but this can be modified by filling the multiplier argument when the current is initialized.

Several different target currents can be initialized into CoilOptimization, as long as there are enough distance in between these target currents. Issues has been noticed when target currents are initialized too close to each other.

In order to generate a file which can be inputted to CoilOptimization, the following procedure can be followed:

- STEP 1:** From the workbench dropdown select Image
- STEP 2:** Click the icon "Create a planar image in the 3D space"
- STEP 3:** Select the image of the workpiece
- STEP 4:** Select desired plane and Offset = -0,10 mm and click OK
- STEP 5:** Select Arch from the workbench dropdown
- STEP 6:** Choose the polyline tool (Draft -> Creation -> Polyline)
- STEP 7:** Draw a closed loop
- STEP 8:** Choose Points from the workbench dropdown
- STEP 9:** Mark the polyline in the left pane
- STEP 10:** Click Points -> Convert to points...
- STEP 11:** Input max distance and click OK
- STEP 12:** If needed, right click the points in the left pane and click Transform in order to transform into another plane
- STEP 13:** Mark points in left pane
- STEP 14:** Click Points -> Export points (.asc)

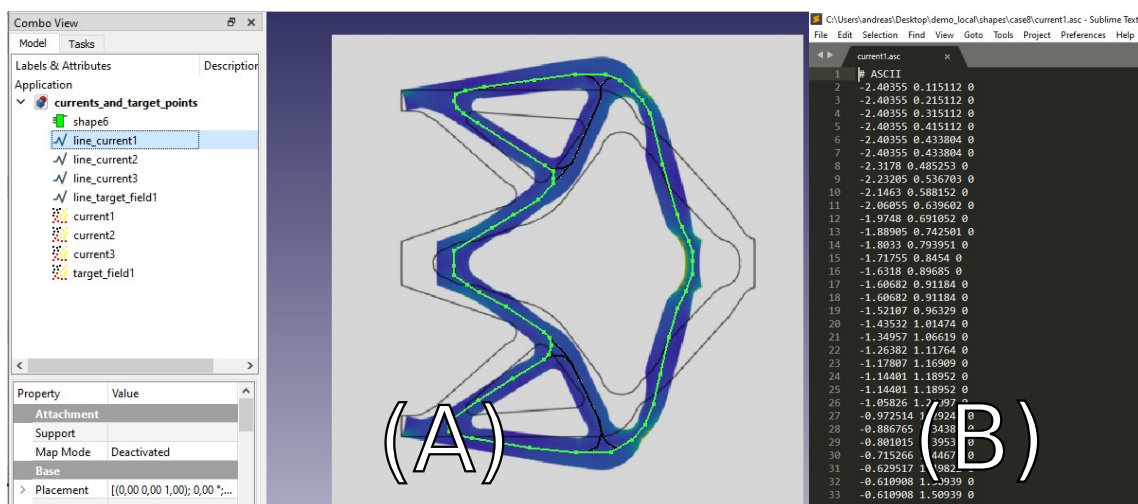


Figure 2.1: (A) Screenshot from FreeCAD when generating target current files to input in CoilOptimization. (B) Shows how the exported points for the target current "current1"

In figure 2.1 it is seen that multiple target currents are created in the same main FreeCAD file. Though each of the individual loop currents (current1, current2, current3) are exported into 3 individual files.

2.3 Output

The output from CoilOptimization are the primary coils. They are visualized, and they are exported to either numpy format or Matlab format.

Also the stream function is outputted into a .dat file, which can also be loaded back into

CoilOptimization again. This provides the ability to conduct the computation in one computer, then transfer the stream function .dat file to another computer and only do visualization on the second computer.

2.4 Overview of methods

FILE	FUNCTION	PARAMETERS
design_space.py	setup_design_space()	file: filename of the coil mesh file twin: True for a sandwiched design space else False distance: distance in mm between the coil mesh and the origin plane: ['XY', 'XZ' or 'ZY'] input what plane the coil mesh and the target points lies in
target_current.py	add_target_current()	file: filename of a target current file multiplier: Integer, to indicate number of unit currents to run in this target current
	add_target_points()	file: filename of the target points file side: ['BOTH', 'LEFT', 'RIGHT'] indicates if the points should be generated on the inside, the outside or on both sides of the current loop (as seen when going in the direction of the current) number: Integer [1-5] defines how many lines of points are generated distance: indicates the distance between the lines of points
	add_target_spec()	
	add_target_spec_zero()	file: filename of points file
solving.py	solve()	
plotting.py	plot_target_current()	
	plot_target_points()	
	plot_target_field()	
	plot_design_space()	
	plot_stream_function()	
	plot_coils_2d()	side: ['plus', 'minus'] determines which of the two coil meshes are plotted for a sandwiched setup
	plot_coils_3d()	
	plot_magnetic_field()	
	create_points_plane()	plane: ['XY', 'XZ', 'ZY'], determines what plane the points should lie in limit_negative: determines the size of the plane limit_positive: determines the size of the plane points: determines how many points in each dimension position: positional offset on the axis normal to the plane
output.py	export_to_matlab()	file: filename of the exported file
	export_target_current_to_matlab()	file: filename of the exported file
	export_stream_function()	file: filename of the exported file
	import_stream_function()	file: filename of file to import N_contours: Integer, determining how fine the stream function is discretized

Table 2.1: Table of methods in the CoilOptimization module. The parameters for each method are listed with a description in the 3rd column.

All methods of the CoilOptimization module are listed in table 2.1. For each method the parameters are described. For all the methods without any parameters, it should be noted, that they are working on object variables. Instead of passing data as arguments, the methods are storing the data on the object, which are then loaded by the next method.

Chapter 3

RESULTS

Many simulations has been conducted in order to come to the final implementation of CoilOptimization. Here is first present a problematic simulation, followed by a more successful one.

In this simulation it is desired to investigate the behavior, when two well known Helmholtz coils are forced closer together side by side.

The results can be seen in figure 3.1. (A), (B) and (C) looks as expected, similar to simulating 2 Helmholtz coils individually.

Now because it is a situation which occur in many practical application, it is of interest to see the result when these 2 target currents comes closer. In (D) it is seen how it immediately gives rise to a very abrupt stream function. This obviously translates into a very dense collection of coils in (E), and finally it can be seen from the volumetric loss density in (F) where the power dissipated in the central region is many times higher compared to in the loops.

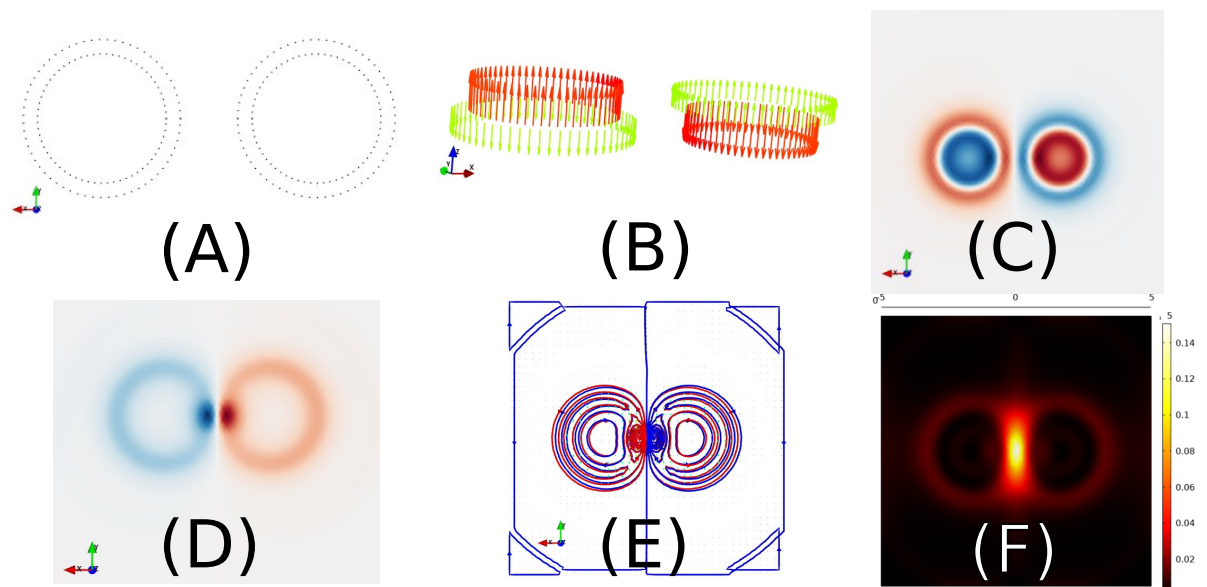


Figure 3.1: (A) Target points, (B) Target fields, (C) Stream function, (D) Stream function when target currents approach each other, (E) Discretization of (D), (F) Volumetric loss density in the workpiece, when unit currents are running in the coils of (E)

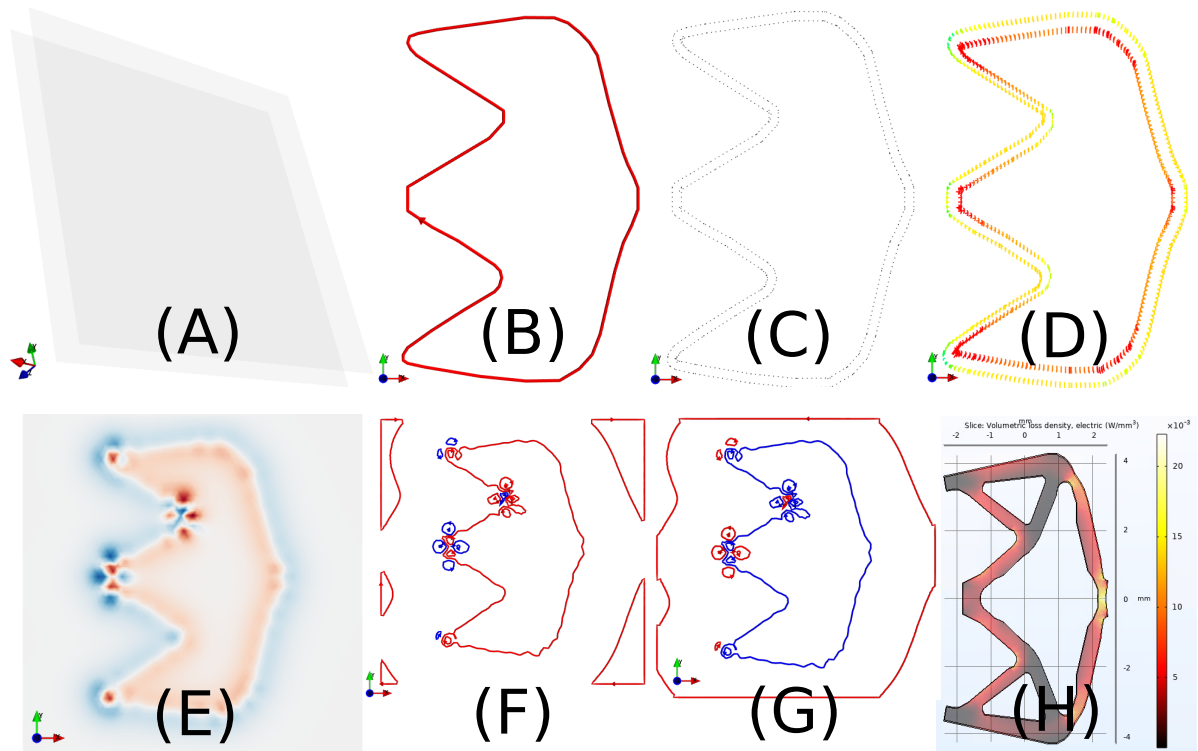


Figure 3.2: (A) Design space, (B) Target current, (C) Target points, (D) Target field, (E) Stream function, (F) Coil design on one of the 2 planes in (A), (G) Coil design on the other of the 2 planes in (A), (H) Comsol verification of volumetric loss density

In figure 3.2 all plots from CoilOptimization is shown. (A) shows 2 planes 10 by 10 mm placed parallel in the XY plane. They are separated by a variable distance, here 1 mm. The 2 planes are meshed in a triangular mesh with a mesh size that gives 7788 faces per plane. The target region is in between these two planes, meaning that picture (B)-(D) are placed in the space in between the 2 planes.

(B) shows the desired current in the workpiece.

(C) shows the target points. These are generated separately from the initialization of the target current, but in a very similar manner. For the target points, CoilOptimization is looping through a given list of points, and for each point it generates target points on the inner side and/or on the outer side depending of the given arguments. Here is chosen a single point on both sides, and the target current point-file is reused as input. (D) shows the the calculated magnetic field at the target points. The more red color indicates higher intensity of the magnetic field and vice versa for green color. The direction of the field is determined by the direction of the current in (B), here the inner region is pointing into the page and the outer region is pointing out of the page.

The magnitude is normalized to the maximum value, and afterwards vector components less than a certain threshold (here 0.0001) has been neglected. Also all vector components has been limited to the mean value in order to issues of too abrupt stream functions.

(E) shows the stream function with red being positive values and blue being negative values. It is seen how there are higher intensities where a big change of the magnetic field is needed over a short distance.

(F) and (G) shows the proposed coil windings on the coil meshes. These are the isolines of

the stream function, and they have been generated by discretizing the stream function. Each loop carries a unit current. (H) Shows a simulation from Comsol, in order to verify the power dissipation in the workpiece. In this plot more white indicates a higher volumetric loss density given in $\frac{W}{mm^2}$. From the Comsol simulation it can be seen that there is power dissipated in the desired region. The simulation was made with nitinol as material of the inverter, and using the real geometry of the inverter.

Chapter 4

CONCLUSION

In this semester project we have seen how Bfieldtools can be used to solve magnetostatic problems from different area of science compared to what it was originally designed for. A new python library, CoilOptimization, is proposed in order to optimize the design of primary coils for induction heating.

The approach has been verified by simulating the resulting coil design in Comsol. From the simulations it was clear that power is dissipated in the desired heating pattern. Based on these simulations we can conclude that the approach and the implementation succeed for some problems. At the same time it has been shown how the proposed approach fails for more complicated problems.

4.1 Recommendation

1. It is recommended to setup the solver in an iterative loop, which decrease the error tolerance every time the solver converge. This way the program would be able to find better solutions compared to the implementation with static error tolerances.
2. When good results has been verified with Comsol simulations, the resulting coil design should be turned into a practical PCB layout and tested in laboratory.
3. Implement a custom optimization function, which maximize the power dissipation in the workpiece. This would change the problem into a direct problem, and would cause the program to design the optimum coil design for the desired heating pattern. The computational power would be spend on optimizing the desired parameter, instead of parameters which are actually don't cares.

Time Schedule

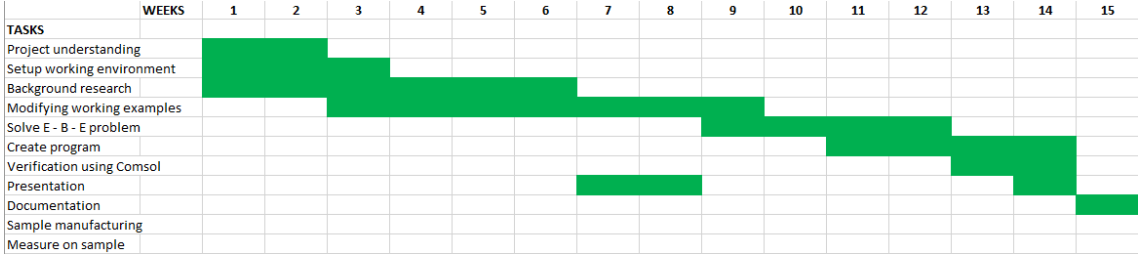


Figure 4.1: Actual time schedule of the project

Bibliography

- [1] Raymond L. Cook Valery Rudnev, Don Loveless. *Handbook of Induction Heating, Second Edition*. Manufacturing Engineering and Materials Processing. CRC Press, 2 edition, 2017.
- [2] Henry W. Ott. *Electromagnetic Compatibility Engineering*. John Wiley & Sons, 1 edition, 2009.
- [3] Antti J. Mäkinen, Rasmus Zetter, Joonas Iivanainen, Koos C. J. Zevenhoven, Lauri Parkkonen, and Risto J. Ilmoniemi. Magnetic-field modeling with surface currents. part i. physical and computational principles of bfieldtools. *Journal of Applied Physics*, 128(6):063906, 2020.
- [4] Rasmus Zetter, Antti J. Mäkinen, Joonas Iivanainen, Koos C. J. Zevenhoven, Risto J. Ilmoniemi, and Lauri Parkkonen. Magnetic field modeling with surface currents. part ii. implementation and usage of bfieldtools. *Journal of Applied Physics*, 128(6):063905, 2020.

LAI

EPFL

Rue de la Maladière 71b

2000 Neuchâtel

Switzerland

<https://www.epfl.ch/about/campus/neuchatel-en/>

Tel: (+41) 216954211