

Open-Source Platform for Fast Prototyping of Planar Magnetic Components for Space Applications

Alberto Vital
*Centro de Electrónica Industrial
Universidad Politécnica de Madrid*
Madrid, Spain
alberto.vital@alumnos.upm.es

Alberto Delgado^{id}
*Centro de Electrónica Industrial
Universidad Politécnica de Madrid*
Madrid, Spain
a.delgado@upm.es

Jesús A. Oliver^{id}
European Space Agency
Noordwijk, Netherlands
jesus.oliver@esa.int

Abstract—Optimizing magnetic components is key for developing robust and efficient power electronics converters for space applications. While analytical calculations can help with preliminary designs, more advanced tools such as Finite Element Analysis (FEA) tools are necessary. This work introduces an open-source python-based platform designed for fast prototyping of magnetic components using planar conductors. The tool enables users to analyze the electromagnetic parameters of planar transformers or inductors based on finite element results using the open-source software FEMM®, without requiring expertise in FEA. It models magnetic cores and windings to perform 2D axisymmetric simulations, speeding up the analysis process, and generates PCB designs for the designed and simulated magnetic components.

Index Terms—Finite Element Analysis (FEA), magnetic components, open-source, PCB generation, planar conductors

I. INTRODUCTION

Design of magnetic components is a crucial task when developing power electronics. This is even more important when working on space applications, where this kind of elements need to be as efficient, reliable, replicable, power-dense and light-weight as possible.

Many Finite Elements Analysis (FEA) tools are available on the market but either the cost of the license is high for ordinary users or using them is very complex for non-expert ones. Some well-known examples are Ansys Maxwell [1] and COMSOL Multiphysics [2].

Open software alternatives are also various, some of them with scopes that go far beyond magnetic components, such as Elmer FEM [3], a multiphysics software suite, and others directly focused on solving the Maxwell Equations on magnetic components, like Finite Element Method Magnetics (FEMM®) [4], a suite of programs for solving low frequency electromagnetic problems on two-dimensional planar and axisymmetric domains. However, either the ease of use is a problem for users, or limitations, such as axisymmetry or just supporting 2D simulations, require advanced knowledge or intermediate tools that provide a more user-friendly experience.

Many FEM (Finite Element Modeling) simulation possibilities are also implemented in libraries written in c++, like Sparselizard [5], Python, like scikit [6] and SfePy [7], or MATLAB, like the one proposed in [8], but, again, the user requires knowledge on programming in these languages and the learning curve many times does not compensate for the results that are expected to be found.

In an attempt to simplify the process of designing magnetics, some companies have introduced tools that act as interfaces between users and databases and algorithms. The former provide design specifications and the latter search for the best solution for the proposed design. Renowned names are Open Magnetics [9], Auto Planar Magnetics [10] and Frenetic [11], the first two in the form of an online tool, which many times is preferred by the user. However, they normally simply access internal databases, sometimes based on real simulations, but do not study the specific posed problem.

This platform has been developed under this context. It is an open-source python-based tool designed for fast prototyping of planar magnetics by non-expert users. It simplifies the design, simulation, validation, and PCB file generation processes, making it accessible without requiring deep technical knowledge from the user and assessing the real design that the user is proposing.

Finally, this platform is conceived to be used in space applications, although the designs and results are valid at any field. The choice of planar magnetic components is not casual: these type of designs, compared to other traditional winding technologies, have great advantages, such as higher power density, repeatability or predictable parasitics [12], just to mention some. These benefits are specially valuable in the space industry, where reliable and cost-efficient components are key to ensure missions success.

II. PLATFORM GENERAL OVERVIEW

This project consists in a platform that allows designing, simulating, validating, and generating the files to manufacture magnetic components. It uses additional software, such as

FEMM® and KiCad, to launch the simulations and generate the PCB files based on the information introduced by the software in the platform.

Three main stages compose the platform workflow, represented in Fig. 1. The first step is to introduce both the core configuration and winding configuration of the design. Geometry and materials are specified at this point.

Then, in the second stage, the design is validated. The user configures and launches simulations from which relevant information, such as impedance matrices, is obtained. Additional tests can be launched to evaluate the magnetic component performance.

Finally, in the third stage, the user specifies additional parameters related to PCB design rules and implementation preferences to generate a PCB file with the winding design.

III. DESIGN CONFIGURATION

The design configuration section is the first step in the workflow. Here, the user sets the core and the winding configuration. Both configurations define the magnetic component to appropriately launch the corresponding simulations at the validation stage. During the whole process, the user can check the interactive view to see how the design changes dynamically, so that errors (mainly geometric ones) can be fixed before jumping into the validation and synthesis stages.

A. Core Configuration

Configuring the core is the first step in the design configuration. This tab receives two kind of inputs: geometric inputs and magnetic material inputs. With respect to the

geometry inputs, two options can be chosen: either select one of the commercial cores or design a fully custom core. An intermediate possibility is added if some dimensions of a commercial core are manually modified by the user.

Currently, the platform supports four different magnetic core types: RM, PT, PH and 2D axisymmetric (POT core). These geometries are represented in Fig. 2. Although they all have differences, the four of them can be characterized just using seven common dimensions, which can be easily found in the manufacturers datasheets. This eases the use of the platform by non-expert users. Some commercial TDK [17] and Ferroxcube [18] cores are already preloaded on the platform. In future releases, other cores and manufacturers may be included.

A legend showing the meaning of each of the dimensions can be displayed by clicking the corresponding bottom. Finally, the gap can also be defined in this part.

The simulations, that will be launched in FEMM®, will assume that the magnetic core is axisymmetric. However, this assumption is not valid for RM, PT and PH cores. References [19] and [20] suggests some methods to counteract this effect.

Commercial magnetic cores tend to have the same –or very similar– internal and external areas (blue and red areas in Fig. 3). Maintaining the original dimensions and simulating assuming axisymmetry increases significantly the external area. In the particular case of Fig. 3, which deals with a RM8 core, the external area assuming axisymmetry is 2.5 times greater than the real area.

Geometric modifications to the core have been applied in order to compensate for this effect. The inner area (blue) will always be a disc, and its value can be calculated applying (1). C corresponds to the window inner diameter and C' corresponds to the middle cylindrical hole diameter of the core, as can be seen in Fig. 2.

$$A_{blue} = \pi \cdot \left(\frac{C^2}{4} - \frac{C'^2}{4} \right) \quad (1)$$

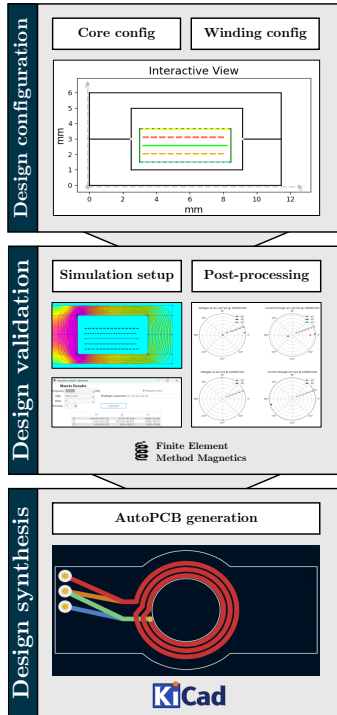


Fig. 1: Platform workflow.

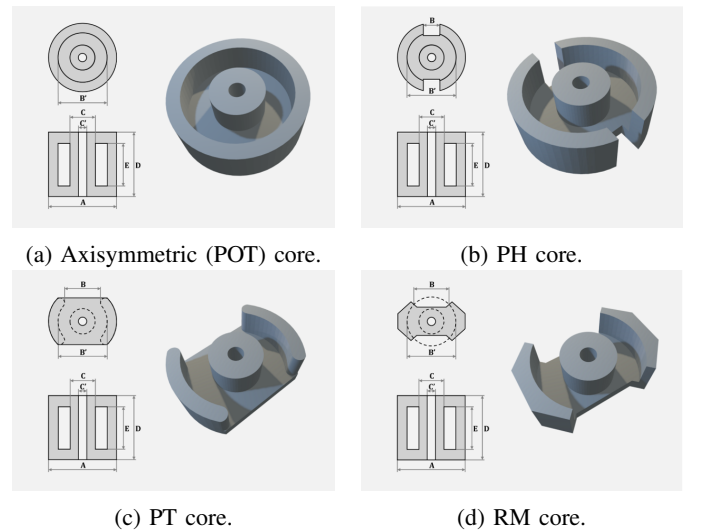
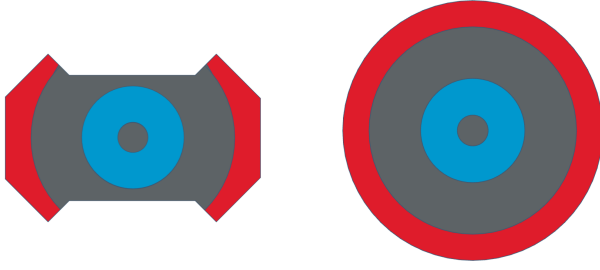


Fig. 2: Legends for the different available core geometries.



(a) Real RM core geometry. (b) Axisymmetric core geometry.

Fig. 3: Internal (blue) and external (red) areas for real (3a) and axisymmetric (3b) core geometries.

The outer area (red), under the assumption of axisymmetry, is also a disc and can be computed with (2). A and B are core geometric dimensions, as can be seen in Fig. 2.

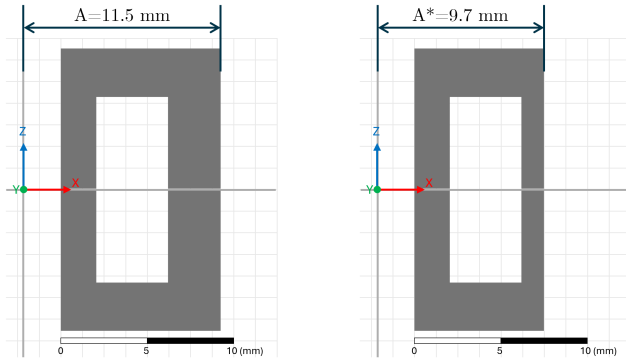
$$A_{red,axisymmetric} = \pi \cdot \left(\frac{A^2}{4} - \frac{B'^2}{4} \right) \quad (2)$$

Equalizing both (1) and (2) and clearing for A, (3) is obtained. A new outer diameter, A_{new} , is found to equalize inner and outer areas under axisymmetric assumption.

$$A_{new} = \sqrt{C^2 - C'^2 + B'^2} \quad (3)$$

Geometry variation for areas equalization could have been done by modifying any of the radial parameters of the core, but the A parameter was chosen because the modification of other parameters (in particular B' and C) could lead to intersection issues between the PCB and the core. An example of A variation in a core section is shown in Fig. 4.

In the case of the winding, it complies with the axisymmetric assumption as the turns will be formed by spirals. Therefore, it is not necessary to apply additional modifications or adjustments to minimize its effects.



(a) Original RM8 core geometry. (b) Modified core geometry.

Fig. 4: Geometry modification in a RM8 core.

Comparing the results obtained using Ansys for the real 3D core representation with the 2D axisymmetric assumption with geometric corrections reduces the overall absolute error by about 10%. Additional adjustments may be implemented in future releases to get even more accurate simulations.

With respect to magnetic material inputs, again two options can be chosen: either select a commercial material or design a fully custom one, and the possibility to mix both options. The parameters required for the material characterization are permeability and conductivity. Some common materials are preloaded on the platform. An overview of this tab can be seen in Fig. 5.

B. Winding Configuration

Introducing the winding configuration is the second step in the design configuration. Here, the user introduces both geometric and material inputs. The geometric inputs describe the characteristics of the planar winding: position with respect to the core, tolerances and winding characteristics (number of layers, number of turns, and width and height of the copper tracks among others). A legend with each of the parameters can be displayed by clicking on the legend button.

The conductive material must also be defined. The required parameters for the conductive material characterization are permeability and conductivity, and, as in the case of the core materials, some common materials are already preloaded in the platform. An overview of this tab can be seen in Fig. 6.

IV. DESIGN VALIDATION

After defining the core and winding in the design configuration stage, the user can validate the design according to the specifications that the magnetic component should comply with. To do so, some simulations are launched first. Then, the information can be managed in the Post-Processing Tab and some additional experiments can be performed.

A. Simulation setup

The Simulation Setup window, which can be seen in Fig. 7, allows the user to set up and launch the simulations.

- Sweep type: linear or logarithmic distribution of the frequencies that are going to be simulated.

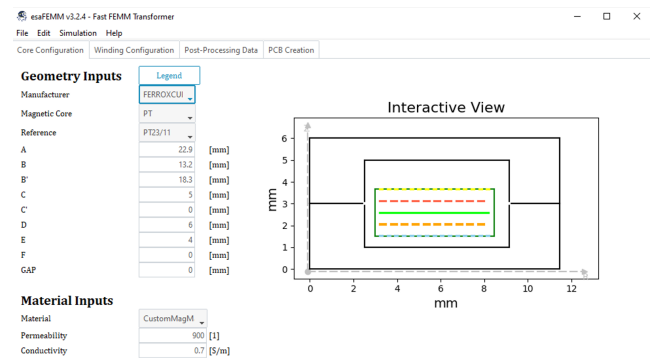


Fig. 5: Core Configuration tab overview.

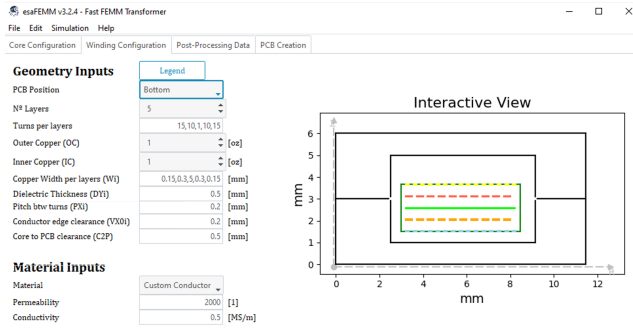


Fig. 6: Winding Configuration tab overview.

- Minimum frequency: lowest frequency of the interval.
- Nominal frequency: nominal operating frequency of the component.
- Maximum frequency: highest frequency of the interval.
- Number of points: number of frequency values that will be simulated (simulation points).

The frequencies are calculated based on the indicated parameters. The nominal frequency is always simulated, even though it does not belong to the interval generated with the type of sweep, maximum/minimum frequencies, and number of points. The simulation configuration can be saved in an Excel file or imported using the same file format. After setting up the simulation configuration, they can be launched directly in FEMM® by clicking on the button.

Unlike other FEA solvers, such as Ansys Maxwell, FEMM® does not compute impedance matrices in a direct way. For this reason, the simulations launched in FEMM® must be designed in a way that this matrix (and other similar ones) can be computed from the data that FEMM® obtains and saves. The way to do this is to run a simulation for each of the layers at each frequency point. Therefore, for n frequencies and m copper layers, $m \times n$ simulations will be launched.

B. Post-processing: simulation files opening

After simulation, FEMM® files can be opened directly from the interface. The user can choose to open the results obtained for all frequencies or just for a single selected frequency.

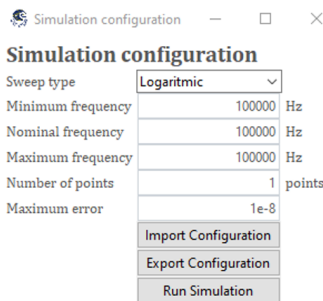


Fig. 7: Simulation Setup window.

C. Post-processing: impedance matrix laboratory

The impedance matrix laboratory allows the user to manipulate the impedance matrix of the design for any of the simulated frequencies. Inductance and coupling coefficient matrices are also available in the laboratory. Also, it is possible to indicate any equivalent connection –series and/or parallel– between the existing windings to compute the equivalent impedance matrix. It can be seen in Fig. 8.

D. Post-processing: experimental setup

The last functionality on the post-processing tab is the experimental setup. It is a virtual test laboratory in which the designer can evaluate the design response at a certain frequency –among the simulated ones– specifying the equivalent connections between the windings and the transformer setup: open-circuit, short-circuit or with an certain output resistive load. The resulting variables (voltage and current through each of the equivalent windings) are represented graphically in polar graphs or saved as an excel file. An example of the first option is represented in Fig. 9.

To indicate the equivalent connection between the windings, the user must enter the connection manually using the platform notation. Four symbols are enough to completely define an equivalent winding.

- Comma [,]: separation between two windings.
- Plus [+]: series connection.
- Asterisk [*]: parallel connection.
- Parentheses [()]: group windings.

A practical example with the winding in Fig. 10 is briefly explained to better understand the notation.

Windings w_1 and w_4 are connected in series. This is expressed as $w_1 + w_4$. Then, they are connected in parallel with w_2 , equivalent to writing $(w_1 + w_4) * w_2$. Finally, w_3 is connected in series, so the final equivalent winding can be expressed as $(w_1 + w_4) * w_2 + w_3$. The notation follows the classical hierarchy order of operations in mathematics: parentheses, multiplication, and addition. There are alternative expressions that can define the same winding connection. Some examples:

- Proposed expression: $(w_1 + w_4) * w_2 + w_3$
- Redundant parentheses: $((w_1 + w_4) * w_2) + w_3$
- Reordered windings: $w_3 + (w_2 * (w_1 + w_4))$

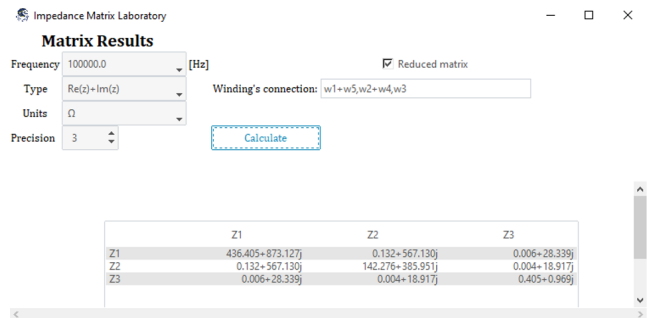


Fig. 8: Impedance Matrix Laboratory window.

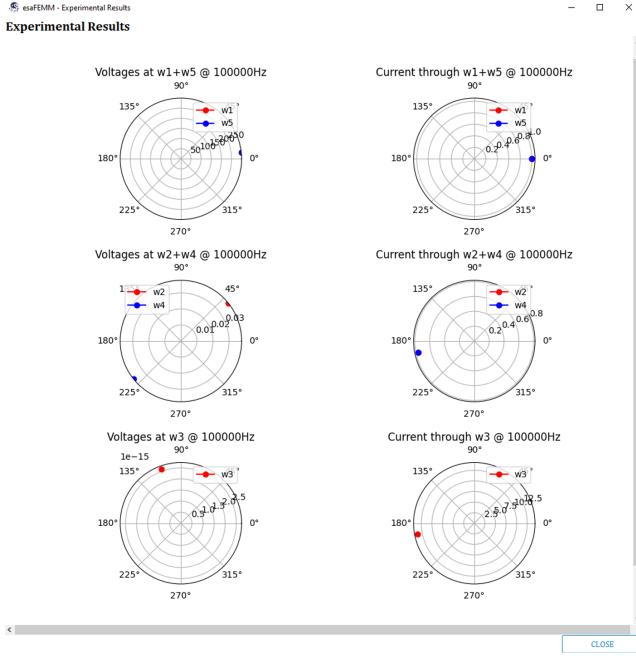


Fig. 9: Experimental results example.

- Another possibility: $(w_2 * (w_1 + w_4)) + w_3$

If two or more independent windings are defined, they will be separated by commas (e.g. $w_1 + w_4, w_2 * w_3, w_5$). The windings do not need to appear ordered (e.g., w_4 can appear before w_2 , as shown in the random example) and a winding can not be repeated in different independent windings. The terms in the impedance matrix will appear in the same order as the one written in the expression (e.g., for the same random example, index 1 in the impedance matrix will correspond to $w_1 + w_4$, index 2 will correspond to $w_2 * w_3$ and index 3 will correspond to w_5).

Now, in order to compute equivalent impedance matrices and perform experiments, a tree-like approach has been implemented. The winding expression is taken as input to build the tree. N trees are built, being N is the number of independent winding groups.

To rebuild the original expression of the winding connection, it must be done from bottom to top, as shown in Fig. 11.

V. DESIGN SYNTHESIS

The last step in the platform workflow consists of the automatic Printed Circuit Board (PCB) design, or AutoPCB

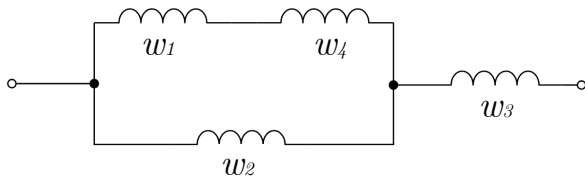


Fig. 10: Winding example.

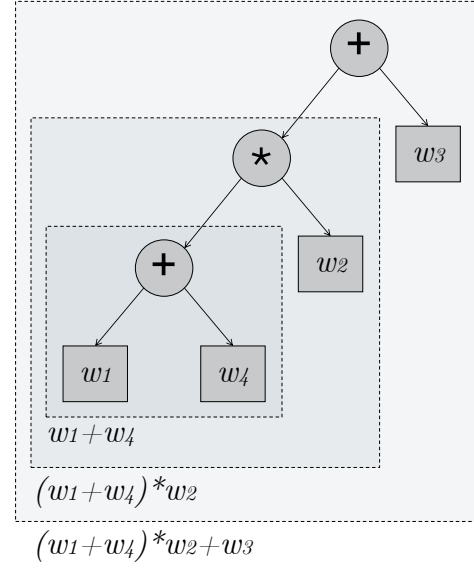


Fig. 11: Tree corresponding to $(w_1 + w_4) * w_2 + w_3$.

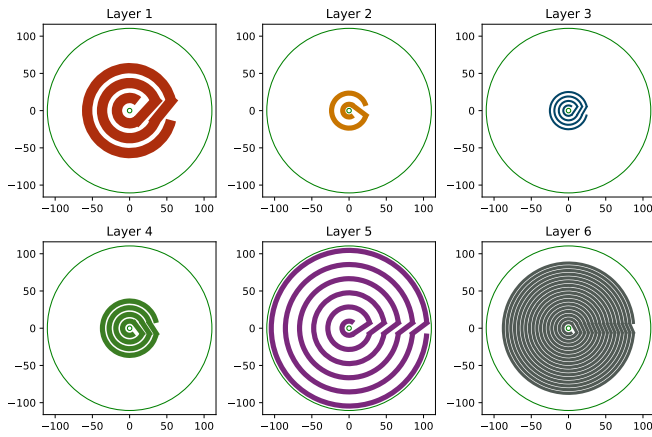
tool. Many different possibilities appear when integrating planar magnetics on PCBs regarding geometry, as explained in [21]. Although this last stage is still being developed, two different approaches for winding generation have been followed, both shown in [22]. The first option is to draw concentric arcs and join their initial and end points. Although the number of points to completely define this geometry is low (in KiCad three points are enough to define an arc [23]), when the ratio between the copper width and the pitch between turns is low, sharp angles appear, as can be seen in Fig. 12a. The second approach consists of drawing real spirals, as in Fig. 12b, which removes the sharp angle issue but requires many more points (even orders of magnitude higher depending on the desired resolution for each turn). Also, the end radius of the real spirals tends to be slightly bigger than in the previous approach, resulting in wider boards.

Different cases are represented in Fig. 12 to compare between both approaches and see that the arc-based approach may be valid for some specific cases, but the implementation is very poor for others. Also, resolution is important when working with the real spiral approach. Low resolution effects can be seen in the layer 5 spiral (purple) in Fig. 12b.

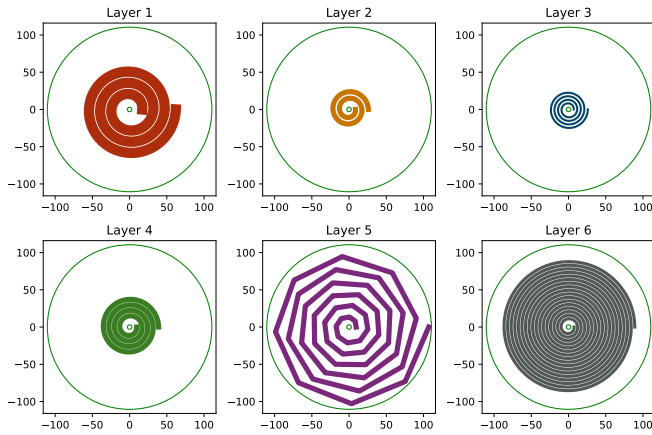
Although it is not yet implemented, the connections between windings are planned to be made using buried vias to avoid interfering with other layers.

VI. CONCLUSIONS

The presented open source tool aims to facilitate the design and prototyping of magnetic components based on planar conductors. To this end, the platform consists of an interface that allows the geometrical parameters of the core and winding to be entered in a very simple way, as well as the selection of materials for each geometry. An extensive set of validation resources has been implemented, including simulations,



(a) Windings formed with arcs.



(b) Windings formed with real spirals.

Fig. 12: Two different approaches for winding generation.

an impedance matrix visualizer and a virtual laboratory for launching experiments. Finally, in future releases, the platform will include a tool for generating the PCB design of the designed coil or transformer. All this has been implemented using open source software.

ACKNOWLEDGMENT

This project is being developed under a programme of, and funded by, the European Space Agency.

REFERENCES

- [1] Ansys Maxwell [Software]. Ansys. Available: <https://www.ansys.com/products/electronics/ansys-maxwell>
- [2] COMSOL Multiphysics [Software]. COMSOL. Available: <https://www.comsol.com/>
- [3] Elmer FEM (version 9.0), [Software]. CSC – IT Center For Science LTD. Available: <https://www.csc.fi/web/elmer/>
- [4] David Meeker, “Finite Element Method Magnetics. User Manual (Version 4.2),” May 16, 2020. [Software], Available: <https://www.femm.info/wiki/Documentation/>
- [5] A. Halbach, “Sparselizard: The user friendly finite element c++ library,” January 17, 2023, Available: <https://www.sparselizard.org/DOCUMENTATION.pdf>
- [6] T. Gustafsson and G. D. Mcbain, “scikit-fem: A Python package for finite element assembly,” *Journal of Open Source Software*, 2020, 5 (52), pp.2369, doi: 10.21105/joss.02369
- [7] R. Cimrman, V. Lukeš and E.Rohan, “Multiscale finite element calculations in Python using SfePy,” *Advances in Computational Mathematics* 45, 1897-1921, 2019, doi: <https://doi.org/10.1007/s10444-019-09666-0>
- [8] A. Lehtikainen, T. Davidsson, A. Arkkio and A. Belahcen, “A High-Performance Open-Source Finite Element Analysis Library for Magnetics in MATLAB,” 2018 XIII International Conference on Electrical Machines (ICEM), Alexandroupoli, Greece, 2018, pp. 486-492, doi: 10.1109/ICELMACH.2018.8507235
- [9] Open Magnetics. [Online tool]. Available: <https://openmagnetics.com/>
- [10] Auto Planar Magnetics. [Online tool]. Available: <https://autoplanar.com/>
- [11] Frenetics. Available: <https://www.frenetic.ai>
- [12] Ouyang, Z., Andersen, M. A. (2013). “Overview of planar magnetic technology—Fundamental properties”. *IEEE transactions on Power Electronics*, 29(9), 4888-4900.
- [13] N. Foerster, T. Piepenbrock, P. Rehlaender, O. Wallscheid, F. Schafmeister and J. Boecker, “An Open-Source FEM Magnetics Toolbox for Power Electronic Magnetic Components,” *PCIM Europe 2022; International Exhibition and Conference for Power Electronics, Intelligent Motion, Renewable Energy and Energy Management*, Nuremberg, Germany, 2022, pp. 1-10, doi: 10.30420/565822103.
- [14] N. Förster et al., “An Open-Source FEM Magnetic Toolbox for Calculating Electric and Thermal Behavior of Power Electronic Magnetic Components,” 2022 24th European Conference on Power Electronics and Applications (EPE’22 ECCE Europe), Hanover, Germany, 2022, pp. P.1-P.9.
- [15] Hung-Ming Chen, Yu-Chin Lin, “Web-FEM: An internet-based finite-element analysis framework with 3D graphics and parallel computing environment,” *Advances in Engineering Software*, Volume 39, Issue 1, 2008, Pages 55-68, ISSN 0965-9978, <https://doi.org/10.1016/j.advengsoft.2006.12.001>.
- [16] H. Li, S. R. Lee, M. Luo, C. R. Sullivan, Y. Chen and M. Chen, “MagNet: A Machine Learning Framework for Magnetic Core Loss Modeling,” 2020 IEEE 21st Workshop on Control and Modeling for Power Electronics (COMPEL), Aalborg, Denmark, 2020, pp. 1-8, doi: 10.1109/COMPEL49091.2020.9265869
- [17] TDK catalogue, “Ferrites and Accessories. EPKOS Data Book,” 2013, [online]. Available: <https://www.tdk-electronics.tdk.com/download/519704/069c210d0363d7b4682d9ff22-c2ba503/ferrites-and-accessories-db-130501.pdf>
- [18] Ferroxcube catalogue, “Soft Ferrites and Accessories. Data Handbook,” 2013, [online]. Available: <https://www.ferroxcube.com/en-global/download/download/11>
- [19] R. Prieto, L. Ostergaard, J. A. Cobos and J. Uceda, “Axisymmetric modeling of 3D magnetic components,” *APEC '99. Fourteenth Annual Applied Power Electronics Conference and Exposition. 1999 Conference Proceedings (Cat. No.99CH36285)*, Dallas, TX, USA, 1999, pp. 213-219 vol.1, doi: 10.1109/APEC.1999.749512.
- [20] R. Prieto, J. A. Oliver and J. A. Cobos, “Study of Non-axisymmetric Magnetic Components by means of 2D FEA Solvers,” 2005 IEEE 36th Power Electronics Specialists Conference, Dresden, Germany, 2005, pp. 1074-1079, doi: 10.1109/PESC.2005.1581762.
- [21] S. S. Mohan, M. del Mar Hershenson, S. P. Boyd and T. H. Lee, “Simple accurate expressions for planar spiral inductances,” in *IEEE Journal of Solid-State Circuits*, vol. 34, no. 10, pp. 1419-1424, Oct. 1999, doi: 10.1109/4.792620
- [22] Y. Pascal, “Modelling of a Planar Magnetic Component for PCB-Integration”, KTH, School of Electrical Engineering, Stockholm, Sweden, 2016. Available: <https://www.diva-portal.org/smash/get/diva2:931335/FULLTEXT01.pdf>
- [23] KiCad “S-Expression Format,” 2024, [online]. Available: <https://dev-docs.kicad.org/en/file-formats/sexpr-intro/index.html>