

# Mutual Coupling Modeling of NFC Antennas by Using Open-Source CAD/FEM Tools

U. Muehlmann, M. Gebhart, and M. Wobak

**Abstract**— Near Field Communication (NFC) antennas are not comparable to standard communication antennas used for traditional transmission services. The geometries are complex, narrow conductor cross-sections and long loops located close to materials sensitive to electromagnetic interaction, when integrated, make the application of analytical coupling formulations inapplicable, even when it comes down to non-linear relations forced by ferrite shielding. Commercial finite element (FEM) solvers are powerful, but they do not serve the needs of NFC applications due to complexity, inadequateness and code obscurity. An intensive open source code investigation has identified one alternative solver which is capable to deal with all NFC RFID related coupling effects one can imagine. Automated scripts, the tool-chain, and geometry macros have been developed for rapid prototyping of such. Amongst other open source tools listed, the ElmerFEM solver is the most promising solver for linear and non-linear quasi-static electro-magnetic (EM) NFC problems. The solver has a powerful interface and delivers results close to reality even when computational complexity is a trade of physical resolution and memory capacity of standard commercial workstations.

## I. INTRODUCTION

THE mutual coupling characteristic of NFC devices operating at 13.56 MHz is an important system parameter and defines the operating volume appropriately. The technical challenges of new innovative use cases require a fundamental understanding of power transfer and antenna detuning behavior of such, especially when battery-less communication is required. This is of particular importance because the amended standards [1, 2] offer the possibility of small antenna classes with an effective antenna area linkage of a couple of square centimeters and below. An optimized design of such small antennas for NFC transponders is published in [3].

Furthermore, the integration to mobile phones is continuously going to be more complex, since mobile devices, most notable smart-phones, are getting more convenient for payment or other NFC applications under the assumption of higher functional integration density, innovative look and design, but with no additional costs. This demanded a reassessment of the classical RFID technology and has enabled novel technical concepts such as zero coupling antenna geometries for transmitter and receiver separation, active load modulation [4], and smart antennas with an adaptive matching feature [5]. Especially, the adaptive matching feature requires some sort of system

identification normally exposed to dynamics caused by movement of human interactions. Hence, a finite element analysis (FEM) of the mutual coupling phenomenon will be required to study such effects even when ferrite materials have to be placed close to the operating antenna for magnetic shielding purposes. Furthermore, automated FEM analysis will support a rapid antenna prototyping to some extent, primarily when analytical formulations yield insufficient results and a verification by measurements is not feasible.

In general, commercial FEM tools are capable of doing this job. But, the incredible high licenses costs, the unknown algorithms used, the non-transparent data pre-conditioning behind, the limited scripting support, and the ability of performing more complex task not directly tailored to the present problems calls the usage into question. Good open-source tools are rare and difficult to handle. But, the algorithms used are normally described and easily made visible by simply looking into the source-code.

We have dedicated this article to lift the myth that only commercial tools are the only option, even when a proper understanding of the Maxwell's equations is missing. As a brief overview, section II gives a general introduction of open-source solvers capable to investigate NFC antenna problems operating in the range of 13.56 MHz. Section III is dedicated to the powerful open source *ElmerFEM* multi-physics solver including the proposed simulation setup dealing with geometry definitions, meshing requirements, the solver setup and a possible batch file setup concept. Section IV gives the comparison of the results of the FEM solver with measurements and analytical formulations. Section V concludes with the general findings and further outlook.

## II. OPEN ELECTROMAGNETIC CODE FOR NFC ANTENNAS

### A. Antenna Shape and Geometry Considerations

In general, NFC antennas are not comparable to commonly known communication antennas used in the MHz range and above. The antenna shape has to be capable of transferring both power and data. These characteristics can only be achieved by the principle of an air coupled transformer tuned to the operating frequency, whereas the desired bandwidth for data communication is normally controlled by the antenna quality factor  $Q$ . Best power transfer is given when the coupling factor  $k$

$$k = \frac{L_{12}}{\sqrt{L_1 L_2}} = \frac{L_{21}}{\sqrt{L_1 L_2}} = \frac{M}{\sqrt{L_1 L_2}} \quad (1)$$

converges to unity. According to (1), this is the case when the mutual inductance  $M$  is maximized by close coupling and by  $L_1$  equal  $L_2$  (equal geometries). A detailed study on an efficient antenna design for maximum power transfer is presented in [6].

Furthermore, the well known analytical *Neumann formula*

$$M = \frac{\mu_0 \mu_r N_1 N_2}{4\pi} \oint_{C_1} \oint_{C_2} \frac{ds_1 \cdot ds_2}{|r_1 - r_2|} \quad (2)$$

defines the mutual inductance  $M$  at geometry level, where  $\mu_0$  is the magnetic permeability of vacuum,  $\mu_r$  the relative permeability covering the material properties,  $N_1$  and  $N_2$  the turn counts,  $C_1$  and  $C_2$  the contour integrals of the linked area,  $r_1$  and  $r_2$  are the source and destination point of observation, respectively. According to (2),  $M$  will be maximized through close coupling, meaning large linkage area, close located winding tracks, and short distance.

From a manufacturing perspective, NFC antennas are often designed as vacuum metalized copper or aluminum tracks on nonconductive film carriers (see figure 1), etched or milled printed circuit boards (PCB) with galvanized vias for track cross-bondings.

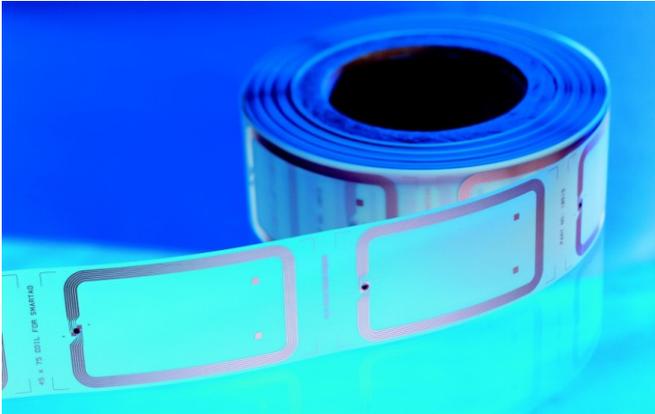


Fig. 1. Typical endless tape of vacuum metalized copper tracks RFID antennas on nonconductive film carriers

### B. Brief Solver Overview and Capabilities

A couple of relevant open source solvers have been studied whether they are applicable to the modeling of the mutual coupling of NFC antennas in general. The first solver investigated is *4NEC2*, a boundary element method (BEM) based 3D full-wave solver for wire networks and plates [7]. This solver is capable to model such problems. But, the given geometries are 50 to 100 times smaller than one wavelength. This is a known problem for BEM based solvers and the computed results may be inaccurate or the solver fails completely to converge. Furthermore, when complex antennas have to be investigated, especially for PCB antennas with cross-bonded vias, *4NEC2* is not applicable. The second open source solver investigated here is the *FEMM* 2D quasi-static FEM based solver [8]. This is a great

tool for 2D and axial-symmetrical problems which is also capable to import AutoCAD dxf-file, and fits well to the specification of NFC antennas. 3D coupled antennas with arbitrary azimuth and elevation orientation to each other are difficult to map to a 2D or axial-symmetrical problem, so the usage is limited to parallel orientations by assuming electrical long cross-sections. The third solver to notice here is the *FastHenry* 3D impedance extraction code for conductors based on the generalized minimal residual algorithm [9, 10]. This code is useful when an exact estimation of the self and mutual inductances of complex geometries are of particular interest. It cannot deal with non-linear materials (e.g. ferrite) when shielded structures have to be investigated and even the proprietary file format makes the transfer of existing geometries more difficult to use. The last open source solver mentioned in this article is the *ElmerFEM* multi-physics, multi-platform (Linux, Windows) FEM software package [11]. This solver is fully applicable to all problems dealing with NFC antenna structures. It comprises a FEM AV-formulation solver [12], electro and magneto quasi-static solvers, and uses direct and indirect, linear and non-linear matrix solvers which can deal with a large number of unknowns.

## III. ELMERFEM SOLVER TOOL CHAIN

### A. Geometry Input Data

Most open source solvers lack a good graphical interface for geometry definition and standard data format import capabilities. It often takes long to understand the input data format, tricks and pitfalls are discovered only through experience. This is different when using the *ElmerFEM* software. It offers a good interface to the *Salome platform*, a pre- and post-processing application for numerical simulations [13].

A long research of data conversion tools for EagleCAD PCB antenna design files to common 3D formats (for instance: STEP, BREP, or IGS) has produced unsatisfactory results. To our knowledge, there exists no open-source utility which is capable doing that, and even tested commercial tools have problems with galvanized vias, resolution of arcs and pads, track thickness, and accuracy of the resulting 3D PCB representation. For proper meshing and high solver convergence probability, we recommend redrawing of any antenna design unless 3D formats are already available. The *Salome platform* provides a python (see [www.python.org](http://www.python.org)) programming interface for template designs. By using this interface, rapid prototyping is ensured, since the antenna parameters, length, width, wire (track) thickness, and track spacing can be varied very easily with a python script file. For vacuum metalized antennas we define the conductor center line track and extrude the cross-section face along it to gain the 3D loop solid. For printed antennas, we have defined macros for the galvanized vias, which are fused with the 3D tracks when necessary. Additional bodies, (carrier

materials, shielding, and air volume) are normally generated as simple solids (plates, boxes, cubes, and cylinders) and Boolean operations are used to cut out the complex wire shapes to gain body-in-body representations. This generates the required common interfaces for subsequent meshing and solving. See a particular example of an ID1 PCB NFC antenna in figure 2.

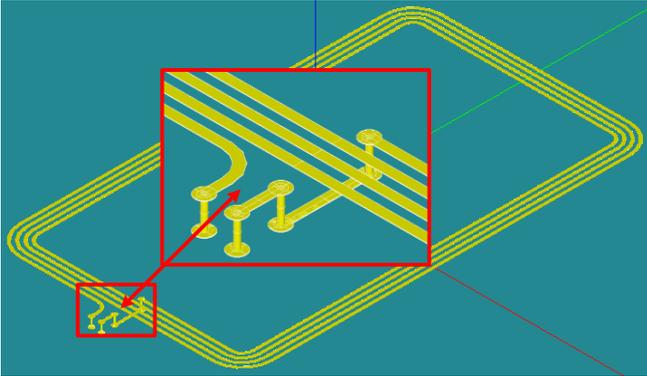


Fig. 2. 3D geometry representation of an ID1 PCB NFC antenna designed with Salome platform showing pads, vias and wire tracks

### B. Meshing and Data Conversion

Meshing the geometries sounds simple but that's not true. The mesh defines the computational complexity the solver has to deal with. A very fine mesh will result in more detailed results. In fact the solver may run out of memory. By comparison, a very coarse mesh may influence the convergence behavior dramatically. We have studied different cases and can conclude that the total number of 3D volume elements should be less than one million in order to be computed on a standard commercial 64Bit workstation with 8GByte RAM. On laptops with RAM sizes lower than 4GByte, we recommend numbers between 250k and 500k elements. It has to be noted that the surrounding air volume around the antennas has to be meshed as well. An improper selection of the air cube (or sphere) dimension will probably explode the total number of 3D volume elements. By fitting to these constraints, we selected the embedded Netgen 1D-2D-3D mesh generator setting growth rate to 0.5, number of segments per edge to 1, and number of segments per radius to 2. All other settings are used by default. Unfortunately, 2<sup>nd</sup> order elements are not supported by the *ElmerFEM* solver, so the meshing is composed of first order tetrahedrons, only. In order to preserve the body-to-body interfaces for the solver, one has to partition and group the geometries and its corresponding sub-meshes. This procedure is well explained in the solver user guide. One may force individual conditions to special boundary surface of a body, in our case the antenna terminal voltage. Therefore, boundary faces have to be created. By using the universal mesh format (UNV) for the data transport to the *ElmerFEM* mesh input format (mesh.\*), the enclosed *elmergrid* utility has to be used. The UNV format is capable of holding individual body information. But, it cannot preserve boundary faces when converted that way. For that purpose, the *ElmerGUI divide*

*surface by sharp edges* feature has to be used to gain back the relevant boundaries. This is a major drawback when batch processing (parameter studies) is of particular interest. In such cases we recommend to install the computer aided engineering Linux distribution (*CAELinux*). Within this distribution one will find some useful python scripts (*unv2gmsh.py*, *unv2ElmerUNV.py*) translating named boundaries and sub-shape groups into the *ElmerFEM* input format.

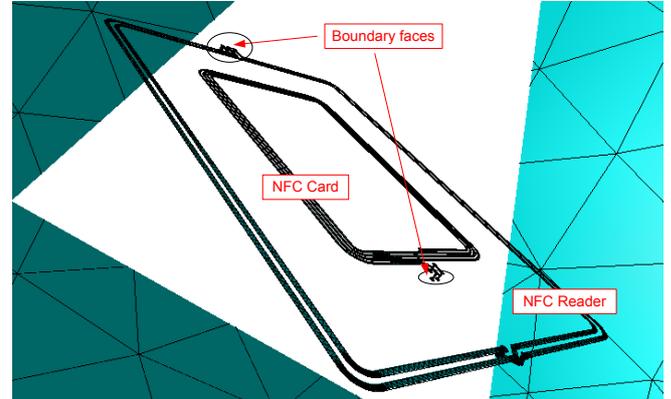


Fig. 3. 3D mesh representation of an ID1 to ID2 PCB NFC antenna coupling simulation inside air cube

As an example of use, figure 3 shows the final meshing results and the applied boundary surfaces of an ID1 to ID2 PCB NFC antenna coupling inside the air cube ready for solving.

### C. Solver Setup and Computation

The solver requires a special configuration file, the so call solver input file (SIF-file). The general simulation setup, the target models, solver and sub-solver configurations, the body forces, the boundary conditions, and material properties have to be added to this file.

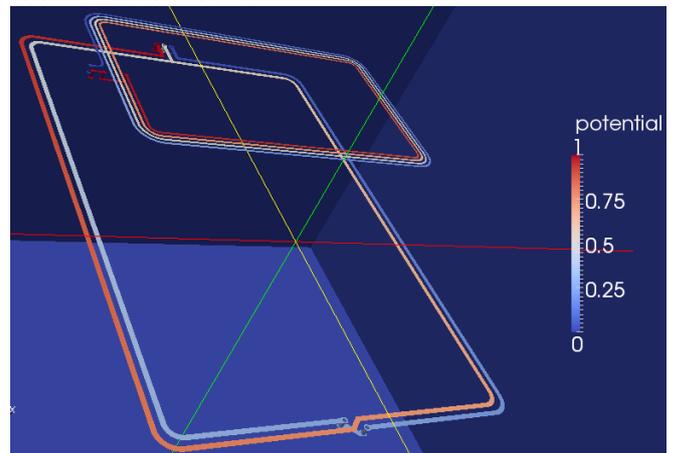


Fig. 4. 3D electrical potential distribution (voltage) along the wire tracks of the ID1 NFC reader and ID1 NFC card at 2 cm displacement, parallel arranged and 90° rotated

We use three individual solvers for the mutual coupling computation in total. At first stage, the current density of the wire tracks has to be computed. For that purpose, we use the

*StatCurrentSolve* sub-solver procedure. It solves for the electrostatic potential in conductive media and derives the volume current used for the subsequent magnetic field solver as boundary condition input.

The terminal voltages ( $U_1$  and  $U_2$ ) are forced on the boundary faces according to figure 3. The given geometries require some matrix preconditioning so that the solver converges properly. The available direct solver method LAPACK and Umfpack are not applicable because they require more RAM than is available on the desired target workstations. Hence, we have tried the available Krylov-based iterative methods, called Conjugate Gradient (CG), the Biconjugate Gradient Stabilized (BiCGStab), and the optionally available (in *CAElinux*) HYPRE for solving large, sparse linear systems of equations. By proper preconditioning (ILU4 or higher), the applied algorithms converge within 300 iterations. As a side product, the *StatCurrentSolve* procedure calculates the effective ohmic resistance and the antenna current when *Calculate Loads* is enabled. These numbers will be required finally to compute the air coupled transformer equation, beforehand. As an example see the 3D electrical potential distribution (voltage) along the wire tracks of the ID1 NFC reader and ID1 NFC card in figure 4. The open-source *ParaView* post-processing application is used for result visualization.

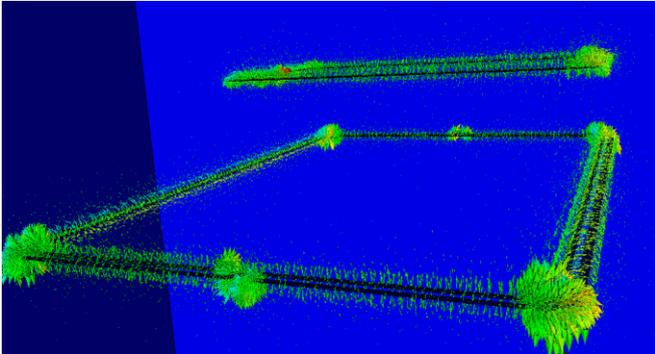


Fig. 5. 3D representation of the magnetic flux density vector field along the wire tracks of the ID1 NFC reader and ID1 NFC card at 2 cm displacement, parallel arranged and 90° rotated

The second and third solvers make use of the *MagnetoDynamics* procedure with the *WhitneyAVSolver* and *MagnetoDynamicsCalcFields* sub-solver procedures. The vector potential formulation (or AV formulation) is used to solve the Maxwell equations. The solver is applicable to static or quasi-static problems, using lowest order Whitney 1-forms (edge elements). The settings for the convergence behavior of the *WhitneyAVSolver* are sensitive. It took a long time to find appropriate settings in order that the solver converges within reasonable time. We have used the CG method without any preconditioning with a convergence tolerance set to 1.0e-6. Dependent on the geometrical relations (antenna separation, and orientation) the algorithm converged within 600 and 1300 iterations. The *MagnetoDynamicsCalcFields* sub-solver acts as a post-processor of the *WhitneyAVSolver* and requires no additional

iterations.

Aside from the 3D vector potential and the 3D magnetic flux density, the solver calculates the total magnetic field energy within the defined volume. See the 3D representation of the magnetic flux density vector field along the wire tracks of the ID1 NFC reader and ID1 NFC card illustrated in figure 5 by using the integrated *ElmerGUI VTK postprocessor*.

#### D. Self and Mutual Inductance Calculation

The self and mutual inductance is calculated with the well known relation [14]:

$$W = L_1 \frac{I_1^2}{2} + MI_1 I_2 + L_2 \frac{I_2^2}{2}. \quad (3)$$

Where  $W$  is the total magnetic field energy,  $L_1$  the inductance of the NFC reader,  $L_2$  the inductance of the NFC card antenna, respectively.  $M$  is the mutual inductance and  $I_1$  and  $I_2$  represent the electric currents through the wire tracks of the individual antennas. In order to solve for (3), three individual simulations have to be performed in sequence. We use  $U_1=U_2=1V$ ,  $U_1=1;U_2=0$ , and  $U_1=-U_2=1V$ . The *ElmerSolver* SIF file allows running simulations in sequence where previous calculations are used for iteration initialization of the current run. This will lead to a fast convergence of successive simulations, if needed.

#### E. Batch Processing Setup

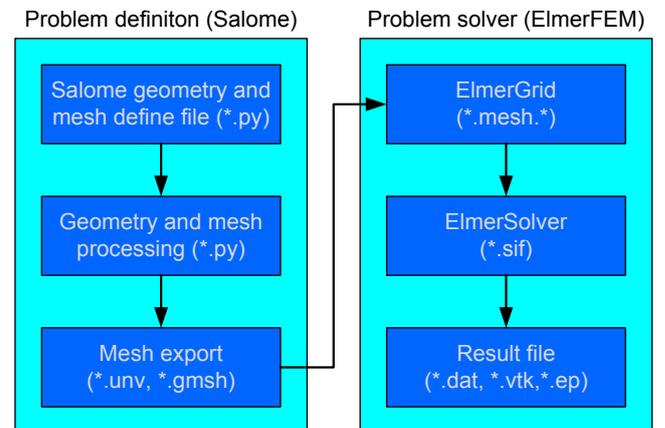


Fig. 6. Batch file processing steps to use *Salome* and *ElmerFEM* as the processing engines

Batch Processing is feasible with the tools described here in principal. The batch setup file should include the proposed steps listed in figure 6. All the geometry definitions, the meshing algorithms, and all auxiliary settings have to be defined for *Salome* by a python script. The *Salome platform* can be started in batch mode (without GUI interface) with the desired python script attached. After 3D geometry rendering and mesh processing, the resulting UNV meshing data file has to be converted to a *Gmsh* [15] formatted file (unv2gms.py) in order to preserve the boundary numbers which are required to be clearly related to the desired boundary definitions of the solvers. After running the *ElmerSolver* console, the individual result files can be used

for further computation or post-processing. By closing the loop with a modification of the Salome geometry script file, the next cycle can be computed.

#### IV. VERIFICATION OF THE APPLIED METHOD

The verification of the applied method is performed by means of measurement results and analytical calculation using (2). The measurement setup consists of an ID2 NFC reader antenna ( $L_1$ ) and an ID1 reference PICC antenna ( $L_2$ ). By positioning the two antennas to appropriate distances and by the application of a terminal AC voltage on  $L_1$ , the coupling factor is determined. For that purpose, both terminal voltages are measured and recorded for all distances. The self inductances  $L_1$  and  $L_2$  are measured by means of a vector network analyzer (NVA). Finally, the coupling factor is calculated with:

$$k = \frac{U_2}{U_1} \sqrt{\frac{L_1}{L_2}}. \quad (4)$$

$U_1$  and  $U_2$  represent the terminal voltages of  $L_1$  and  $L_2$ , respectively. The analytical method is based on (2). A detailed formulation of this method can be found in [16]. The results of all three methods are illustrated in figure 7. It turned out that all three methods produce comparable results. The slight variations can be explained that the feeding lines of the antennas are not modeled in the FEM analysis, nor considered in the analytical calculation.

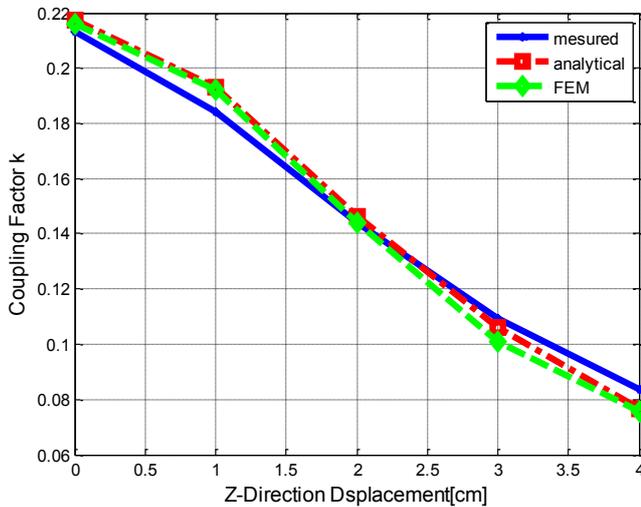


Fig. 7. Measurement results of the coupling factor  $k$  of an ID2 NFC reader and ID1 NFC card versus displacement in Z-direction from 0 to 4 cm.

#### V. CONCLUSION

We have presented an alternative way to commercial FEM based magnetic field solvers for studying NFC related antenna coupling behaviors. The cross-platform (Linux, Windows) open-source packages *Salome* and *ElmerFEM* are identified to be the most promising combination. *Salome* is used for 3D geometry definition and proper meshing, whereas *ElmerFEM* is used to solve for the total magnetic energy, the current density, electrical potential, and antenna current in 3D. The self and mutual inductances are calculated

by post-processing based on the total magnetic field energy relation. A full simulation including post-processing finishes within 12 min. By comparison to measurements and analytical calculation, the present approach is a valid and applicable to more complex scenarios as presented here. Future work will include non-linear investigations and the study of different magnetization effects.

#### REFERENCES

- [1] NFC Forum, *Consortium Homepage*. 401 Edgewater Place, Suite 600, Wakefield, MA 01880, USA: <http://www.nfc-forum.org>, 2012, (online; accessed March 13th, 2012).
- [2] ISO/IEC 14443-2:2010(E), *Identification cards – Contactless integrated circuit cards – Proximity cards – Part 2: Radio frequency power and signal interface*, ISO, Geneva, Switzerland, 2010.
- [3] M. Gebhart and R. Szoncsó, “Optimizing design of smaller antennas for proximity transponders”, in *Proceedings of the 2nd International Workshop on Near Field Communication (NFC '10)*, Monaco, April 2010, pp. 77–82.
- [4] K. Finkenzeller, “Battery powered tags for ISO/IEC 14443, actively emulating load modulation”, in *Proceedings of RFID SysTech 2011; 7th European Workshop on Smart Objects: Systems, Technologies and Applications*, 2011, pp. 1–8.
- [5] M. Roland, H. Witschnig, E. Merlin, and C. Saminger, “Automatic impedance matching for 13.56 MHz NFC antennas”, in *6th International Symposium on Communication Systems, Networks and Digital Signal Processing, CNSDSP 2008*, 2008, pp. 288–291.
- [6] C. Reinhold, P. Scholtz, W. John, and U. Hilleringmann, “Efficient Antenna Design of Inductive Coupled RFID Systems with High Power Demand”, in *Journal of Communications*, Vol. 2, No 6, Nov. 2007.
- [7] A. Voors, *4nec2a, NEC based antenna modeler and optimizer*, Available: <http://home.ict.nl/~arivoors> (online; accessed June 15th, 2012).
- [8] FEMM, *Finite Element Method Magnetics*, Available: <http://www.femm.info/wiki/HomePage> (online; accessed June 15th, 2012).
- [9] M. Kamon, M.J. Tsuk, and J.K. White, “FASTHENRY: a multipole-accelerated 3-D inductance extraction program”, in *IEEE Transactions on Microwave Theory and Techniques*, 1995, pp. 1750–1758.
- [10] FastHenry, *M.I.T. source code distribution*, Available: [http://www.rle.mit.edu/cpg/research\\_codes.htm](http://www.rle.mit.edu/cpg/research_codes.htm) (online; accessed June 15th, 2012).
- [11] ElmerFEM, *Open Source Finite Element Software for Multiphysical Problems*, Available: <http://www.csc.fi/english/pages/elmer> (online; accessed June 15th, 2012).
- [12] O. Biro, and K. Preis, “On the use of the magnetic vector potential in the finite element analysis of three-dimensional eddy currents”, in *IEEE Transactions on Magnetics*, 1989, pp. 3145–3159.
- [13] Salome, *Generic platform for Pre- and Post-Processing for numerical simulation*, Available: <http://www.salome-platform.org> (online; accessed June 15th, 2012).
- [14] O. Biro, “Computation of the flux linkage of windings from magnetic scalar potential finite element solutions,” in *IEEE Transactions on Proc.-Sci. Meas. Technol.*, vol. 149, no. 5, Sep 2002.
- [15] Gmsh, *a three-dimensional finite element mesh generator with built-in pre- and post-processing facilities*, Available: <http://geuz.org/gmsh/> (online; accessed June 15th, 2012).
- [16] C. L. W. Sonntag, E. A. Lomonova, and J. L. Duarte, “Implementation of the Neumann Formula for Calculating the Mutual Inductance between Planar PCB Inductors”, in *Proceedings of the 2008 International Conference on Electrical Machines*, 2008, pp. 1–6.