

University of Twente

EEMCS/Electrical Engineering
Control Engineering



A finite difference time domain model of an eddy current based measurement system

Oskar Janssen

MSc report

Supervisors:

prof.dr.ir. P.P.L. Regtien
MSc. W. van Hoorn
dr.ir. R.M. de Ridder

August 2009

Report nr. 014CE2009
Control Engineering
EE-Math-CS
University of Twente
P.O. Box 217
7500 AE Enschede
The Netherlands



Summary

Industrial companies often use large steel tanks to store all kind of liquids. Due to stress and weather influences the tanks protecting coating can get damaged and corrosion can occur. Corrosion is an auto-catalytic chemical process that needs to be detected in an early stage to prevent leakage. A scanner has been developed to detect these corrosion pits by means of eddy currents. The scanner uses 2 active coils in bridge configuration to generate time-varying electromagnetic fields and measure responses. Magnets are used to magnetize the steel plate under investigation. Magnetization mitigates magnetic inhomogeneities and increases scanning depth. Based on the difference in complex impedance of the coils conclusions can be drawn about the condition of the steel.

Although eddy current testing is a powerful method it appears to be hard to interpret signals of non-artificial defects. Therefore it is needed to gain more knowledge about eddy current responses. In this project a software model has been implemented using the *finite difference time domain* technique. This technique is especially developed for electromagnetic problems and calculated propagating waves in the time domain.

Simulation results of artificial defects in unmagnetized steel show equivalent responses compared to measurements. A magnetization process has been implemented, but it has instability issues when using non-linearities. The model can be further improved by implementing properties of corrosion and optimizing calculations.

Preface

Exactly 8 years ago I started studying Electrical Engineering at the University of Twente. After a year and a few months I decided to continue my study at the 'Saxion Hogeschool Enschede', because the results at university level were disappointing. In 2006 I graduated and decided to give it another try at the UT. This time it was more successful given the fact that I am now writing this report.

The subject of my thesis was quite different from the majority of courses that I followed. Although I could not apply much knowledge from these courses I found it useful to learn more about the measurement side of Control Engineering. The project came about in cooperation with Applus RTD. Hopefully the results of this project are useful to them in the future.

I would like to thank Wim van Hoorn and Chris Groenhout from Applus for their support during the project. I thank my supervisor Paul Regtien for many hours of discussion on the subject. Furthermore thanks to Rene de Ridder for being part of the committee. Thanks also to my family for their support and my girlfriend Ingrid for her everlasting patience during the lonely weekends.

I dedicate this report to my father Harry who died in 2003. He always encouraged me to finish my study at the university. I am sure he is proud that I made it.

Oskar Janssen
Hengelo, August 2009

Contents

1	Introduction	1
1.1	Problem definition	2
1.2	Assignment goals	2
1.3	Report outline	2
2	Eddy current testing	3
2.1	Parameters influencing the measurement	4
2.2	Eddy current measurement system	6
3	Electromagnetic modeling	9
3.1	Overview of methods	9
3.2	The finite difference time domain method	10
3.2.1	The perfectly matched layer	12
3.2.2	Programming language	13
4	Modeling of the eddy current measurement system	15
4.1	Model of an air-cored coil	15
4.1.1	Coil calculations	17
4.1.2	Simulation results	17
4.2	Model of the complete system	20
4.2.1	Unmagnetized scanning	20
4.2.2	Magnetized scanning	23
5	Measurement results	27
5.1	Top holes	28
5.1.1	Tests without DC field	28
5.1.2	Tests with DC field	29
5.2	Bottom holes	30
5.3	Conclusions	32
6	Conclusions and recommendations	33
6.1	Conclusions	33
6.2	Recommendations	33
A	Software manual	35
A.1	Overview	35
A.2	View Model	35
A.3	Edit crack	38
A.4	Magnetize plate	39
A.5	Simulate	40
A.6	Data structure	42
B	Software structure	43
B.1	GUI classes	43
B.2	Simulator classes	45
B.3	Model classes	46

1 Introduction

Industrial companies often use large steel tanks to store all kind of liquids (oil, chemicals, etc.). These tanks are subject to varying weather conditions and metal fatigue due to loading and unloading. As a result, cracks can originate in the protecting coating of the tank. The bare steel underneath these cracks is then exposed to the liquid inside the tank or moisture from outside. This is the start of a corrosion process. Corrosion often starts as a pit and keeps on growing due to an auto-catalytic chemical process. If the corrosion process is not stopped in time, a 'through hole' can be formed and the tanks content can leak away, causing environmental and economical damage. It is therefore important to detect corrosion in an early stage.

Detection of corrosion can be done with a non-destructive testing (NDT) method which can evaluate the properties of material without causing damage to it. For steel tanks it is possible to use electromagnetic fields (EM fields) for crack detection. One of the methods using EM fields is magnetic flux leakage (MFL), which uses a powerful magnet to magnetize the steel piece under investigation. At places where the steel is damaged, magnetic field leaks from the steel. This leakage can be measured with a separate sensor. Another method, called eddy current testing (ECT), uses time-varying electromagnetic fields to generate eddy currents in steel. When corrosion affected the steel plate, the eddy current flow will be disturbed which can be measured as a change in complex impedance of the probe.

ECT has several advantages over MFL. Using ECT measurements can be done with a single probe (coil) that both generates the EM fields and detects flaws, while MFL needs a separate sensor for detection. Another advantage is depth information that can be obtained from ECT due to its time-dependent nature. This makes it possible to detect defects at different depths in the material under investigation. ECT will be discussed in more detail in chapter 2.

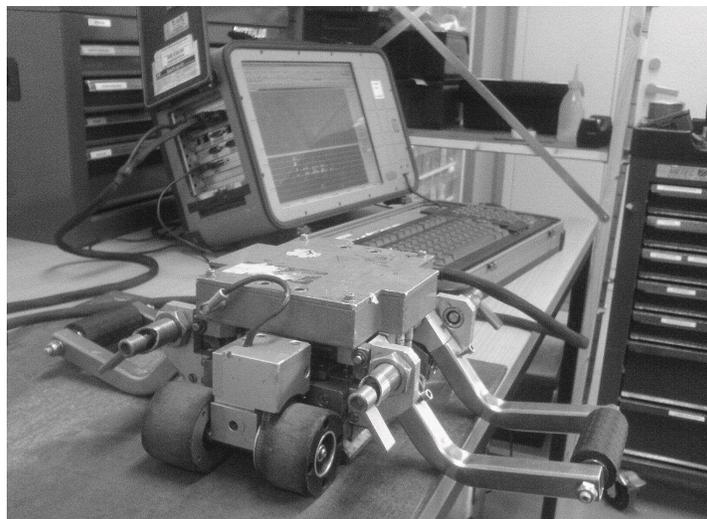


Figure 1: Eddy current measurement system

1.1 Problem definition

Although eddy current testing is a powerful method to detect defects, it is also hard to interpret its measurement signals. ECT is very sensitive to inhomogeneities in the material that is being inspected. Environmental conditions like temperature and humidity have also non-trivial effects on the material properties. As a result, the repeatability of measurements is poor. Mechanical adjustments have been made in order to reduce the influence of these disturbances, but more knowledge about eddy current responses is needed to improve the system.

1.2 Assignment goals

The goal of this assignment is to obtain more information about the responses measured with an existing eddy current measurement system in order to improve the measurement system itself and post-measurement data processing. Emphasis in this project is on the development of a model of the current measurement system. Modeling gives the freedom to simulate signals without unknown or unwanted influences. This way the 'ideal' eddy current response can be investigated. Specific disturbances can later be added to the model whenever needed.

1.3 Report outline

Chapter 2 explains the common background about eddy current testing and how it is used in the current measurement system. Chapter 3 compares several computational techniques that can be used to create the model and highlights the technique that has been used (finite difference time domain). Chapter 4 discusses the implementation of the measurement system using FDTD. Chapter 5 compares model results to practical measurements. In chapter 6 conclusions and recommendations will be discussed.

2 Eddy current testing

Eddy current testing is a technique to detect defects (cracks, corrosion, etc) in conducting materials. By inducing alternating electromagnetic fields near the material under investigation, eddy currents start to flow because of the conductivity. This can be seen from formula 1 (although it is not specifically applicable to eddy currents, because of its frequency independency). Here \mathbf{J} is the current density vector approximated by a multiplication of the electrical conductivity (σ) and the electric field strength vector (\mathbf{E}) [1].

$$\mathbf{J} = \sigma \cdot \mathbf{E} \quad (1)$$

This formula also shows that the eddy currents increase with either stronger EM fields or better conductivity. Eddy currents flow in loops, so their presence is maximized in flawless, large volumes of conducting material. Cracks and corrosion disturb possible eddy current paths, resulting in weaker currents.

A common way to generate the magnetic fields is brought about by a coil carrying an AC current. The same coil can also serve as a sensor (active probe), since the voltage over the coil depends on the time-derivative of the magnetic flux passing through it (which in turn is affected by eddy currents). Figure 2 shows this measurement principle [2]. When eddy currents flow, energy from the applied EM field is dissipated. This dissipation shows up as resistance in the probe signal. Eddy currents also induce their own EM fields. These fields oppose the original applied field (Lenz's law). The weaker net field affects the inductive reactance of the probe.

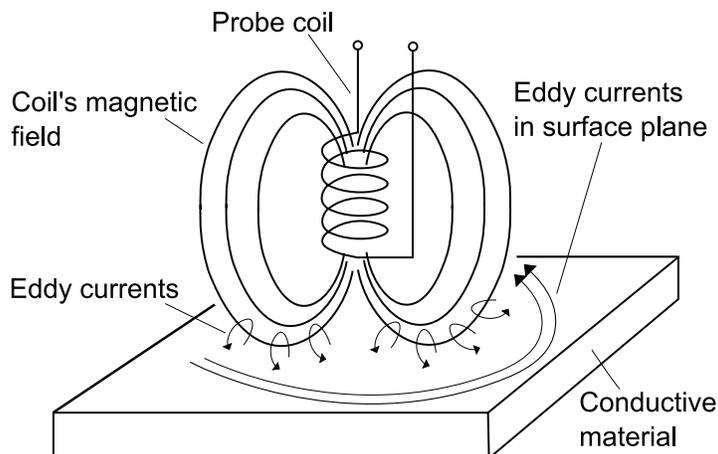


Figure 2: Measurement principle for flaw detection using eddy currents

2.1 Parameters influencing the measurement

Results of eddy current measurements are usually represented by means of a complex impedance plane. In such a plane the horizontal axis represents the resistance (real part) of the probe impedance and the vertical axis represents the inductive reactance (imaginary part). The result of a measurement depends on several parameters.

Most important are the magnetic properties of the (electrical conductive) material under investigation. In case of a non-magnetic material (e.g. aluminum) the measured inductive reactance will *decrease* when the probe approaches the material. This is caused by the opposing fields as a result of eddy currents. However, in case of ferromagnetic materials (e.g. steel) the inductive reactance will *increase* because of the high relative permeability (μ_r) of the material. The permeability concentrates the applied magnetic field and overshadows the field caused by eddy currents.

Another important parameter is the distance between the probe and the material. This is called 'lift-off'. It is important to keep the lift-off as small as possible to get a strong measurement signal. The amplitude of the response signal will quickly reduce with a growing lift-off. In practice there will always be a certain lift-off in order to scan a (rough) surface without damaging the sensors.

Figure 3 shows the effects described above for both non-magnetic (aluminum) and magnetic (steel) materials. It shows that the lift-off curves for both materials are different, but responses to cracks are similar [3].

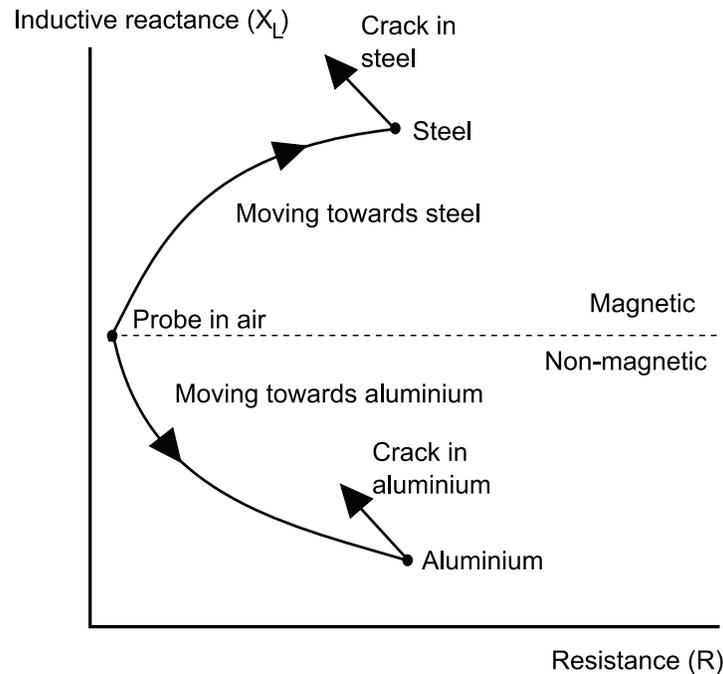


Figure 3: Eddy current response representations in a complex impedance plane

The frequency (f) that is used to excite the probe coil, along with the relative permeability ($\mu_r = \frac{\mu}{\mu_0}$) and the conductivity σ of the material, determine the *standard depth of penetration* (δ). This can be seen from formula 2.

$$\delta \approx \frac{1}{\sqrt{\pi \cdot f \cdot \mu \cdot \sigma}} \quad (2)$$

The standard depth of penetration is a measure for the depth at which the eddy current density has decreased to $\frac{1}{e}$ (about 37 %) of the surface density (as shown in figure 4). Due to this phenomenon eddy current measurements contain information about the depth of a crack. At relative 'high' frequencies only surface defects will be detected while at relative 'low' frequencies also subsurface defects influence the response.

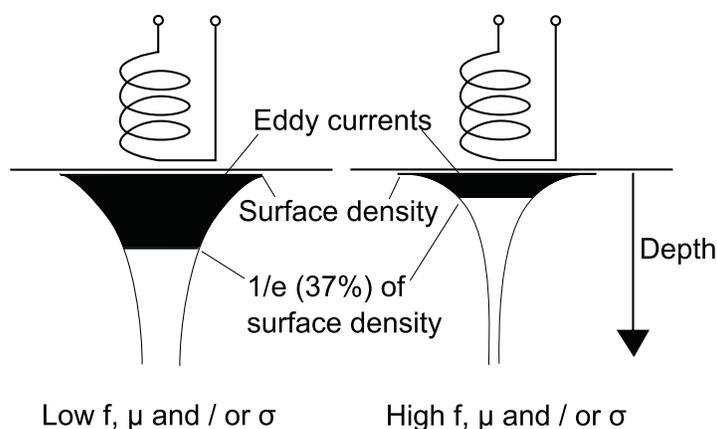


Figure 4: Standard depth of penetration of eddy currents

The design of the probe has an influence on measurements. The sensitivity of a coil increases with increasing self-inductance. The self-inductance mainly depends on the number of turns, coil length, the area enclosed by a turn and the core material. Furthermore details like shape (round, square) and wire thickness have a small effects. Formulas and calculators are widely available to calculate the self-inductance of a specified coil (e.g. [4] and [5]). To improve the measurement, probes can also be shielded to avoid reflections from unwanted objects. By placing a ferrite ring around the probe coil magnetic fields can be concentrated close to the probe (magnetical shielding). Electrical shielding can be done with a ring of highly conductive material around the probe. However, shielding is not an issue in this project.

2.2 Eddy current measurement system

In the eddy current measurement system to be modeled two active coils are used to perform a *difference measurement*. The coils have opposite turns and are placed in a measurement bridge as shown in figure 5. The big advantage of a difference measurement is the elimination of common influences (like temperature and slightly varying material properties). This improves the repeatability of the measurements. A drawback of this approach is the loss of absolute measurement values. Measurements can only be compared relative to other measurements.

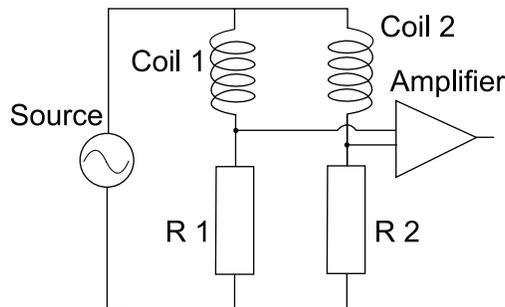


Figure 5: Bridge configuration to perform difference measurement

In contradiction to the 'conventional' configuration described in the previous sections, the coils main axes are placed in parallel (rather than perpendicular) with respect to the material to be tested (figure 6). The advantage of this positioning is the smaller 'footprint' of the coils. The coils only scan a small part of the surface and need much less space in comparison with the conventional set-up. Figure 6 shows both situations.

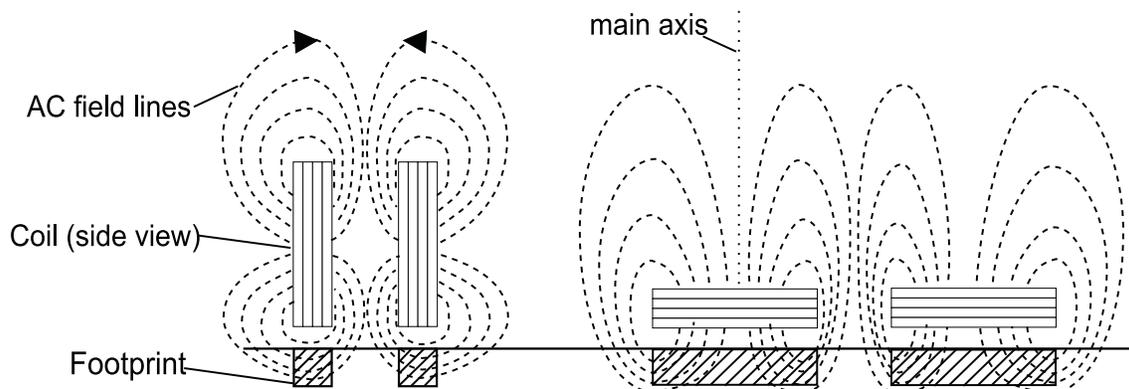


Figure 6: Implemented coil configuration versus conventional configuration

Since the purpose of the measurement system is to test steel plates, the magnetic properties of steel have been used to improve the repeatability of the measurements. This is done by placing a magnet over the probe coils such that a DC magnetic field is generated in the steel plate below the coils. By applying this DC field, the working point on the magnetization curve is shifted from the origin to a higher point (see figure 7). As a result, the influence of local inhomogeneities

of magnetic nature is mitigated (that is, the magnetic history is (partly) erased). However, if the magnet is too strong (the working point approaches saturation level), the superimposed AC fields from the probe are negligible because the field lines are mainly determined by the DC magnet. Hysteresis can make the measurements less repeatable because steel does not completely demagnetize after being exposed to a magnetic field. This results in a slightly varying relative permeability. The same issue occurs with different types of steel. Although magnetization curves of ferromagnetic materials all have globally the same shape (see figure 7), absolute values vary. Calibration before a series of measurements can mitigate these phenomena.

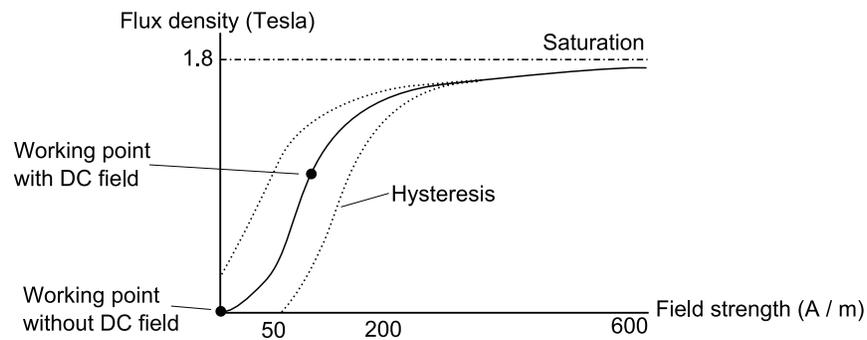


Figure 7: Typical magnetization curve of iron

The applied DC field not only mitigates little disturbances, it also enables the scanner to detect subsurface defects. This is illustrated in figure 8. The hole at the bottom side of the plate enforces the DC field lines to bend around it. As a result, the flux density increases locally. Due to this change, the response of a coil close to the hole will be affected.

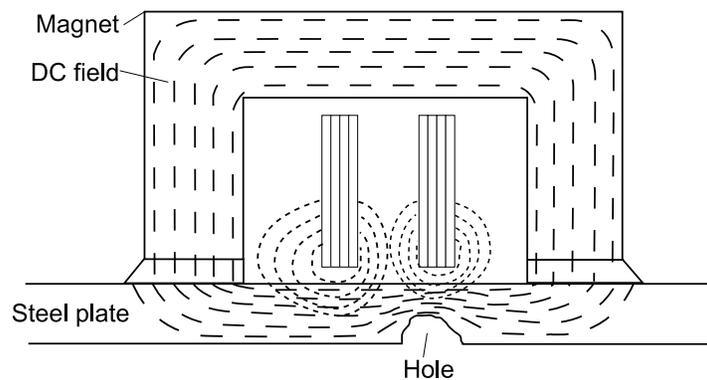


Figure 8: Effect of the applied DC magnetic field

3 Electromagnetic modeling

Most real-world problems dealing with electromagnetics are not analytically solvable due to geometrical irregularities or other object properties. In these cases computational numerical techniques can be used to solve equations across the problem domain in order to find the electrical and magnetical field distributions. Maxwells equations (equations 3-6) form the base of all computational methods. These four equations, together with the Lorentz force law ($\mathbf{F} = q \cdot (\mathbf{E} + \mathbf{v} \times \mathbf{B})$), are the complete set of laws of classical electromagnetism.

$$\text{Gauss' law for electricity: } \nabla \cdot \mathbf{D} = \rho_f \quad (3)$$

$$\text{Gauss' law for magnetism: } \nabla \cdot \mathbf{B} = 0 \quad (4)$$

$$\text{Faraday's law: } \nabla \times \mathbf{E} = -\frac{\partial \mathbf{B}}{\partial t} \quad (5)$$

$$\text{Ampere-Maxwell law: } \nabla \times \mathbf{H} = \mathbf{J}_f + \frac{\partial \mathbf{D}}{\partial t} \quad (6)$$

Equations 3-6 are formulations in terms of free charge and current. Those are most relevant in ECT. Here divergence is denoted by $\nabla \cdot$ and curl by $\nabla \times$. \mathbf{D} is the electric flux density, \mathbf{B} is the magnetic flux density, \mathbf{E} is the electric field and \mathbf{H} is the magnetical field. \mathbf{J}_f the free current density and ρ_f the free charge density.

3.1 Overview of methods

The choice of a computational method depends on the kind of problem to be solved. Most methods create a mesh to divide the problem space into small elements. These elements are then updated separately. Several popular methods will be discussed in order to motivate the chosen technique for this project.

Method of moments (MoM) (or boundary element method (BEM)) solves partial differential equations (PDE) in boundary integral form. This method calculates only boundary values rather than values throughout space and is computationally efficient in case of a small surface / volume ratio. However, in contrast to several other methods, elements are not only locally connected to other elements, which results in a square increase of computer resources and time with growing problem size. This method is restricted to problems for which Green functions can be calculated. These usually involve fields in linear homogeneous media, which is not desirable in this project [6].

The finite element method (FEM) is used for finding approximate solutions of partial differential equations and integral equations. The solution approach is based either on eliminating the differential equation completely (steady state problems), or rendering the PDE into an approximating system of ordinary differential equations, which are then numerically integrated using standard techniques such as Euler's method, Runge-Kutta, etc. The mesh elements of the FEM method are usually triangular, which gives the ability to approximate complicated geometries with relative ease. Since elements are only locally connected, the computational resources and time grow about linearly with the problem size [7].

The finite difference time domain (FDTD) method uses discretized time-dependent Maxwell equations to calculate EM fields in time. This is done using square elements containing both

electrical and magnetical field properties. Calculation of the fields is done in a leapfrogging manner. Since it is a time-domain method, solutions can cover a wide frequency range with a single simulation run. This method has been originally developed for electromagnetic problems. In contradiction to FEM, computational resources are not only dependent on the problem size, but also on the number of time steps to be used. The advantage of FDTD is its ability to show field propagation throughout space, but this comes at the cost of a longer simulation time. Due to its time-domain nature, it is not practical to simulate low frequencies with relatively small cells (most dynamic phenomena can however be simulated using appropriate scaling) [8].

For every method several software packages are available, both commercial and open source. Most open source / freeware packages have restrictions that are unwanted in this project (e.g. limitations to 2D problems, specific operating systems, etc). To get maximum freedom and insight in the development of the model, the decision has been made to implement a model from scratch. This way a dedicated model is developed with only features that are needed. However, some freeware packages were used to compare results. The method that has been chosen is FDTD because of its ease to implement and its focus on electromagnetic problems. The time aspect might be useful in future research.

3.2 The finite difference time domain method

The FDTD time stepping algorithm and grid were proposed by Kane Yee in 1966 [9]. He thought of a way to calculate and represent electric (E) and magnetic (H) fields in an easy way, using cubed cells to form a spatial grid. Each cell contains both the electric field components (on the edges of the cube) and the magnetic field components (perpendicular to the faces of the cube). If both fields are split into separate cells, the magnetic field will be shifted half a cell with respect to the electric field. Figure 9 shows this. The right hand figure already shows how the Ampere-Maxwell law is implemented (a change in E_x causes rotation of the magnetic field components H_y and H_z).

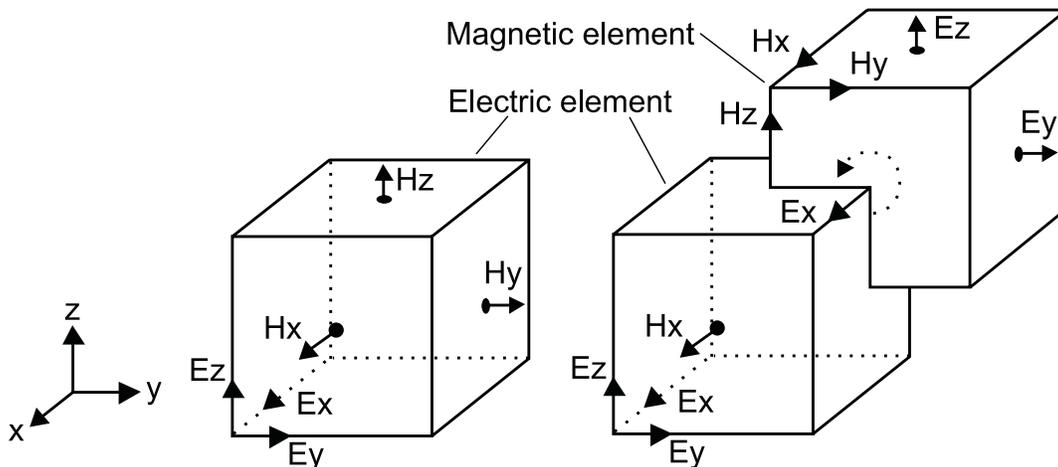


Figure 9: A single Yee cell (electric) and a combination of an electric and a magnetic element

Field updates take place in a leapfrogging manner. This means that from a given situation in time one field is calculated using values of the other field that remains constant. Thereafter the other field is calculated while the first field remains constant. This time stepping algorithm is shown in figure 10. Here the subscript (x, y) indicates the field component in Cartesian coordinates. A time step is denoted as 'n' and a grid position by 'k'. The figure shows that first for time step n-1/2 all E_x components in the grid are calculated. Thereafter all H_y components are calculated for time n. Finally the next time step for E_x is calculated. This is an example of a 1-dimensional situation (only E_x and H_y components). In case of three dimensions, first all components of one field (e.g. $\{E_x, E_y, E_z\}$) are calculated before proceeding to the other field.

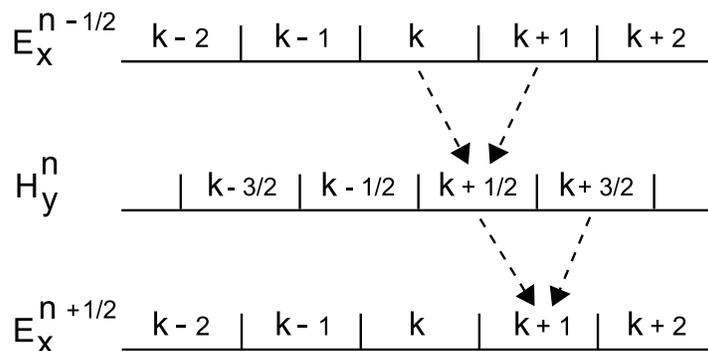


Figure 10: Fields are updated in a leapfrogging manner

It can be derived that the update equations for figure 10 can be written as formulas 7 and 8 [10].

$$E_x^{n+1/2}(k) = E_x^{n-1/2}(k) - \frac{\Delta t}{\varepsilon_0 \cdot \Delta x} [H_y^n(k+1/2) - H_y^n(k-1/2)] \quad (7)$$

$$H_y^{n+1}(k+1/2) = H_y^n(k+1/2) - \frac{\Delta t}{\mu_0 \cdot \Delta x} [E_x^{n+1/2}(k+1) - E_x^{n+1/2}(k)] \quad (8)$$

These formulas are only valid for 1-dimensional EM waves propagating in free space and for a grid existing of cubic cells (with edge length Δx). The time step for updating is denoted as Δt . ε_0 is the permittivity of free space and μ_0 the permeability of free space. This basic form can be extended to a more useful form including permittivity (ε), permeability (μ) and conductivity (σ). This way each cell can be assigned a specific material. It is also possible to assign current density sources (electrical (J) and magnetical (M)) to a cell. Full explicit update equations can be found in [8]. In order to keep these equations clear to understand, they are generalized first (formulas 9-10)¹.

$$E^{n+1/2}(k) = Ca \cdot E^{n-1/2}(k) + Cb \cdot (\text{curl}_H - J) \quad (9)$$

$$H^{n+1}(k+1/2) = Da \cdot H^n(k+1/2) + Db \cdot (\text{curl}_E - M) \quad (10)$$

Constants Ca , Cb , Da and Db depend on material properties and the time step size used during

¹It appears that 'M' ([A/m]) and curl_E ([V/m²]) cause a unit conflict. Reference [8] is not clear on this issue. A possible solution could be to use $\nabla \times M$ in the E-field update equation instead (bound current).

the simulation. These values are determined before the simulation starts and are defined for every Yee cell. A common way to express these constants is given in equations 11 and 12 for constants Ca and Cb . Similar equations can be derived for Da and Db by replacing electrical conductivity (σ_E) for a magnetic loss term² (σ_H) and ε for μ .

$$Ca = \frac{1 - \frac{\sigma_E \cdot \Delta t}{2 \cdot \varepsilon}}{1 + \frac{\sigma_E \cdot \Delta t}{2 \cdot \varepsilon}} \quad (11)$$

$$Cb = \frac{\frac{\Delta t}{\varepsilon}}{1 + \frac{\sigma_E \cdot \Delta t}{2 \cdot \varepsilon}} \quad (12)$$

From the equations can be seen that if the conductivity equals zero, the medium is lossless. E.g. an EM wave in free space will propagate forever without being damped. Both Ca and Cb decrease with an increasing conductivity. This means that both the own ('previous') field strength as well as the influence of the other field is limited. A different situation occurs as ε or μ is relatively high. It can be seen that in this case the own field remains relatively strong, while the influence of the other field is limited. An example of this is flux concentration in a steel plate (relatively high μ).

Two values are needed to update a field component: the previous value of the component and curl of the other field at that location in space. The curl is determined in the plane perpendicular to the component to be calculated. This is done by taking the difference between two adjacent cells divided by the cell size in that direction. Equation 13 shows an example of this for the curl needed to update an x-component of the H-field at time step $n + 1$.

$$curl_{Ex}^{n+1/2} = \frac{E_y^{n+1/2}(k+1) - E_y^{n+1/2}}{\Delta z} - \frac{E_z^{n+1/2}(j+1) - E_z^{n+1/2}}{\Delta y} \quad (13)$$

3.2.1 The perfectly matched layer

The previous section discussed the implementation of the FDTD method for an infinite large space. In practice however, the model has a finitely size. If the values just outside the model are assumed to be zero, huge reflections will occur because the 'energy' stored in the waves cannot simply vanish. As a result the model outcome is useless. Therefore it is needed to define absorbing boundary conditions (ABC) that can handle the waves. In a one dimensional case it is possible to absorb the waves exactly, but for higher dimensions wave interactions become too complex to be solved exactly. From all the research that involved this problem, the *perfectly matched layer* (PML) became the most popular solution. This method implements an absorbing layer all around the model. The idea is that at the interfaces of the PML and model match exactly (impedance matching) such that no reflections occur. Thereafter the conductivity increases exponentially with increasing distance to the model (the smooth curve in figure 11).

²In simulations this variable is set to 0. A different formulation for 'nonuniform grids' mentioned in [8] does not involve this loss term at all.

In practice the PML has to be discretized into a finite number of steps. This discretization is a trade-off between a smooth interface transition and small discretization error deeper into the PML (the steeper the curve, the larger the error). Usually the curve is a 3rd to 4th order polynomial. Figure 11 shows an example of a 6-cell wide PML.

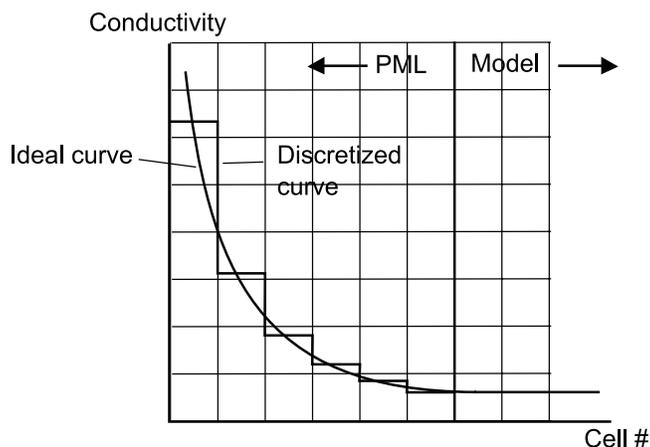


Figure 11: Conductivity curve in PML

To determine the optimal maximum conductivity of the PML (σ assigned to the outer cells) formula 14 can be used [8]. In this formula η is the wave impedance ($\sqrt{\frac{j\omega\mu}{\sigma + j\omega\varepsilon}}$), Δx the edge length of a PML cell and m the order of the polynomial used. In simulations η is always set to the wave impedance of free space ($\approx 377 \Omega$) because most of the radiating energy enters the PML from air cells. If waves propagate through several layers of steel cells (often > 10), the remaining energy can be neglected in comparison with energy from waves propagating through air. The advantage of this approach is that the conductivity of a PML cell is only position dependent (figure 11).

$$\sigma_{opt} = \frac{0.8 \cdot (m + 1)}{\eta \cdot \Delta x} \quad (14)$$

Besides the shape of the conductivity curve inside the PML, the number of discretization steps is also of importance. A large PML will result in small reflections, but adds a significant computational burden. On the other hand, if the PML is too small reflections may be unacceptable. Usually a good choice is about 6 to 10 cells.

The above described PML is known as *uniaxial PML* (UPML) and is a reformulation of the original *split-field PML*. It is one of the most popular PML's because of its efficiency and simplicity. An even better PML has been developed using stretched-coordinates. This PML is a little more complex and adds extra variables to the update algorithm. For simulations in this project the UPML has been used because of its robustness.

3.2.2 Programming language

To implement the model a programming language has to be chosen. The most important factor is computational speed. To determine the best choice a simple test has been performed. A small

program has been written in M-code (Matlab), C and Fortran. Matlab has some visualization tools but is known to be worse at dealing with 'for'-loops. Fortran is the language that is used by most scientists because of its 'number crunching' capabilities. It is used for example in weather forecasts. C is a widely used and well-known language.

The test program simulates a Gaussian pulse propagating in 1D space (see figure 12). It is an example containing the most elementary update equations (like previously mentioned formulas 7 and 8). When the pulse hits the border it is simply reflected back into the problem space. Since the pulse is initiated only once, this example will remain numerically stable even for a large number of time steps.

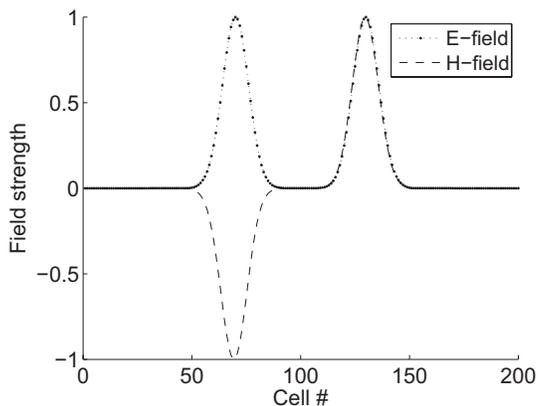


Figure 12: 1D wave used for test

The test has been performed for several numbers of time steps for every piece of code³. The results can be found in table 1. As expected, Matlab is not a good choice because *m-code* (Matlab code) is clearly the slowest language. The other two languages perform about equal, with Fortran having a slight advantage (3% relative to C).

Time steps	Calculation time (sec)		
	Matlab	C	Fortran
10^3	0.9	< 0.1	< 0.1
10^4	7.4	< 0.1	< 0.1
10^5	70.5	0.5	0.4
10^6	–	2.4	2.3
10^7	–	21.0	20.4
10^8	–	207.2	201.2

Table 1: Results of the computational speed test for 3 different languages

From this test, C has been chosen as being the language to be used. Although it is slightly slower than Fortran the author is familiar with C but completely new to Fortran. The time it would cost to learn a new language is a bigger disadvantage than a 3% loss in simulation time.

³All tests were executed on a 'standard' PC (2.5 GHz dual core processor and 3 GB of internal memory)

4 Modeling of the eddy current measurement system

The theory explained in section 3.2 is now applied to implement a model of the eddy current measurement system. First only a simple coil is modeled to compare results with analytical formulae. Thereafter the coil will be included in the complete system.

4.1 Model of an air-cored coil

The modeling procedure starts with determining the dimensions that need to be modeled. A single coil in the real eddy current sensor has the following specifications:

- Number of turns: 40 (10 turns x 4 layers)
- Wire diameter: 0.18 mm (round)
- Shape: square (inner diameter: 22 x 10 mm)
- Core material: epoxy glass

From these specifications it can be seen that a round wire is used, while in the model only rectangular cells can be used (in this project only cubic cells are used for simplicity). This problem can be mitigated using a much smaller cell size with respect to the wire diameter and thus using more cells to approximate the wire shape. However, if the total size of the coil is taken into account, this is not desired. Even if a cell size of 0.2 mm is used (approximation of the wire diameter), a model of at least 118 x 10 x 58 cells is required (exclusive PML!).

A method called *thin wire modeling* assumes thin wires to be infinitely small and situated in the middle of a square of 4 adjacent cells. The effect of this 'virtual' wire can be simulated by modeling the fields in the cells around it. However, as tests will show later on in this chapter, it is not necessary to implement this (more complex) method. By using a real cell to implement the wire, current (density) can easily be assigned as shown in figure 13. The FDTD algorithm takes care of the field propagation caused by the flowing current.

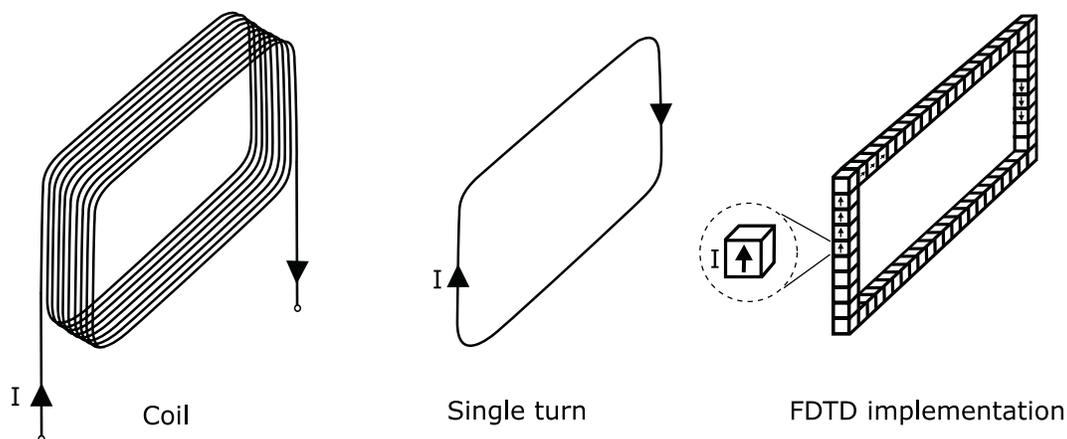


Figure 13: Implementation of a single coil turn in FDTD lattice

Although in the real system a multi-layer coil is used, for the time being a single turn will be simulated. The reason is that reliable analytical formulas for the self-induction of a square coil

are only available for single layer, single turn coils. One formula proposed by Terman [11] is given in formula 15. This formula assumes air to be the core material (which is a good approximation of the real core material). Note that this formula is only an approximation based on the geometry of the coil. This formula is used in [5] as reference for other, more simplified formulas.

$$L = 0.4 \cdot ((a+b) \cdot \ln\left(\frac{4.0 \cdot a \cdot b}{d}\right) - a \cdot \ln(a+g) - b \cdot \ln(b+g)) + 0.4 \cdot (2.0 \cdot g + d - 2.0 \cdot (a+b)) \quad [\mu H] \quad (15)$$

In this equation a and b are the sides of the loop in meters and d the diameter of round wire (also in meters). The rectangle diagonal is denoted by g ($=\sqrt{a^2 + b^2}$).

To simulate the single turn coil first the time step has to be determined. This time step depends on the cell size according to formula 16, known as the *Courant condition*. By obeying the Courant condition numerical stability is ensured.

$$\Delta t = \frac{1}{\sqrt{\left(\frac{1}{\Delta x}\right)^2 + \left(\frac{1}{\Delta y}\right)^2 + \left(\frac{1}{\Delta z}\right)^2}} \cdot c \quad [s] \quad (16)$$

In this formula Δx , Δy and Δz are the edges of the Yee cell. The time step is denoted by Δt and c represents the speed of light. The cell size itself is determined either by the smallest geometry or by the highest frequency to be simulated (it depends on the model which of the two is critical). In the latter case a rule of thumb can be used to define the maximum cell size (formula 17).

$$\Delta x_{max} = \frac{\lambda}{10} = \frac{c}{10 \cdot f} \quad [m] \quad (17)$$

With λ being the wave length, c the speed of light and f the frequency.

Eddy current testing is usually done at a frequency of 10 to 110 kHz. If the real coil dimensions are used ($\Delta x = 2 \cdot 10^{-4} \text{ m}$), a time step (Δt) of 0.385 picoseconds is required (formula 16). It would however be practically impossible to simulate a 110 kHz EM-wave, since it requires $\frac{1}{f} \cdot \frac{1}{\Delta t} = \frac{1}{1.1 \cdot 10^5 \cdot 3.85 \cdot 10^{-12}} = 2.3$ million time steps to simulate only one period. Therefore a higher frequency is chosen to show proper behavior of the coil.

Simulating at a (much) higher frequency is not a problem if the coil is placed in free space, but the situation changes as soon as frequency-dependent objects (like steel) are present. Most important in eddy current testing is the change in standard depth of penetration, as mentioned in section 2.1 (formula 2). To still be able to simulate at a much higher frequency, the original model is scaled down. This works due to the fact that the frequency appears as square root in formula 2. Figure 14 shows in 3 steps what happens. The original situation is depicted in the left figure. The middle figure shows that the frequency is multiplied by n ($n > 1$) to be able to reduce simulation time by factor n . As a result, the standard depth of penetration decreases by \sqrt{n} . To compensate for this loss in depth (expressed in cells), the cell size is reduced by \sqrt{n} (right figure). However, the maximum time step is linearly depending on the cell size. This results in an increased simulation time of factor \sqrt{n} . Summarized: increasing the frequency by a factor n yields a reduction in simulation time by $\frac{n}{\sqrt{n}}$.

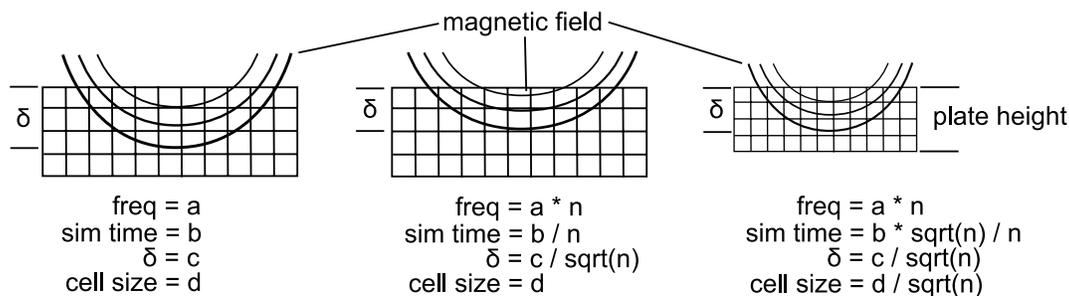


Figure 14: Down-scaling of the model after frequency increase

4.1.1 Coil calculations

One way to determine the accuracy of the implemented coil is by calculating its self-inductance based on simulation results. The calculated value can be compared to the analytical result of formula 15.

In order to obtain the self-inductance by simulation, an alternating current with frequency f_0 is 'assigned' to the cells being part of the coil. The direction of the current in each cell depends on its position in the coil (recall figure 13). By using this approach the current is exactly known, while an electromotive force (emf, symbol V) is induced in the coil as a result of flux changes (formula 18). By Fourier transforming (\mathcal{F}) both emf (formula 19) and current (formula 20), and dividing these quantities with respect to each other, the (absolute) impedance is obtained (formula 21) [12]. The self-inductance can then be found by dividing $|Z|$ by ω_0 (with $\omega_0 = 2\pi f_0$) as shown in formula 22.

$$V = -N \cdot \frac{d\phi}{dt} \quad (18)$$

$$\mathcal{F}(V(f_0)) = \sum_n V(n\Delta t) \cdot e^{j\omega_0 n\Delta t} \quad (19)$$

$$\mathcal{F}(I(f_0)) = \sum_n I(n\Delta t) \cdot e^{j\omega_0 n\Delta t} \quad (20)$$

$$|Z| = \frac{\mathcal{F}(V(f_0))}{\mathcal{F}(I(f_0))} \quad (21)$$

$$L = \frac{|Z|}{\omega_0} \quad (22)$$

In these formulas N is the number of coil turns⁴. Flux is denoted by ϕ and n is the number of time steps (Δt).

4.1.2 Simulation results

As shown in figure 14, simulation of a complete eddy current measurement system cannot be done at millimeter level using FDTD. If down-scaling is applied to get a reasonable depth of

⁴This is actually only valid for infinitely long coils, because of flux divergence at both ends of the coil. In simulations only the field component perpendicular to a turns surface is taken into account to solve this.

penetration (at least a few cells, depending on the plate thickness) at a reasonable amount of time steps (1000 ~ 10000), it turns out that the cell size must be set to about $5 \cdot 10^{-13}$ m. A frequency of about $5 \cdot 10^{17}$ Hz can then be simulated within a reasonable time (less than 1 hour) while EM fields can penetrate about 10 cells. Although depth of penetration is not an issue while testing the coil (there is no plate involved), above mentioned values are used to make sure the models work in all coming tests.

The inductance test described in section 4.1.1 has been executed several times using varying parameters. One parameter is the frequency. According to the rule of thumb for the maximum frequency given a certain cell size (formula 17), the frequency may not exceed $\frac{c}{10 \cdot \Delta x} = \frac{3 \cdot 10^8}{10 \cdot 5 \cdot 10^{-13}} = 6 \cdot 10^{19}$ Hz. To make sure that the effect of the frequency below this value is negligible, various frequencies have been tested.

The model size is also of interest. When the PML is relatively far away with respect to the coil, fields can propagate in space without being affected too much. On the other side the simulation time grows with a growing model. Several 'margins' have been tested (the number of cells between PML and coil). All simulation runs use a PML width of 7 cells. Results of the calculated and analytical self-inductance (always $4.90 \cdot 10^{-17}$ H) can be found in figure 15.

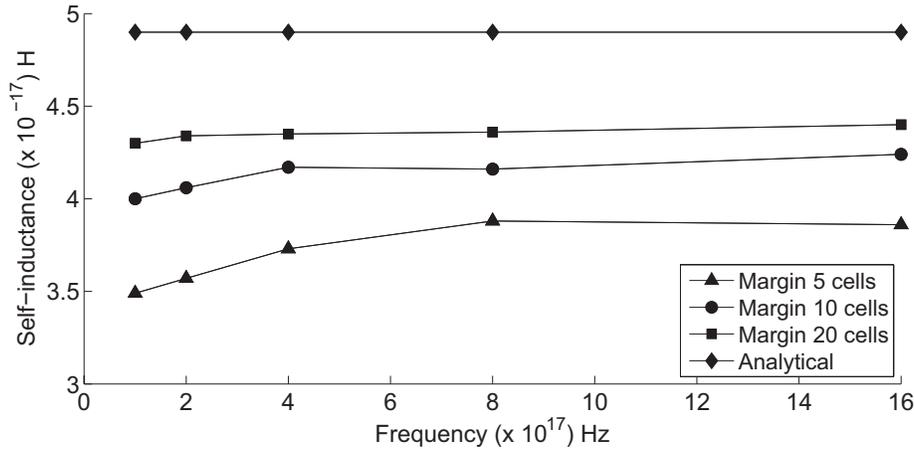


Figure 15: Testing the coil model using varying frequency and margin

The test results show that an increasing model size improves the accuracy of the model calculations. This is an expected result, because the PML absorbs energy from penetrating field lines. The less field lines penetrate, the less energy is lost. This is also the reason that all calculated self-inductance values are lower than the analytical value.

An increasing frequency appears to have a positive effect on the simulation results. This is however against the expectations, because accuracy decreases with increasing frequency (lesser time steps per period). The reason that the calculated values increase with the frequency results from bigger reflections of the PML (a smaller part of the energy is absorbed by the PML).

The following test shows that downscaling has negligible effect on a static problem. Two coils

with square cross-section (27 x 27 cells) have been simulated with a DC current of 1 A. One using unscaled cells of $2 \cdot 10^{-4}$ m, while the other using downscaled cells of $5 \cdot 10^{-13}$ m. Figure 16 shows results of the H-field strength in the direction of the main axis. For the H-field inside a solenoid, formula 23 can be used. Here N is the number of turns, I the current strength (in A) and l the length (in m) of the coil [13].

$$H = \frac{N \cdot I}{l} [A/m] \tag{23}$$

For the model this formula reduces to $H = \frac{I}{\Delta x}$, because all turns have length Δx . The theoretical field strength for figure 16a becomes 5000 A/m and for figure 16b this value is $2 \cdot 10^{12}$ A/m. These values are not reached because the coils simulated are too short to approach a solenoid. However, it can be seen that both coil fields have the same shape and relative 'error' with respect to the solenoid values (about 40 %).

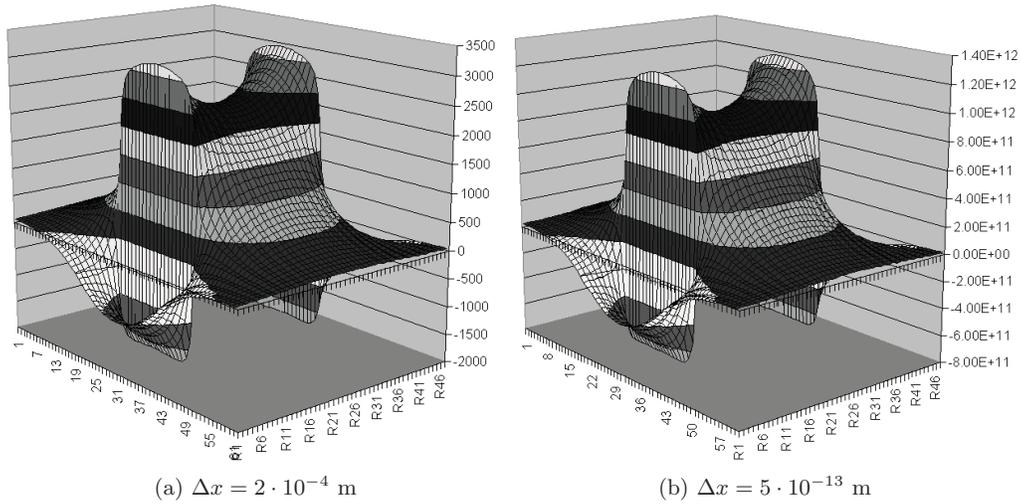


Figure 16: Comparison of coil H-fields using different cell sizes

It can be concluded that the model is sufficiently accurate for margins varying from 5 till 20 cells and a frequency well below the limit of formula 17. Since the coils are used in a bridge circuit, it can be expected that the absolute errors shown in figure 15 almost completely cancel out (both coils encounter about the same error).

4.2 Model of the complete system

The coil described in the previous section is part of an eddy current sensor. The eddy current sensor consists of 2 coils in bridge configuration, as mentioned in section 2. An array of these sensors is mounted in a frame below a big magnet to enlarge the scanning area⁵. A schematic drawing of the system and eddy current sensors is depicted in figure 17. The left figure shows how the system is built up. On the top side big magnets are mounted to induce a DC magnetic field. The field is guided through the yokes on both sides down to the plate (not drawn here). The poleshoes improve the connection between yokes and plate. Below the magnet, aluminum brackets are mounted to hold the sensors⁶. The thick black lines in the figure represent the coils. The 2 coils in the right drawing of figure 17 are comparable to the more detailed coil in figure 13. Note that the coils are wound in opposite direction with respect to each other and that they exist of multiple layers (see the summary at the start of this chapter).

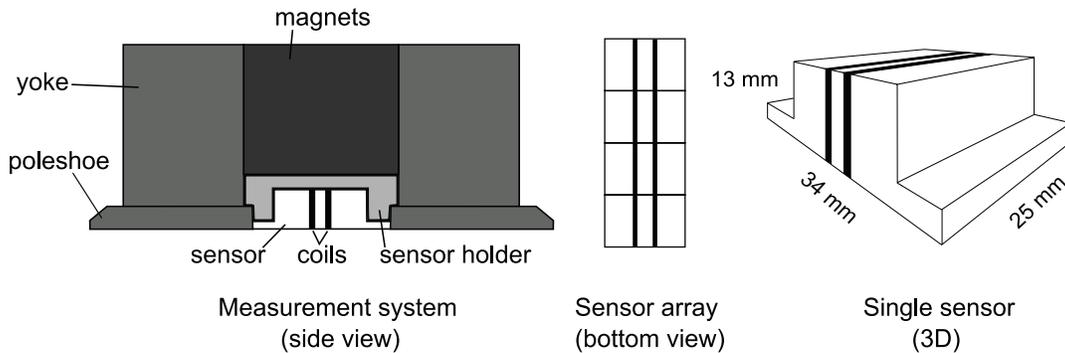


Figure 17: Schematic drawing of the measurement system

4.2.1 Unmagnetized scanning

For the time being, the magnetization process is omitted. First only the eddy current sensor and steel plate are modeled to obtain results that can be compared to the situation *with* magnetization later on. The steel plate has the following parameters:

- Relative permeability (μ_r): 5000
- Conductivity (σ): $10^7 \text{ S}\cdot\text{m}^{-1}$

A magnetization curve is not implemented, because it is assumed that measuring at a frequency of more than 10 kHz does not magnetize the plate. In FDTD lattice, the plate is always positioned in the horizontal plane (xy-plane), starting at the bottom cells. It is assumed that it always spans an infinite surface (that is, it continues inside the PML). As a result, only *height* (z-direction) will be varied during tests. The sensor is modeled as 2 coils in parallel, like in the right drawing of figure 17.

If the system would have been modeled at real size, the wire thickness determined the cell size:

⁵The scanner used for practical measurements in this project contained 4 sensors

⁶For practical reasons (e.g. simulation time) only 1 sensor will be modeled.

$2 \cdot 10^{-4}$ m (approximately). The dimensions of the coils and plate then become:

- Inner dimensions coil: 110 x 50 cells (0.022 x 0.01 m)
- Outer dimensions coil: 118 x 58 cells (inner dimensions + 4 layers)
- Length coil: 10 cells (10 parallel turns)
- Distance between coils: 18 cells (0.0036 m)
- Plate height: 30 cells (0.006 m)
- Maximum lift-off: 25 cells (0.005 m)

Besides these dimensions, holes with diameters varying from $1 \cdot 10^{-3}$ m to $2 \cdot 10^{-2}$ m (5 - 100 cells) need to be detected. To simulate a scanning motion, the hole moves in a number of steps underneath the sensor. As a result, the y-axis consists of at least twice the hole diameter plus the space needed by the coils. If also free space around the coil is taken into account (about 15 cells), the total size becomes roughly: 150 x 240 x 130 cells. A total of 4.7 million cells. If an PML of 7 cells is added, the total number of cells increases to almost 6 million. Practice shows that these numbers of cells cannot be simulated in a reasonable amount of time with a sufficient number of time steps (1000 - 10000). The easiest way to get rid of this problem is to sacrifice some accuracy. This has been done by using a single turn coil (1 layer), reducing free space to 10 cells and limiting the hole size to 0.0016 m. The reduced hole size is a result of the practical measurements that are used in chapter 5. These reductions result in a model size⁷ of 100 x 120 x 68 cells + PML. A total of about 1.3 million cells. Using this reduction, simulation of a static model (1 position of the hole with respect to the sensor) takes about 1 hour to complete. Figure 18 shows a 3D view of FDTD lattice containing 2 coils and a plate that is used for simulations. Note that for the sake of clarity not all cells are drawn.

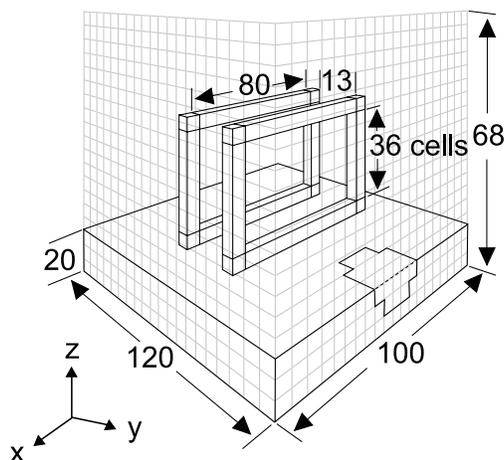


Figure 18: 3D view of implemented model

As already mentioned, the model will be down-scaled to keep the simulation time relatively short. Two models have been simulated to show the effect of down-scaled cell sizes. Except for cell size and simulation frequency, both models are exactly the same. The dimensions of figure 18 are used. One model contains cells with edge sizes of $\Delta x_1 = 1 \cdot 10^{-13}$ m, the other uses

⁷The z-value is 68 when the lift-off is 0. Cells are added if the lift-off increases.

edge sizes of $\Delta x_2 = 0.25 \cdot 10^{-13}$ m. The respective frequencies used are $f_1 = 5 \cdot 10^{18}$ Hz and $f_2 = 2 \cdot 10^{19}$ Hz. During the test, a round hole with a diameter of 16 cells was moved from one side of the plate to the other. Different depths have been simulated: 25%, 50% and 100% of the plate thickness. Figure 19 shows a simplified schematic overview. The left figure shows a top view of the model, the right figure a side view representing various hole depths.

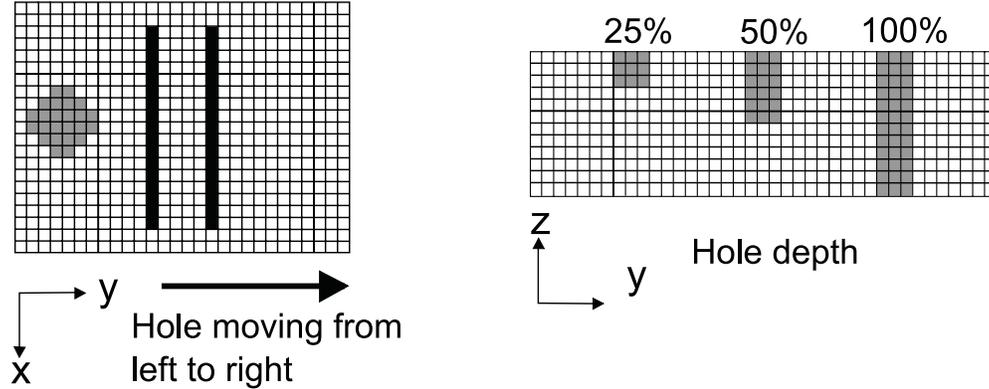


Figure 19: FDTD lattice representation of executed tests

The test results are represented in a complex impedance plane (figure 20). It can be seen that both graphs are almost identical. They both show a *figure 8* shape (that is, two loops of about equal size with an intersection in the middle). This figure results from the way the sensor is moving over the hole. At the starting position (no hole present), both coils generate the same amount of eddy currents in the steel plate. The difference is then -in theory- equal to zero. When the sensor approaches a hole, the complex impedance of the front coil starts to change. The 'peek' (maximum absolute impedance difference) value is reached when the hole is right below the front coil. Thereafter, the impedance difference decreases until the hole is right below the back coil (negative peek). When both coils move away from the hole, the impedance difference returns to its initial position in the plane.

The results show a relatively large offset with respect to the amplitudes of the figures. This offset is a result of numerical errors. Although the offset *looks* big, it is not. The absolute impedances of the coils are about 600Ω , while the difference impedance is about 1.7Ω (0.28 %). In fact, the effect of the hole on the signal is relatively small.

If the self-inductance is calculated by formula 15, it can be concluded that $L_1 = 2L_2$. Along with $f_1 = 0.5f_2$ it turns out that $Z_1 = R_1 + j\omega_1 L_1 = R_1 + j \cdot 0.5\omega_2 \cdot 2L_2 = R_1 + j\omega_2 L_2$. This confirms that both graphs must have equal amplitudes.

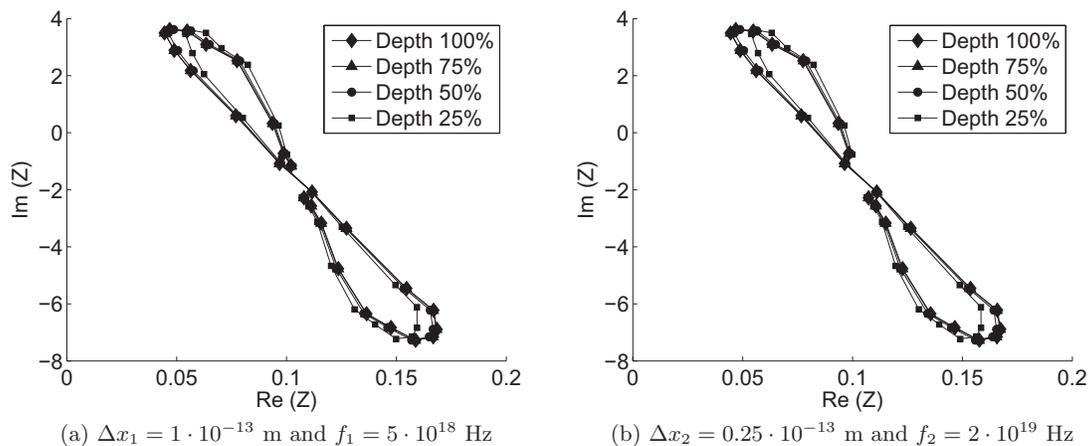


Figure 20: Impedance plots of the test results using different cell sizes and frequencies

The orientation of both figures is also nearly equal, which is an expected result. The orientation of the figures is mainly determined by the *phase lag*. Phase lag can be approximated by formula 24 [2].

$$\theta = \frac{x}{\delta} \text{ radians} \quad (24)$$

Here θ is the phase lag in radians, x is the distance of the defect below the surface and δ is the standard depth of penetration. x is often a weighted average over the distance spanned by a defect. This can also be seen from the little shift among different hole depths in figure 20. The variation in phase lag is limited due to the fact that the eddy current density exponentially decreases, and all holes start at the top side of the plate.

4.2.2 Magnetized scanning

As can be seen from figure 17, big magnets are used to magnetize the steel plate during scanning. Magnetization mitigates the influence of varying magnetic properties within the steel plate. During the magnetization process, *magnetic domains* align in the direction of the applied magnetic field. Magnetic domains are small regions in which individual moments of atoms are aligned with one another. Within a domain of unmagnetized ferromagnetic material (e.g. steel) the net moment is approximately zero. In this state, the free magnetic energy is minimized. The left square in figure 21 shows such a domain. The applied field from the magnet forces the domain walls to move such that the component in the direction of the applied field increases while the others decrease (middle square). If the domain is completely aligned with the external field, the material is saturated (right square).

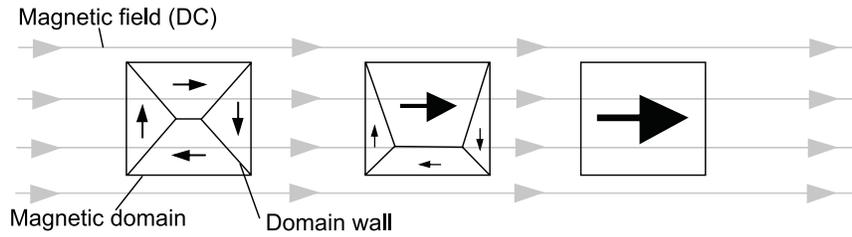


Figure 21: Magnetization process in 3 steps after applying an external magnetic field

Figure 7 of chapter 2 shows a hysteresis curve. This phenomenon originates if the external field is varied between positive and negative saturation due to microscopic phenomena. Hysteresis is not used in the model, because it would add unnecessary complexity. Instead, a single magnetization curve is implemented such that for each H-field strength a unique B-field strength exists (and vice versa). Figure 22 shows an example. Note that the curve only consists of positive values, because it is point symmetric in the origin.

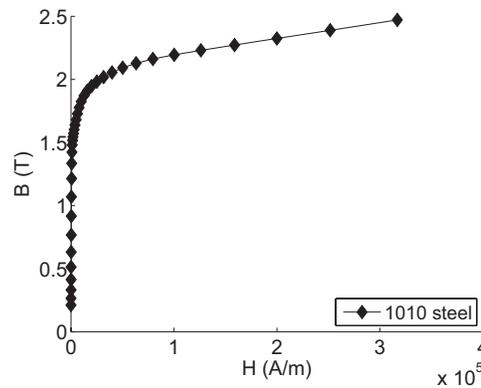


Figure 22: Magnetization curve of 1010 low carbon steel

From this curve the relative permeability can be approximated at each point via formula 25.

$$\mu = \frac{dB}{dH} \quad (25)$$

In the model, H-field components are calculated to represent the magnetic field. The general formula for an H-field update (recall formula 10) shows that magnetization vector (\mathbf{M}) is taken into account. Each cell of the model can thus be seen as a magnetic domain. The net length and direction of this vector are updated after each magnetic field (\mathbf{H}) update during simulation. \mathbf{H} , \mathbf{B} and \mathbf{M} are related by formula 26.

$$\mathbf{B} = \mu_0(\mathbf{H} + \mathbf{M}) \quad (26)$$

It is assumed that the permeability of the steel plate under investigation is *isotropic* (equal in all dimensions) although different cells can have different permeabilities at the same moment in time.

During an update cycle of the magnetization vectors in all (steel) cells, the following steps are taken:

- Calculation of $|H|$
- Calculation of $|B|$ via magnetization curve
- Calculation of μ_r via magnetization curve
- Update \mathbf{M} using formula 26
- Update of μ_r and related variables

The more or less homogeneous H-field needed to magnetize the plate is created by placing the plate inside a coil. The advantage of this approach is that few extra cells are needed (just 2 layers of coil cells in 'x' and 'z' direction (recall figure 18) while a relatively homogeneous field is created.

One of the problems that have been encountered by the eddy current scanner in the past was the strength of the magnets. When all magnetic domains are aligned and the steel plate is saturated, the \mathbf{B} -field will only increase linearly with \mathbf{H} by a factor μ_0 . The magnetization vectors remain constant beyond that point (this is confirmed by formula 26). Due to saturation, the relative permeability is reduced to about 1. The applied field 'sees' the steel plate as air. Figure 23 confirms this by showing 2 simulations of 1010 steel within a coil (cross-sections), both having a hole filled with air. In the upper figure the steel plate is not saturated (average $\mu_r \approx 10$), the field lines bend around the air hole. In the lower figure the applied field is much stronger and the plate becomes saturated ($\mu_r = 1$). The field lines now penetrate the hole. Note that this simulation was executed using a different software package⁸. Although an applied DC field that causes saturation does not contribute to hole detection by itself, the decreased μ_r causes the standard depth of penetration to increase. Practical measurements show however that full magnetization is not optimal.

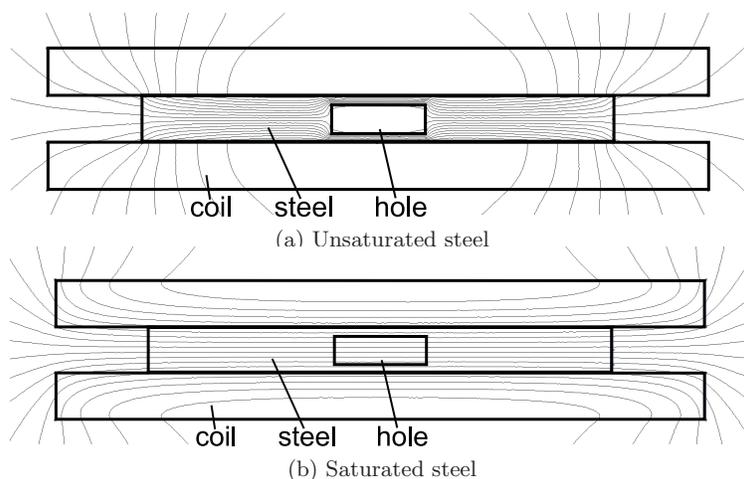


Figure 23: A FEM simulation of saturated 1010 steel with a hole

⁸A 2D FEM package was used for this simulation [14]

Unfortunately the currently implemented model suffers from instability if non-linear magnetization curves are used (like the one in figure 22). However, linear magnetization (constant μ_r) gives good results. Figure 24 shows the FDTD result of magnetization with $\mu_r = 10$. Although the representation is different with respect to figure 23a, both results are equivalent. In figure 24a the 'x'-component of the (B-)field strength is represented for all cells. Around the hole peaks show up. These peaks indicate that the field lines bend around the corner. Figure 24b shows the strength of the 'y'-components. It can be seen that the field inside the hole tends to 0 and while it is strong on the sides, just as in figure 23a. The simulation results shown have been performed with a coil length equal to the plate length. This results in relatively big influences on the corners/sides of the plate. These influences can be mitigated by temporarily stretching out the plate and coil during the magnetization, but this goes at the expense of additional simulation time.

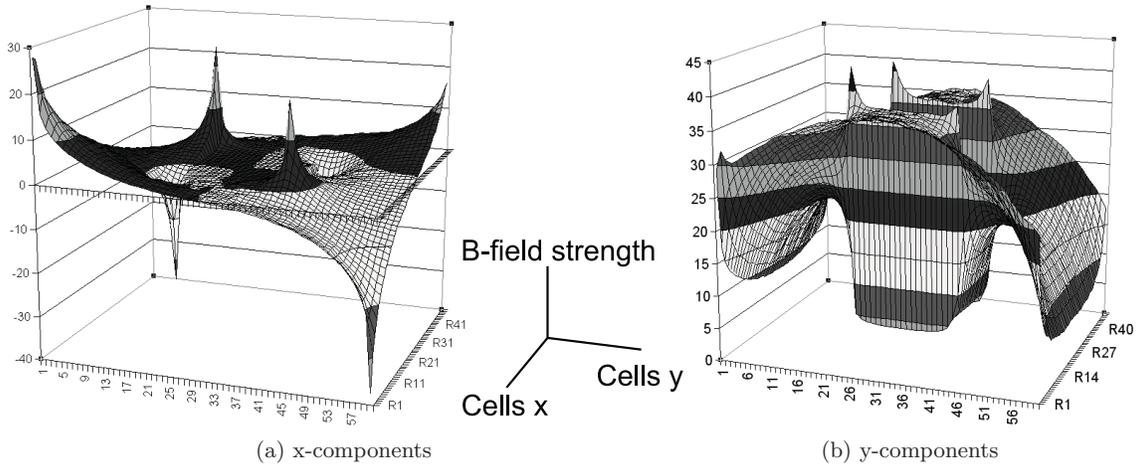


Figure 24: Steady state B-field after linear magnetization ($\mu_r = 10$)

From figure 23 can be concluded that magnetization is most effective if the strength of the external magnetic field only partially magnetizes the steel. Experiments point out that an average μ_r of about 10 gives good results. Besides the advantage of mitigating magnetical inhomogeneities, a lower μ_r also increases the standard depth of penetration (recall formula 2). This means that defects at greater depth can be detected without adapting (decreasing) the scanning frequency.

5 Measurement results

Several real measurements have been performed using a 4-channel scanner and a prepared steel plate. The plate contained two rows of 4 holes (25%, 50%, 75% and 100% depth). One row of holes having a diameter of 16 mm, the other row having a diameter of 10 mm. Figure 25 shows a tree structure of measurements and simulations that have been performed. The simulation settings of figure 18 (chapter 4) are used unless otherwise noted. It can be seen that not all combinations are shown in the tree. It is sufficient to compare different hole sizes in 1 test and then stick to the 16 mm holes⁹. Furthermore bottom holes can only be detected using a DC field, because of the limited standard depth of penetration of unmagnetized steel.

All measurement results exist of AD-converter (ADC) outputs. There is no information available that links the ADC values to impedance values. Therefore measurements can only be judged by their shape or compared to other measurements.

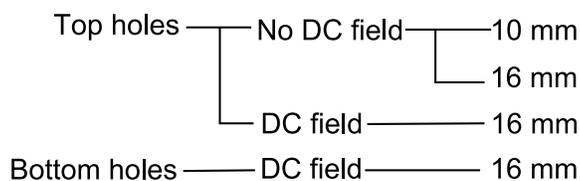


Figure 25: Measurements and simulations discusses in this chapter

Since the current BH-curve implementation suffers from instability, a linear approximation is used around a working point (figure 26). For the coming simulations, a μ_r of 100 is chosen, since this value ensures that the plate is sufficiently magnetized without being saturated. To determine the magnetization strength, the BH-curve of 1010 steel has been used (recall figure 22). The $\mu_r = 100$ point is reached as the H-field strength is about 1600 A/m and the B-field about 1.52 H/m. According to formula 26, the magnetization strength is then $1.21 \cdot 10^6$ A/m. Now the value for the H-field is calculated that realizes an equivalent magnetization strength using magnetic susceptibility: $H = \frac{M}{\mu_r - 1} = 1.22 \cdot 10^4$ A/m. This is the H-field strength used during simulations to magnetize the plate.

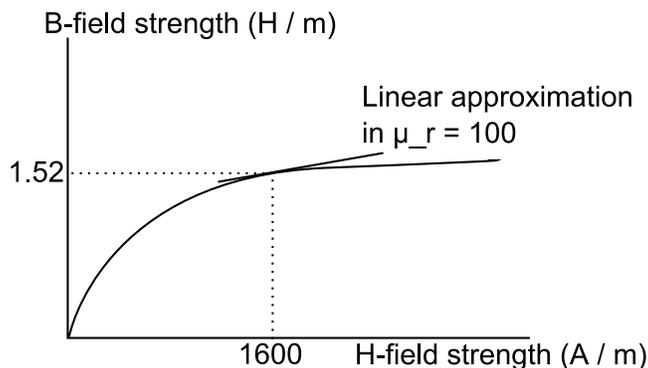


Figure 26: Linear approximation of point $\mu_r = 100$ of 1010 steel

⁹The 10 mm holes appeared to be less reliable due to somewhat unnatural properties

5.1 Top holes

In this section, top holes are investigated. These holes are situated at the side of the scanner. The depth of the holes is 25%, 50% and 75% of the plate height. Measurements have been performed with and without external DC field. Both will be discussed in subsections. Figure 27 gives a schematic overview of the test.

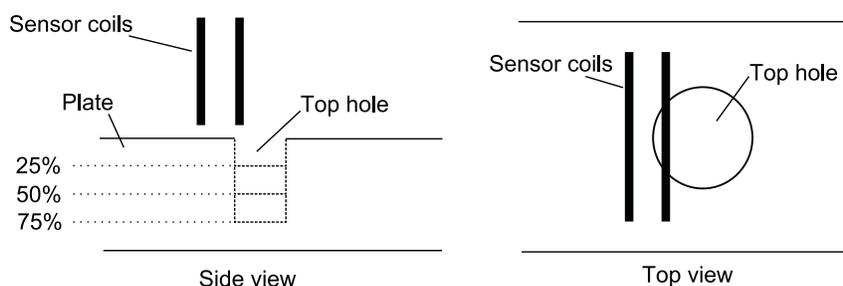


Figure 27: Side and top view of testing top hole responses

5.1.1 Tests without DC field

Measurements without DC field had to be performed by hand. That is, a single sensor was unmounted from the scanner and moved over the holes by hand (which may have a slight effect on the measurement results). Figure 28b shows results of a 16 mm top hole. From the measurement results can be concluded that the amplitude slightly decreases with an increasing hole depth. Only the 25% hole has significant loss in amplitude compared to the others. This indicates that the standard depth of penetration is relatively small: the eddy current density is rapidly decreasing at greater depth. The simulations show equivalent results.

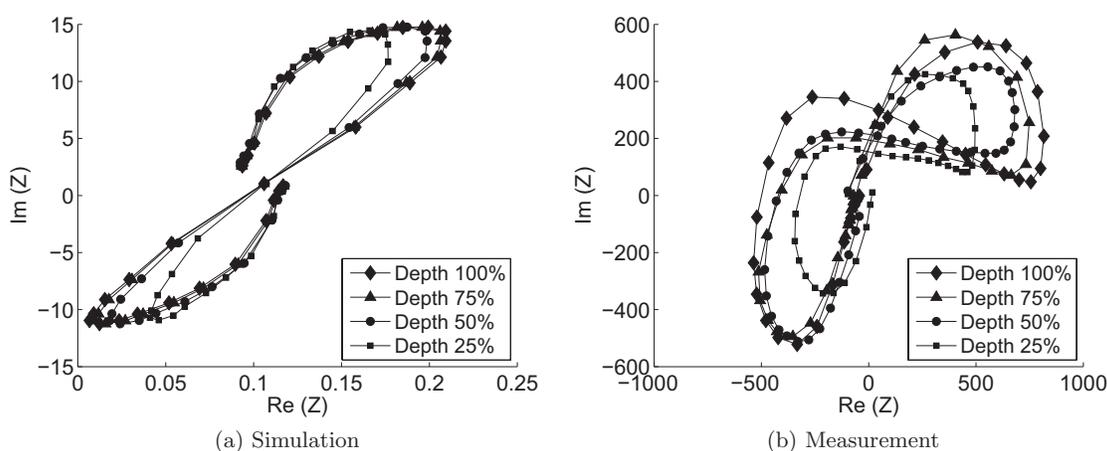


Figure 28: Comparison between top hole simulations and measurements (16 mm)

The same comparison can be made for holes with a diameter of 10 mm (figure 29). The most

important conclusion in comparison with the 16 mm holes is the decrease in amplitude. It seems that both simulation and measurement amplitudes are roughly divided by a factor 2. This is approximately linear with the hole surface ($2.0 \cdot 10^{-4} \text{ m}^2$ and $0.8 \cdot 10^{-4} \text{ m}^2$).

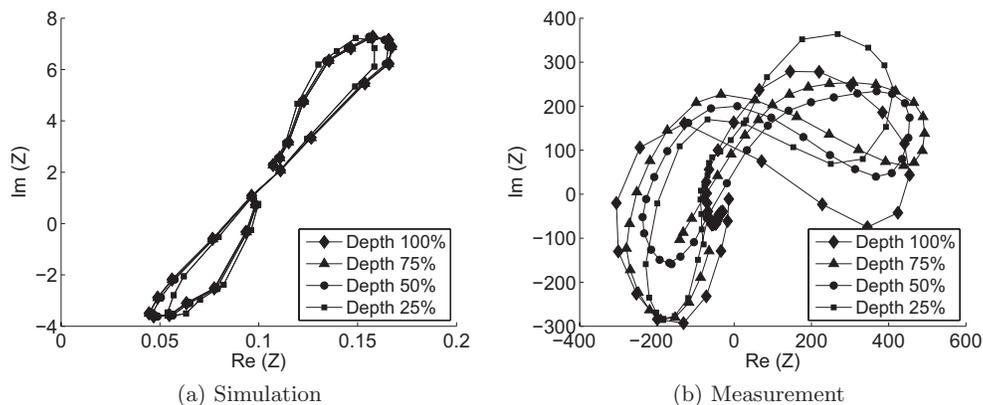


Figure 29: Comparison between top hole simulations and measurements (10 mm)

5.1.2 Tests with DC field

First a comparison is made between the measurements with and without DC field. Figure 30 shows this comparison. Apparently, the external DC field increases the sensitivity at larger depth (bigger differences among amplitudes). This is an expected result, because the applied DC field causes a decrease in permeability and thus an increase in penetration depth. Although the values at the axes are meaningless in absolute sense, they can be compared relative to each other. These values confirm that lift-off plays a big role. Scans *without* DC field were performed at minimum (less than 1 mm) lift-off, while measurements *with* DC field were performed at about 5 mm lift-off (due to assembly reasons).

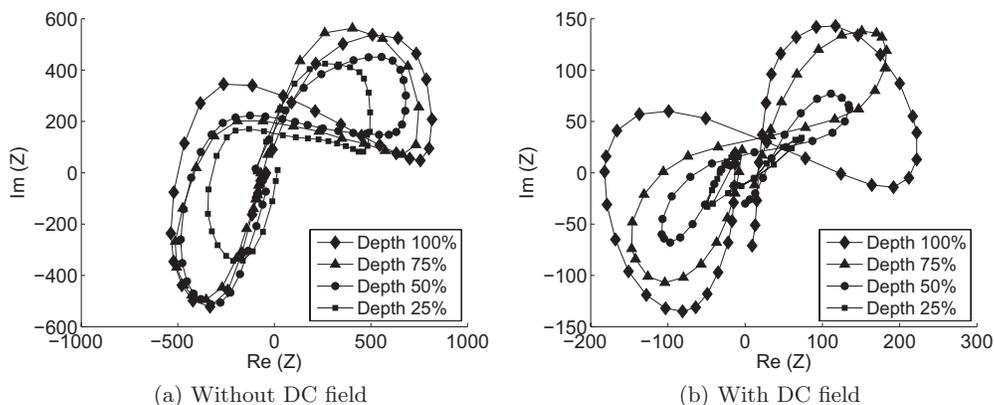


Figure 30: Top hole measurements with and without DC field

Figure 31 shows the simulation results of the top holes with and without DC field. It must be noted that lift-off does not play a role here (0 in both cases). For clarity only a 100% hole and a 25% hole are shown in the right figure. It can be seen that the amplitudes of these simulations do not differ much with respect to each other. This is conflicting with the measurements (which show more variety in amplitude). It is also strange that only the real part of the impedance increases with an applied DC field. In ECT theory, this implies that eddy current flow increases without increasing the opposing magnetic field. This is not confirmed by the measurements.

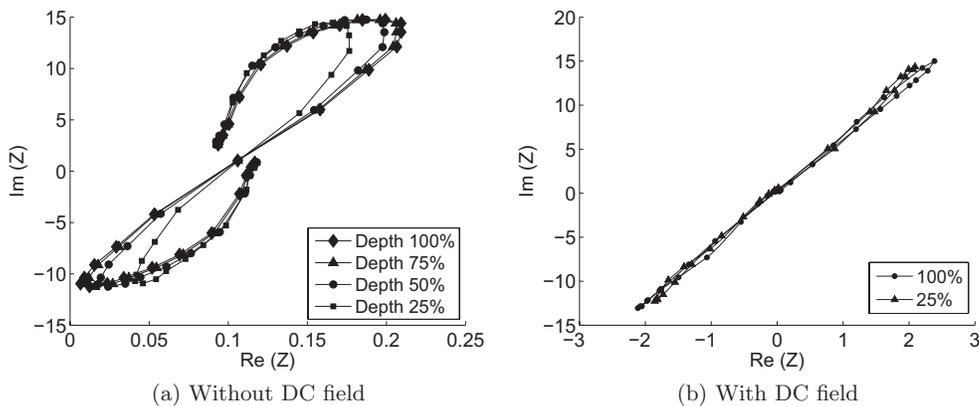


Figure 31: Top hole simulations with and without DC field

5.2 Bottom holes

Bottom holes can only be detected by the scanner if the DC field is active. If there is no DC field, the skin depth alone is simply too small to detect anything at the bottom side of the plate.

In this test, a smaller model has been used (60 x 70 x 40 cells) due to the lack of time for bigger tests. The modeled plate is 10 cells thick. The results of a 70%, 50%, 30% and 10% bottom hole are shown in the simulation result (figure 33a). These results look pretty good on first sight, because the amplitudes seem to be related with the hole depth in a '1/e' way and the orientation changes clearly with depth. Figure 32 shows the maximum amplitudes and orientation angles¹⁰ of the 4 simulations. The standard depth of penetration of the plate is 10 cells. According to formula 24 (phase lag), the angles must be about: 17° (70% hole), 29° (50% hole), 40° (30% hole) and 52° (10% hole). Since these angles are only relative to the orientation of the surface response (through hole, not shown) they can only be used to check the relative angles between simulations. It can be concluded that the simulated angles agree with the theoretical values. However, equivalent results were also obtained without DC field (Figure 34b). Therefore no clear conclusion can be drawn about the influence of the DC field.

¹⁰The angle between the longest axis of the figure and the Re(Z)-axis

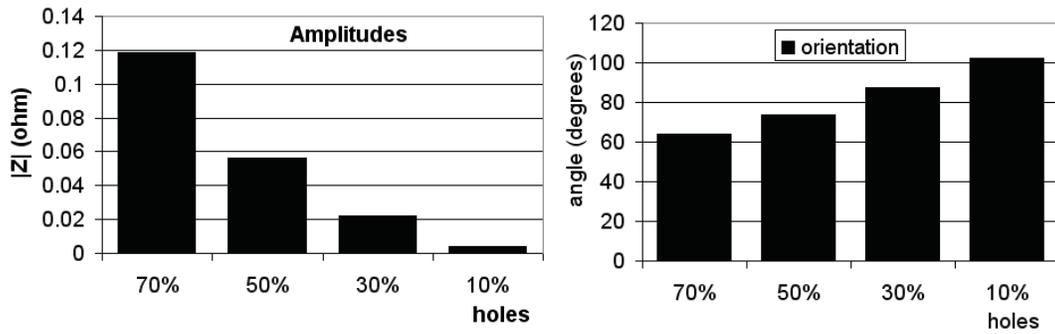


Figure 32: Plots of amplitude and orientation

Measurements also show variation in amplitudes and orientation, but less than the simulations. This is the result of the applied DC field, which increases the depth of penetration. Remarkably the imaginary part of signal increases (except for the through hole), while simulations show an increase in the real part.

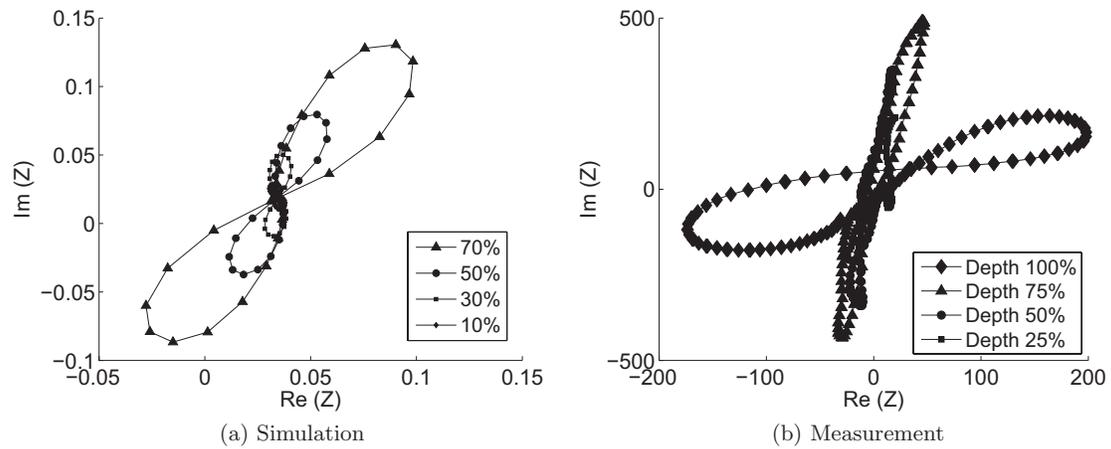


Figure 33: Simulations and measurements of bottom holes

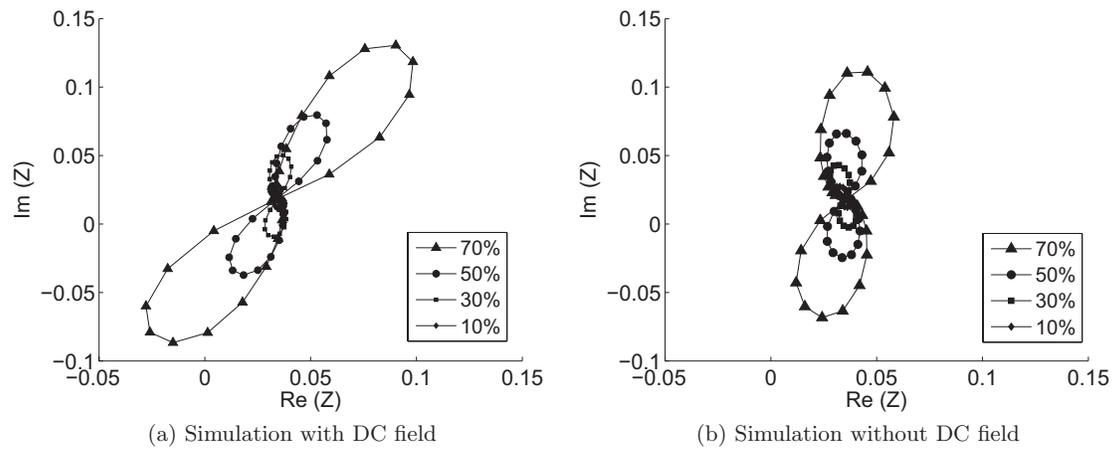


Figure 34: Simulations results of bottom holes

5.3 Conclusions

Simulations have shown that the influence of hole sizes and phase lag are visible in the simulation results. The influence of the DC magnetic field is less clear. The magnetization process has to be improved in order to get better results.

6 Conclusions and recommendations

6.1 Conclusions

Eddy current testing is a powerful method to test steel plates for cracks and corrosion pits. However, the measurement results from such a scanner appear hard to interpret. During this project it has been tried to find a way to improve data abstraction from these measurements. This has been done by creating a model based on a computational technique. The finite difference time domain (FDTD) technique appeared to be most suitable for this project. This method is especially developed for electromagnetic problems and is relatively easy to understand. By solving Maxwells equations in time, wave propagation can be investigated as well as statical problems. A drawback of the method is the relative big computational load that is required.

A model and graphical user interface (GUI) have been built from scratch to get maximum insight in the computational technique and to create a dedicated, easy to use model. A single turn coil have been implemented to compare calculated self-inductances to an approximation formula. This way the influence of the perfectly matched layer (PML) and frequency on the simulation results could be determined. It has also be shown that DC currents in a coil generate correct H-fields for both true dimensions and down-scaled dimensions. Thereafter the model has been extended with a steel plate and artificial holes to compare results with measurements. It appeared that the model calculated realistic results in case of plain AC measurements (without an external DC magnetic field). Implementation of a magnetization process for the steel plate was less successful, although acceptable results were obtained using only linear magnetization. The magnetization process have to be improved before reliable comparisons with measurements can be made.

6.2 Recommendations

Further development of the model is needed before it can be used to improve the eddy current measurement system. Especially the magnetization process needs additional attention to solve current (instability) issues. Furthermore material properties can be made more realistic, like implementation of corrosion and other inhomogeneities. Several frequency dependencies can be introduced too ($\varepsilon(\omega)$, $\mu(\omega)$, etc). At the moment, only the sensor coils and plate are simulated. Other parts that influence the measurements can be added, like the aluminum bracket that holds the sensor.

Besides making the model more realistic, computational efficiency and accuracy can be improved. By implementation of a nonuniform grid or subcells, interesting areas (e.g. the defect area) can be calculated more accurate than less interesting areas (e.g. corners filled with air). The current uniaxial PML can be 'upgraded' to a convolutional PML to handle evanescent waves more effectively. Finally parallelism might be an option to speed up calculations. FDTD is perfectly suitable for parallel calculations [15].

A Software manual

A graphical user interface (GUI) has been created to ease the use of the implemented model. This appendix discusses how the GUI can be used to perform simulations. The GUI has been designed with Microsoft Visual C++ (version 6.0) and can therefore only be used on Windows systems.

A.1 Overview

The GUI has a main menu in the top right corner (figure 35). The options shown in the menu divide the modeling and simulation process into steps that are sorted in a chronological way. Most options can however be accessed at any desired time. The first option (*View model*) gives an overview of the model. Geometry settings can be changed and the model can be viewed in 2D. The second option (*Edit crack*) shows a 'crack editor'. Holes and cracks can be easily created and viewed here. Several tools can be used to move or copy designed defects. The third option (*Magnetize plate*) makes it possible to execute a magnetization process such that a steel plate becomes magnetized to a certain degree. Several magnetization curves can be chosen from a library. The fourth option (*Adapt mesh*) is not implemented yet. This option was originally added to reduce the number of cells by implementing a non-uniform mesh. The last option (*Simulate*) provides the simulation interface. In here specific simulation settings can be done. During simulation several changing quantities can be viewed in graphs.

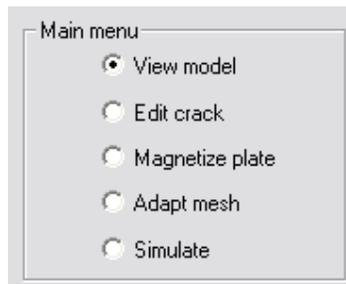


Figure 35: Main menu

A.2 View Model

On start up, the program will go to the *view model* option. Every option is by then disabled, except for the *I/O control* box (see figure 36). The user has to choose either to save a new model or load an existing one. This can be done by entering a path into the editable text field. The path can either be absolute or relative. Entering a relative path only exists of 1 word (e.g. 'my_model') which will be saved (or loaded) as the top directory for the model. See section A.6 for more details about the generated data structure. By hitting [enter] the input will be processed.



Figure 36: I/O control box

After opening a model, specifications can be changed (figure 37). These can be found in the boxes on the left side. The *Cell size* sets the edge length of a single cell. The current program only supports cubic cells, such that all edges are equal in length. The *Dimensions* box contains the total number of cells used in the model. Values entered here must be a multiple of the cell size. *PML* indicates the number of cells added to the model on all sides to absorb field energy. The *Coil offsets* box gives the possibility to add or remove coils. Usually both coils are active because of the difference measurement. The offsets indicate the distance from the origin to the first cell inside the coil (the core). This means the coil windings are not included in this value. *Coil properties* determine the cross sections of the coil cores. Coils are oriented with their main axis in parallel to the 'y'-axis (see figure 18). In this box also the number of turns and layers of wire can be adapted. The last box contains the plate height. This parameter can only be changed in 'z'-direction. A plate has always an 'infinite' surface in the 'xy'-plane.

The image shows a software interface for setting model parameters. It is organized into several distinct sections, each with a title and a set of input fields and checkboxes.

- Cell size:** A text input field containing '2.5' and a dropdown menu set to 'nm'.
- Dimensions:** Four rows of text inputs. The first row is 'x: 375 nm (150 cells)', the second is 'y: 195 nm (78 cells)', the third is 'z: 375 nm (150 cells)', and the fourth is 'PML: 7 cells'.
- Coil offsets:** Two checkboxes, 'Add coil 1' and 'Add coil 2', both of which are checked. Below them are two columns of text inputs for 'x', 'y', and 'z' offsets, each followed by 'nm'. The values are: Coil 1 (x: 50, y: 50, z: 210) and Coil 2 (x: 50, y: 120, z: 210).
- Coil properties:** A section for 'Inner diameter' with two text inputs: 'x: 275 nm' and 'z: 125 nm'. Below these are two more text inputs: 'Turns: 10' and 'Layers: 4 cells'.
- Plate:** A checked checkbox 'Add plate:' followed by a text input '125 nm'.

Figure 37: Model settings

Input in the text fields will be checked when the [enter] button is pressed. All input will then be checked in top-down order. If there are any inconsistencies these will be corrected. For example: if the z-dimension of the model is 10 cells high, the z-offset of a coil cannot be bigger than 10 cells. If this offset is corrected to 10 cells, the coil automatically has a height of 0 and 0 layers (no more space left).

The buttons in the lower left corner have the following functions:

- 'Reset to default': resets default values
- 'Apply changes': applies the settings to the model
- 'Quit': Exit program (global button)
- 'Reset all': has the same effect as a complete restart (global button)

The model can be viewed with the help of the bottom panel, under the drawing area (figure 38). The drawing area shows a 2D view of a selected layer. With the *Plane* box a plane can be selected. A small drawing on the left side of the control panel shows the orientation of the plane. The *Magnify* box enables the user to zoom in to view a quarter of the total model. This is sometimes useful if details have to be viewed. On the right-hand side of the control panel a layer can be selected by typing a number (and pressing [enter]) or by hitting '+' or '-'. If 'magnify' is set to '4x', different parts of the model can be viewed with the 'Part' buttons. Note that the already mentioned orientation drawing shows a highlight of the selected part.

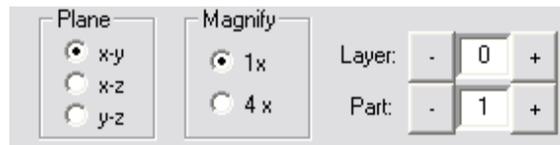


Figure 38: Control panel for viewing
Cells in the drawing area have colors. The meaning of these colors is:

- White: air
- Blue: steel
- Cyan: coil wire (current in negative x-direction)
- Yellow: coil wire (current in negative y-direction)
- Red: coil wire (current in positive x-direction)
- Magenta: coil wire (current in positive y-direction)
- Green: coil wire (current in positive x- and negative y-direction) ¹¹
- Light red: inside of the coil (by default air)

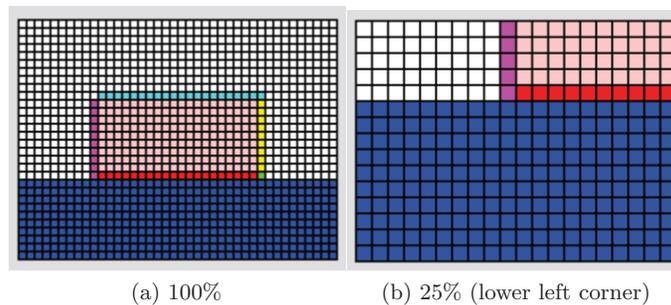


Figure 39: Coil and plate viewed in 'xz'-plane

¹¹This cell solves an issue introduced by the way FDTD calculates fields throughout space

A.3 Edit crack

The *Edit crack* option (see figure 40) allows the user to easily adapt the steel plate. Again, it is possible to load a previously saved crack by simply entering the name in the text field and hitting [enter]. New defects can be created by moving the mouse over the plate shown in the drawing area while the left button is pressed. Cells that turn from blue to green are part of the crack. By moving the mouse over a crack while the right mouse button is pressed crack cells can be erased (green \rightarrow blue). Because most defects are often more or less similar at different layers in the plate, a copy function has been implemented (box *Various functions*). By simply entering the number of the source and destination layer crack layers can be copied. The *Crack information* box shows the maximum dimensions of the crack (that is, the distance between the outermost cells in a certain dimension). The offset is also shown (cell closest to the origin). The '+' and '-' buttons can be used to move the crack cell by cell. Double '+' and '-' move the crack 10 cells. When moving a cell, all layers are moved at the same time. A crack can only be moved if all cells stay within the model after the action (cells cannot be discarded). The 'Blur crack' button causes the crack edge to be blurred. For all cells close to a crack edge new cell properties are set. These properties are a weighted average of the properties of adjacent cells. As a result, transition from steel cells to crack cells becomes smoother. The blur button can be used multiple times, but each time the crack will grow by 2 cells in each direction (the crack size is determined by all non-steel cells). Finally the 'Erase crack' button resets all cells to steel (not shown here).

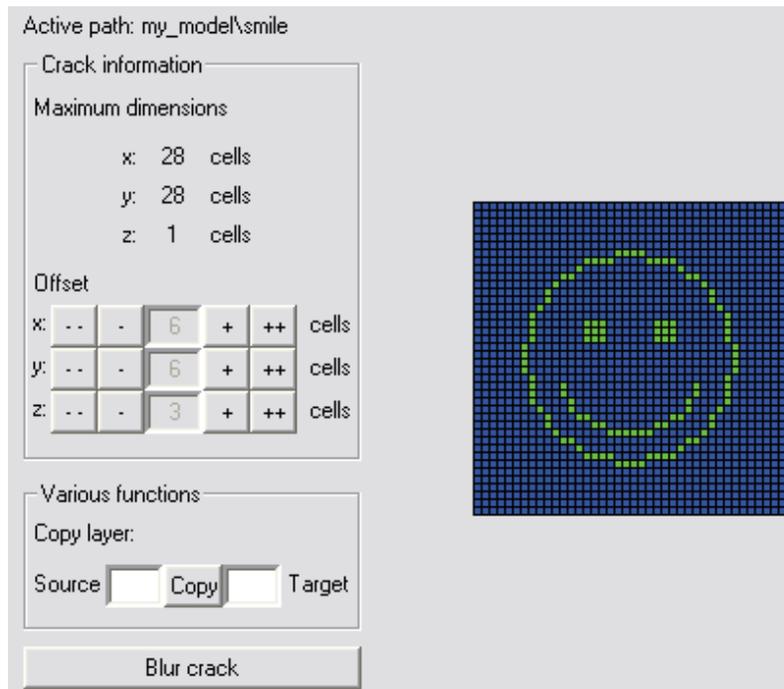


Figure 40: Edit crack

A.4 Magnetize plate

Magnetize plate takes care of magnetization of a single plate with defect. Figure 41 shows part of the interface. By default a steel plate is unmagnetized with a value of $\mu_r = 5000$ assigned to all steel cells. This value can be adapted by entering a different value in the text field. The *Choose magnetization curve* box is initially set to 'no magnetization', which means that simulations will be performed without first magnetizing the plate (this setting is also used in main menu option *Simulate*). If the option 'linear magnetization' is chosen, the plate will be magnetized in a linear manner (μ_r does not change). Other options are BH-curves of several types of steel. These curves are saved as text files in directory 'steel_library'. The user can easily add extra curves by placing additional text files in this directory. If the button 'Magnetize' is pressed, the steel plate with crack is magnetized according to the selection. Note that the radio button in *I/O control* must be set to 'save' and no magnetization directory may exist. It is also possible to load previously obtained magnetization data by selecting 'load' and then hitting the 'Magnetize' button. If it is desired to discard old magnetization data, the directory 'Magnetization' must be removed by hand. During magnetization, the process can be viewed in the drawing area. This area shows the B-field. Red colors represent positive values and blue colors represent negative values. Intensity of the colors represent the strength of the field. By selecting a direction from the *View Magnetization* box the viewed field component can be changed. In the lower right corner, next to the already explained control panel (see section A.2), extreme values are shown that are represented by maximum color intensity. Magnetization vectors can be erased by button 'Erase magnetization' (not shown here). After magnetization several files are automatically saved in the directory 'Magnetization'. This includes B, H and M field components and μ_r throughout space.

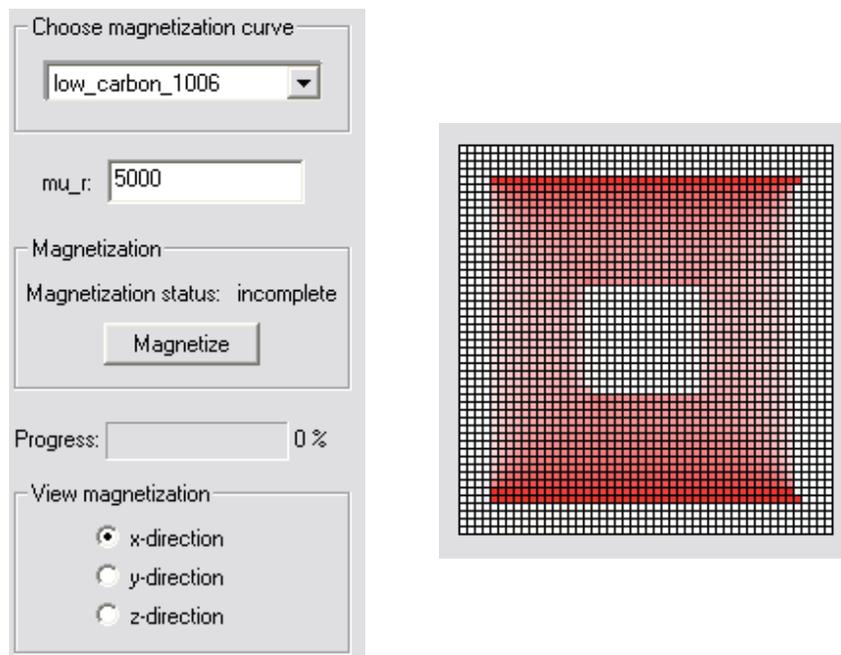


Figure 41: Magnetize plate

A.5 Simulate

When the modeling is done, simulation can be started (figure 42). First of all, several settings have to be made, like current amplitude (peek-peek), frequency and the number of time steps (box *Simulation settings*). The box *Simulation properties* shows immediately several important values that result from the settings ('sim step', 'steps / period' and 'standard depth'). It is important to set the number of simulation steps to a value equal to '*steps/period*' · 1.5 · *n* (with *n* = 1, 2, ...). This is because of the impedance calculation. The Fourier transform is being calculating after half a period (to avoid start up noise). The best results are obtained if complete periods are Fourier transformed. The calculated values of $|Z|$ and $\arg\{Z\}$ are shown in the lower left corner.

The *I/O Control* box has only a 'Load' function. By typing a series name into the bar and hitting [enter] a series of predefined cracks can be simulated automatically. The cracks in the series must have a series name (e.g. 'crack') and a number (starting from 1). The software will search for crack directories containing the series name and numbers (1 and up). As soon as a series number does not exist, the series is ended. By example: if one has predefined cracks 'c1', 'c2', 'c3' and 'c5' and enters 'c' into the text field, the software will simulate the first 3 cracks one after another. A pop-up box tells the user how many cracks have been found. Note that the setting in main menu option *Magnetize* determines if a plate is magnetized before (AC) simulation.

During the simulation process (not the magnetization process), data can be viewed in graphs. The control panel below the graphs give the ability to show different kinds of information (e.g. 'current', 'emf', 'difference emf' etc). If coil dependent information is selected, both coil values are shown in different colors (red and blue). If a 'difference' is selected, 1 color will be zeroed.

Simulations can be 'paused' and 'aborted' when active, but other settings cannot be changed (except data to be viewed). After each simulation, the complex impedance difference (length and argument) is written to a text file named after the series that is being simulated (e.g. in the previous example 'c.sim_results'). These values are equal to those shown in the lower left corner.

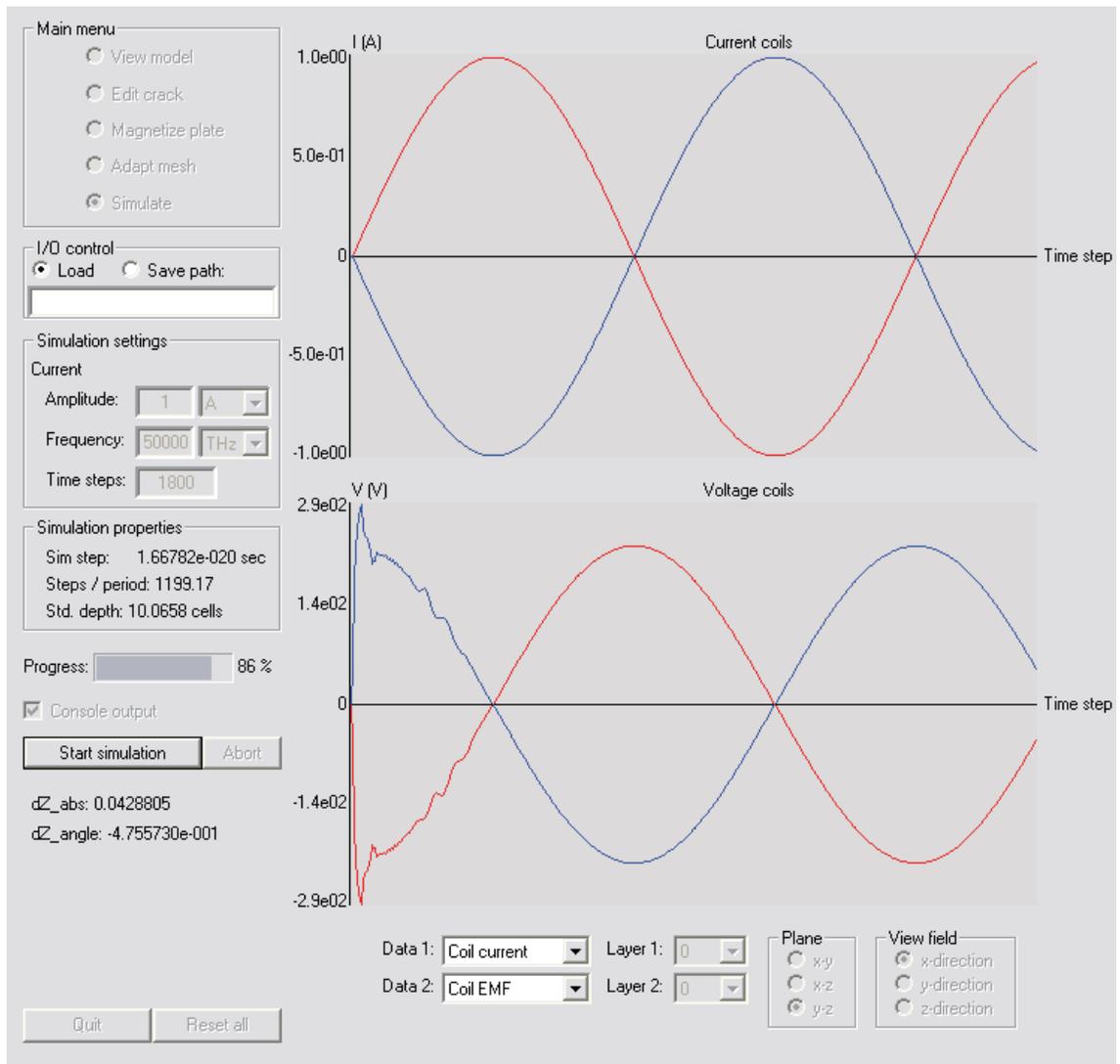


Figure 42: Simulate interface

A.6 Data structure

During the modeling and simulation process, a data structure is generated. Figure 43 shows this structure. The top level is the model name entered after start up (directory 'my_model'). In this directory a file is placed that contains the model settings ('model_settings.txt'). This file is updated each time the 'Apply changes' button is pressed in main menu option *View model*. The next step is to create defects. In figure 43 is shown that directories are created for each new defect ('c1', 'c2', 'c3' and 'c5'). Each directory contains a file 'crack.txt' that contains information about the crack. The file starts with information about sizes and offsets. Thereafter information about material, relative permeability, conductivity, etc. is stored. If the crack (used in a plate) is magnetized, a new subdirectory is created: Magnetization. This directory contains information of the B, H and M fields and μ_r throughout space. Each field component gets its own subdirectory. At this level, text files are stored having names like '0.txt', '1.txt' etc. These numbers stand for the layer ('z'-direction). E.g. a 20 cells thick plate results in 20 text files per subdirectory after magnetization. Each text file contains data of the 'xy'-plane and can be easily plotted in (for example) Excel.

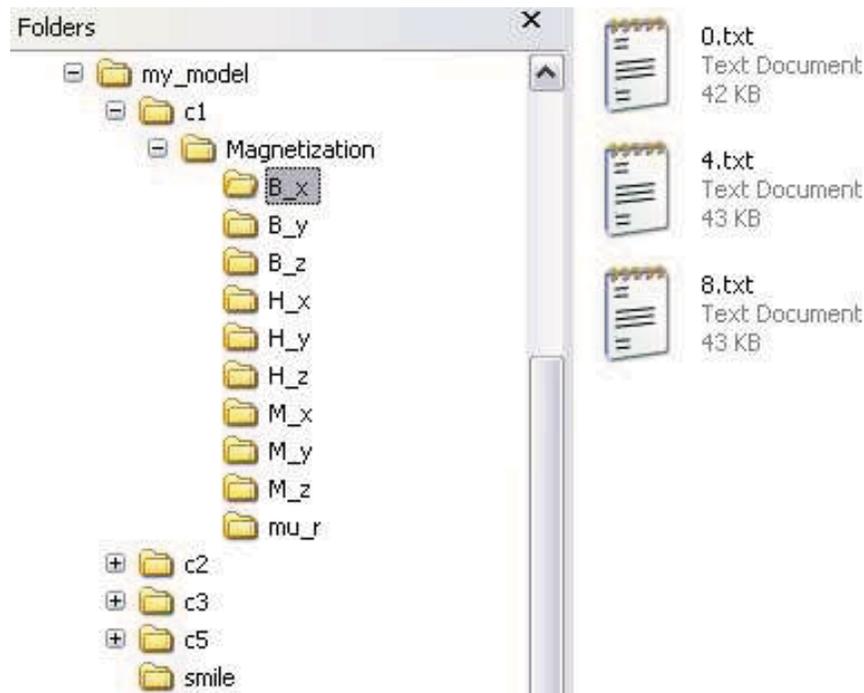


Figure 43: Generated data structure

B Software structure

In this appendix the written software will be discussed. It can be used as a guideline for further improvements. All software is written in C++, and designed in Microsoft Visual C++ 6.0. As a result, the code is not fully compatible with other programming environments or operating systems. However, the graphical user interface (GUI) can be separated from the rest of the source code without much effort after which the code is much more portable.

The software can globally split up in 3 parts as shown in figure 44. Due to the used programming environment, the *GUI* is the main group of classes. This part takes care of the interaction with the user. It uses the *Model* part as resource for displaying all kind of data. The GUI creates an object of the main *Simulator* class by means of 'aggregation'. This has been done to emphasize the independence of both parts. The simulator part contains all classes that somehow modify the properties of the model or calculate field propagation. The *Model* part takes care of the memory management of the model. A pointer to this object is passed to *GUI* via *Simulator*.

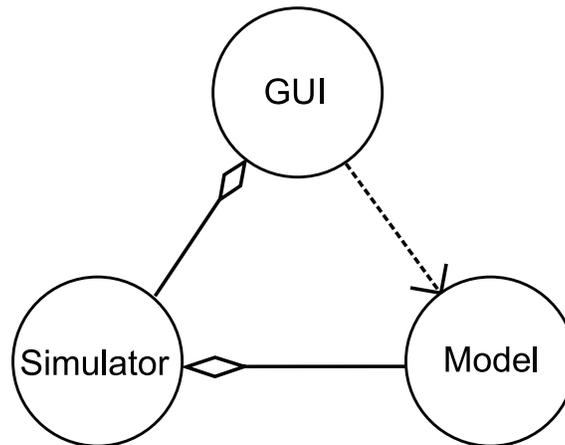


Figure 44: Top level abstraction of software

B.1 GUI classes

Appendix A contains a manual describing the functionality of the GUI. A class diagram of the GUI part is shown in figure 45. The 'heart' of the GUI is shown in the middle ('PE-MEC_MODEL_GUIDlg'). All user interactions are processed here. For each option in the GUI a separate class has been created ('ViewModel', 'EditCrack', 'MagnetizePlate', 'AdaptMesh' and 'SimulateModel') to execute their specific tasks. Since a lot of visualization objects (buttons, text fields, static text, etc.) have to be modified (shown/hidden, enabled/disabled, moved, etc.) for each option, they all have a 'Dlg_props' (dialog properties) class attached to them. This way the representation of these objects is mainly separated from their functionality. The 'CommonControls' class contains the functions of the control panel below the drawing area (see figure 38). It creates an object of the 'Draw_objects' class to actually draw information on screen. This 'Draw_objects' class uses information of the *Model* part mentioned in figure 44. Another

class that uses this functionality is the 'EditCrack' class. As explained in the software manual, the user can draw cracks by clicking and dragging the mouse. To immediately adapt and show the updated model the 'Draw_objects' class is used.

It is shown that all GUI classes connected to the main class 'PEMEC_MODEL_GUIDlg' are defined as 'composition'. This means that they are all destructed when the main class is destructed (they become useless as the main class no longer exists). The situation on the lower side of 'PEMEC_MODEL_GUIDlg' is different. Here a connection with the 'Simulator' part (figure 44) is created by means of 'aggregation'. This indicates that the simulator can still run without the GUI¹². Classes shown in non-solid boxes are not part of the GUI.

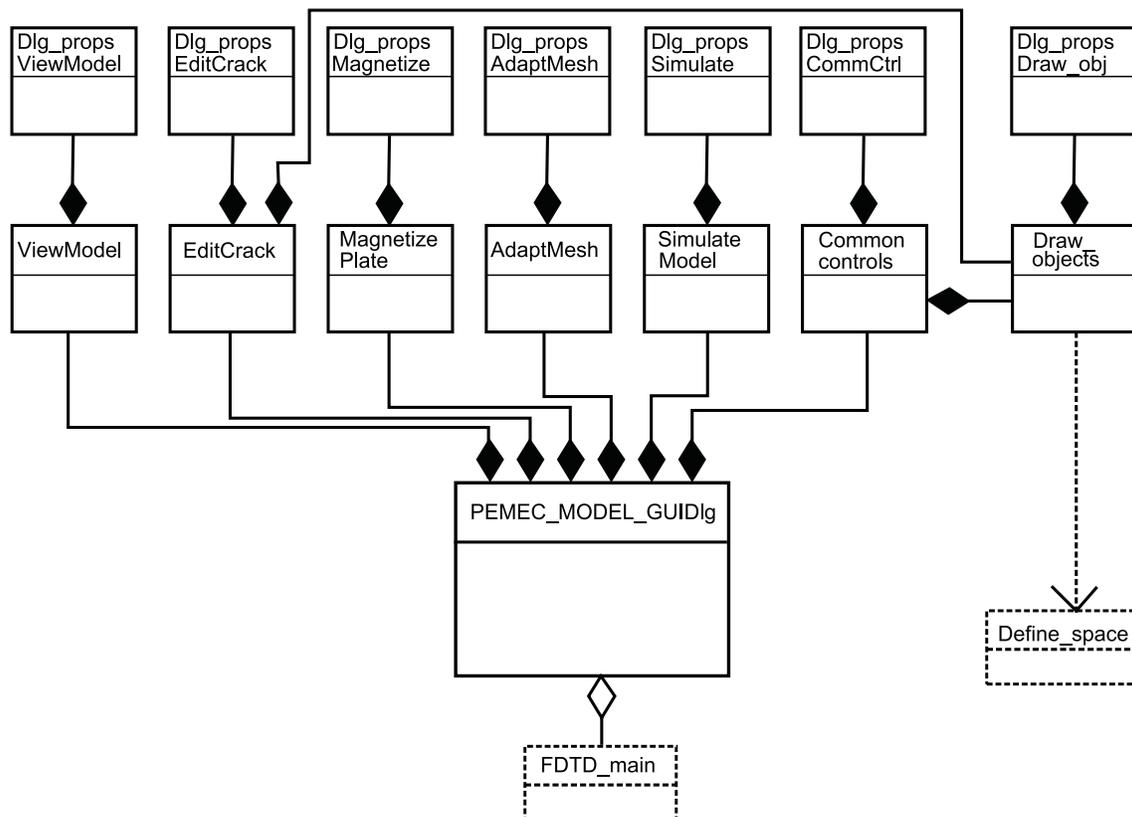


Figure 45: Class diagram of the GUI part

¹²For example: instead of the GUI a text-based interface could be used

B.3 Model classes

The last part is rather simple and consists of only 2 classes (figure 47). The 'FDTD_main' class passes the model specifications to an object of the class 'Define.space'. This class then generates the number of Yee cells needed to build the model. Cells are assigned specific properties (conductivity, permeability, etc) according to the specified material. The class 'Yee_cell' itself contains space for all these properties. A distinction is made between non-PML and PML cells. The latter needs several additional variables to take care of the energy absorption. For every cell needed in the model an object is made of class 'Yee_cell'.

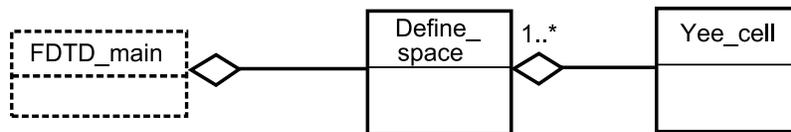


Figure 47: Class diagram of the Model part

References

- [1] J.A. Brando Faria. *Electromagnetic foundations of electrical engineering*. Wiley, 2008.
- [2] NDT-ed. Eddy current testing, 2009. <http://www.ndt-ed.org/EducationResources/CommunityCollege/EddyCurrents/Introduction/IntroductiontoET.htm>.
- [3] J. M. Bluckley. An introduction to eddy current testing theory and technology, 1994. <http://www.joe.buckley.net/papers/eddyc.pdf>.
- [4] technick.net. Inductance calculators, 2009. http://www.technick.net/public/code/cp_dpage.php?aiocp_dp=util_inductance_calculator.
- [5] qsl.net. Inductance formulas, 2009. <http://www.qsl.net/in3otd/rlsim.html>.
- [6] Walton C. Gibson. *The method of moments in electromagnetics*. Chapman and Hall/CRC, 2008.
- [7] Gerard Meunier. *The finite element method for electromagnetic modeling*. Wiley, 2008.
- [8] Allen Taflove and Susan C. Hagness. *Computational electrodynamics : the finite-difference time-domain method*. Artech House, Boston, MA, 2005.
- [9] Kane Yee. Numerical solution of initial boundary value problems involving maxwell's equations in isotropic media. *IEEE Transactions on 14*: 302307, 1966.
- [10] Dennis M. Sullivan. *Electromagnetic simulation using the FDTD method*. IEEE Press, New York, 2000.
- [11] F.E. Terman. *Radio Engineers' Handbook*. London, McGraw-Hill, London, UK, 1950.
- [12] Faezeh Sh.A. Ghasemi, M. S. Abrishamian, and A. MovaFeghi. Analysis of crack detection in metallic and non-metallic surfaces using fdtd method. Technical Report ECNDT 2006 - We.4.3.2, Department of NDT, INRA, Tehran, Iran, K. N. Toosi University of Technology, Tehran, Iran, 2006.
- [13] Richard P. Feynman, Robert B. Leighton, and Matthew Sands. *Lectures on physics*. Addison-Wesley, 1964.
- [14] D.C. Meeker. Finite element method magnetics, 2009. <http://femm.foster-miller.net>.
- [15] Wenhua Yu, Raj Mittra, Su Tao, Yongjun Lui, and Yang Xiaoling. *Parallel finite-difference time-domain method*. Artech house, 2006.