

Interactive and educational simulation tool for permanent magnet synchronous machines

Interactive and educational simulation tool

Alexander Schugardt, Louis Kaiser, Fatih Avcilar and Uwe Schäfer
Department of Electrical Drives Group, Technische Universität Berlin, Berlin, Germany

691

Received 2 November 2023
Revised 26 January 2024
22 April 2024
Accepted 22 April 2024

Abstract

Purpose – This paper aims to present an interactive design and simulation tool for permanent magnet synchronous machines based on the finite-element-method. The tool is intended for education and research on electrical machines.

Design/methodology/approach – A coupling between the software MATLAB and finite element method magnetics is used. Several functionalities are included as modular scripts and represented in the form of a graphical user interface. Included are fully parametrized motor models, automatic winding generations and the evaluation of torque waveforms, core losses and speed-torque-diagrams. A survey was conducted to determine how the motivation of students concerning the covered topics is influenced by using the tool.

Findings – Due to its simplicity and the intuitive visualization of the results, the tool provides direct access to the topic of electrical machines without having to deal with separate scripts. The modular structure of the software allows simple extensions with new functions. Because students can directly contribute to the tool with their own work, their motivation for using and extending it increases.

Originality/value – The presented tool offers more functionalities compared to similar free software packages, e.g. the calculation of core losses and speed-torque diagrams. Also, it is designed in such a way that it can be easily understood and extended by students.

Keywords Permanent magnet synchronous machine, Graphical user interface, Winding design, Simulation of electrical machines, Educational software

Paper type Research paper

1. Introduction

Nowadays, the design and optimization of permanent magnet synchronous machines (PMSM) is done mainly with software based on the finite-element-method (FEM). One of these is finite element method magnetics (FEMM) (Meeker, 2019), which is a free and open-source alternative to commercially available programs but lacks a comprehensive user interface and parametrized machine models. The connection to Octave, MATLAB or Python through predefined Lua script commands offers a wide range of possibilities for the design of electrical machines. The presented simulation tool is based on MATLAB functions that are coupled with FEMM and structured regarding their corresponding tasks. Similar



© Alexander Schugardt, Louis Kaiser, Fatih Avcilar and Uwe Schäfer. Published by Emerald Publishing Limited. This article is published under the Creative Commons Attribution (CC BY 4.0) license. Anyone may reproduce, distribute, translate and create derivative works of this article (for both commercial and non-commercial purposes), subject to full attribution to the original publication and authors. The full terms of this license may be seen at <http://creativecommons.org/licences/by/4.0/legalcode>

COMPEL - The international journal for computation and mathematics in electrical and electronic engineering
Vol. 43 No. 3, 2024
pp. 691-703
Emerald Publishing Limited
0332-1649
DOI 10.1108/COMPEL-11-2023-0554

programs (Bonneeel *et al.*, 2018; de Andrade *et al.*, 2019; Kuptsov *et al.*, 2019) and libraries (Lehikoinen *et al.*, 2018) have been created before, but they either do not offer as many options as the presented tool or are not as accessible for beginners. The machine geometry is fully parametrized using MATLAB scripts. Furthermore, the winding table is automatically generated, and a series of magnetostatic simulations can be executed to create (discrete) time-dependent solutions. All results are evaluated and displayed.

A graphical user interface (GUI) based on a MATLAB application was created to support and simplify the usage of the scripts. First, this allows easier access to the topic and introduces students to the benefits of field simulations. The calculation methods for obtaining all needed quantities are introduced as part of lectures. Subsequently, the modular structure of the scripts allows a wide range of extensions, for example, in the context of student projects. The learning objective is to acquire knowledge about the calculation of e.g. counter-electromotive force (back-EMF), torque and core losses based on the vector potentials and flux densities from the FEM solution.

2. Simulation tool

To simulate a PMSM using the interactive GUI, the user must specify the geometry parameters. Furthermore, information about materials (conductors, iron core and magnets), current waveform and speed is required. The geometry can be plotted any time inside the GUI. Thus, changes in geometry parameters can be seen instantly.

2.1 Geometry and materials

The stator geometry is fully parametrized. It is possible to change the stator type from parallel slots to parallel teeth. This is useful for the simulation of rectangular conductors (e.g. in hairpin windings). Furthermore, details of the slot or tooth head geometry can be changed (e.g. the radius of an arc or the form of the tooth head).

The rotor geometry is also fully parametrized. Currently, the following topologies can be used: surface-mounted magnets, interior magnets with rectangular shapes (as shown in Figure 1) and V-shaped magnets. Switched reluctance machines and synchronous reluctance machines are included as well. It is also possible to add more parametrized rotor topologies as independent MATLAB functions.

In the next step, the materials can be selected. These include conductor material, magnet material and electrical steel. The unmodified library already offers a wide range of materials. Custom materials can be added in a comfortable and intuitive way inside the GUI. A spline interpolation is used for nonlinear $B(H)$ characteristics of e.g. electrical steel. To avoid deviations from the given data, enough points for especially low and high flux densities must be specified.

2.2 Winding and current

The winding design is automatically performed based on the user input for the number of slots N , pole pair number p and winding layers by application of a star of slots. The symmetry conditions for m phases and $t = \text{gcd}(N, p)$ (greatest common divisor) are as follows:

$$\text{one - layer - winding: } \frac{N}{t \cdot m} \in \mathbb{N} \text{ and } \frac{N}{2m} \in \mathbb{N} \quad (1)$$

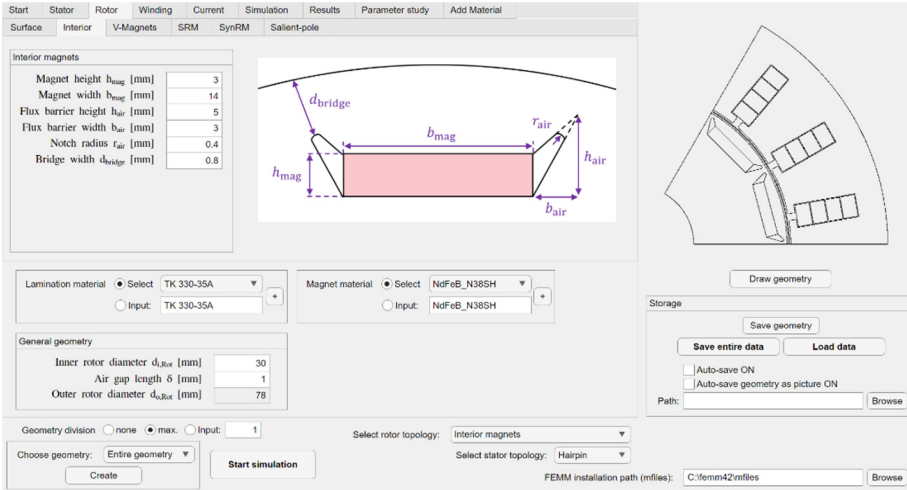


Figure 1. Rotor parametrization with interior magnets and hairpin winding

Source: Figure created by authors

$$\text{two-layer-winding: } \frac{N}{t \cdot m} \in \mathbb{N} \text{ and } \frac{N}{m} \in \mathbb{N} \quad (2)$$

The positions of the positive and negative phases inside the slots are determined based on the phasor diagram of the slot voltages. To demonstrate the concept, a two-layer-winding with $N = 9$, $p = 4$ and $m = 3$ is going to be analyzed. The phasors are drawn with a phase shift of $\Delta\varphi = 360^\circ s^2 p / N = 160^\circ$ to one another (left-hand-side of [Figure 2](#)). Starting with slot 1, the positive phases U+ (current out of the plane) are put into all slots over an angle of $360^\circ / (2m) = 60^\circ$. Then, the next positive phase V+ starts after $360^\circ / m = 120^\circ$. The negative phases are put into the opposite slots of the phasor diagram. The result is shown in [Figure 2](#) on the right ([Pyrhönen et al., 2013](#)).

Based on this phasor diagram, a generated winding table shows the position of all positive and negative phases with their corresponding slot numbers (see [Figure 3](#) at the bottom). Furthermore, the number of slots per pole per phase q and the winding factors are calculated using the following formula ([Müller et al., 2008](#)):

$$\text{slots per pole per phase: } q = \frac{N}{2 \cdot p \cdot m} = \frac{q_Z}{q_N} \quad (3)$$

$$\text{distribution factor: } \xi_{Z,k} = \frac{\sin\left(k \cdot \frac{\pi}{2m}\right)}{q_Z \cdot \sin\left(k \cdot \frac{\pi}{2m \cdot q_Z}\right)} \quad (4)$$

$$\text{chording factor: } \xi_{S,k} = \sin\left(k \cdot \frac{\pi}{2} \cdot \frac{s}{\tau_p}\right) \quad (5)$$

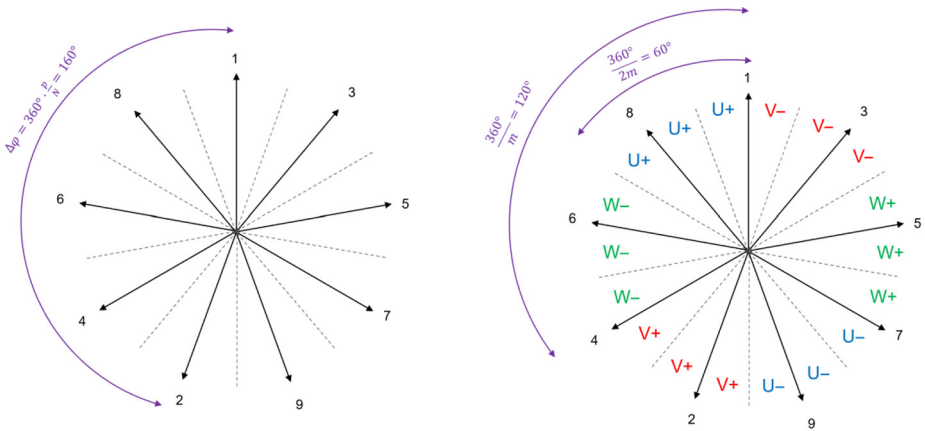


Figure 2. Phasor diagram of slot voltages with positive and negative phases

Source: Figure created by authors

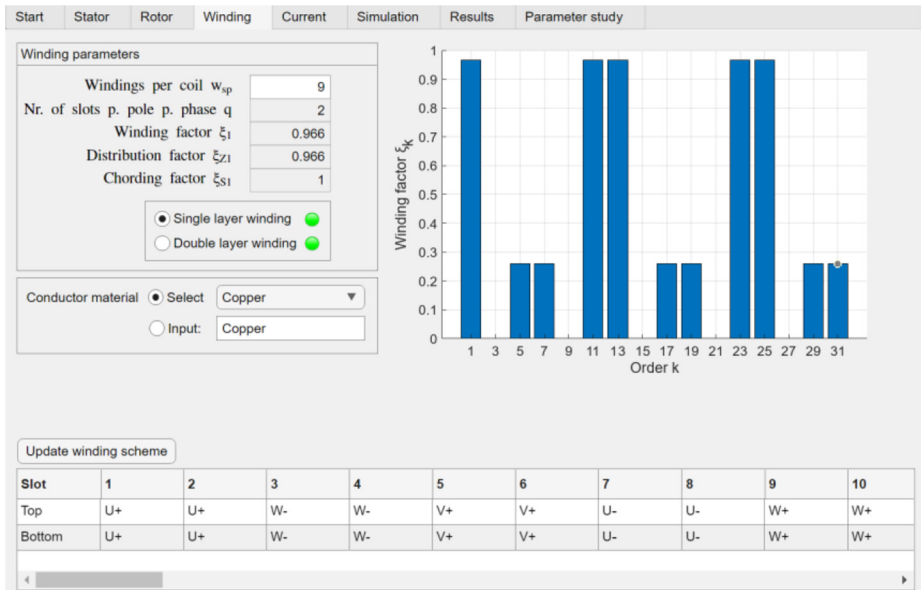


Figure 3. Winding tab of the GUI (data from Toyota Prius 2004)

Source: Figure created by authors

The winding factor is calculated by $\xi_k = \xi_{Z,k} s^2 \xi_{S,k}$ for the spatial harmonic k . The pole pitch equals $\tau_p = \pi \cdot \frac{d_{i,Sta}}{2p}$ using the inner diameter of the stator $d_{i,Sta}$. The skewing s is derived from the winding table. It is the distance between a positive phase and its negative counterpart as an arc length. The winding factor is plotted as a function of the harmonic orders (see Figure 3). Also, it is automatically displayed whether one- or two-layer windings are possible (Burruss *et al.*, 2008; Hsu *et al.*, 2004).

2.3 Meshing and field simulation

For the field simulation, a partial differential equation (PDE) is needed. At first, Maxwell's equations and the material law (Pyrhönen *et al.*, 2013) are considered for the magnetostatic case (time $t = \text{const.}$ and the electrical field is neglected):

$$\text{rot } \vec{H} = \vec{J} \quad (6)$$

$$\text{div } \vec{B} = 0 \quad (7)$$

$$\vec{B} = \mu \cdot \vec{H} \quad (8)$$

\vec{H} is the magnetic field strength, \vec{B} the flux density, \vec{J} the current density and μ the permeability. The magnetic vector potential \vec{A} is defined as $\vec{B} = \text{rot } \vec{A}$. The derivation of the PDE is based on the given equations, and the Coulomb gauge $\text{div } \vec{A} = 0$ is considered (Müller *et al.*, 2008):

$$\text{rot } \vec{H} = \text{rot } \frac{\vec{B}}{\mu} = \frac{1}{\mu} \text{rot rot } \vec{A} = \frac{1}{\mu} \cdot \left(\underbrace{\text{grad div } \vec{A}}_{=0} - \underbrace{\text{div grad } \vec{A}}_{\Delta} \right) = \vec{J} \quad (9)$$

$$\leftrightarrow \frac{1}{\mu} \cdot \Delta \vec{A} = -\vec{J} \quad (10)$$

The second step assumes that the permeability does not depend on the spatial coordinates, and Δ is the Laplace operator. A formulation for the 2D case is enough to model the basic aspects of radial flux machines. Therefore, the PDE can be simplified for an infinitely extended z -axis:

$$\frac{1}{\mu} \cdot \left(\frac{\partial^2 A_z}{\partial x^2} + \frac{\partial^2 A_z}{\partial y^2} \right) = J_z \quad (11)$$

The FEM is used to solve the PDE. At first, the 2D geometry of the machine is meshed in FEMM using first-order triangular elements with three nodes (see Figure 4). The meshing algorithm is based on the Delaunay method (Meeker, 2019).

The z -component of the vector potential $A_z^e(x, y)$ inside an element e is calculated using the area of the element Ω^e and interpolation constants a_n , b_n and c_n . The interpolation function $N_n(x, y)$ is 1 at the position of node n and 0 at the position of the other nodes (Polycarpou, 2006). This results in:

$$N_n(x, y) = \frac{1}{2 \Omega^e} \cdot (a_n + b_n \cdot x + c_n \cdot y) \quad (12)$$

$$A_z^e(x, y) = A_{1,z} \cdot N_1(x, y) + A_{2,z} \cdot N_2(x, y) + A_{3,z} \cdot N_3(x, y) \quad (13)$$

Therefore, the vector potentials $A_{1,z}$, $A_{2,z}$ and $A_{3,z}$ of the three nodes must be calculated first. Using the method of weighted residuals and partial integration, the given 2D PDE [equation (11)] is converted to the FEM matrix equation. Because of the nonlinear permeability, the equation is solved using the fixed-point algorithm or the Newton–Raphson method (Aliprantis and Wasynczuk, 2023). The solution contains the vector potentials of all nodes inside the given mesh. The three corresponding vector potentials of an element e (see Figure 4) are used to calculate the flux density:

$$\vec{B}^e = \text{rot} \vec{A}^e = \text{rot} \begin{pmatrix} 0 \\ 0 \\ A_z^e \end{pmatrix} = \begin{pmatrix} \frac{\partial A_z^e}{\partial y} \\ -\frac{\partial A_z^e}{\partial x} \\ 0 \end{pmatrix} = \frac{1}{2 \Omega^e} \begin{pmatrix} A_{1,z} \cdot c_1 + A_{2,z} \cdot c_2 + A_{3,z} \cdot c_3 \\ -(A_{1,z} \cdot b_1 + A_{2,z} \cdot b_2 + A_{3,z} \cdot b_3) \\ 0 \end{pmatrix} \quad (14)$$

Afterward, the flux densities are smoothed using the nearest neighbor algorithm. This results in flux densities that are not constant inside an element (Meeker, 2019; Polycarpou, 2006).

2.4 Evaluation and results

The flux densities are used to calculate different quantities and waveforms of the machine. A magnetic flux Φ in one part of the machine is derived from the integration of the flux density over the relevant cross-section area. All flux that goes through the coils of one phase contributes to the corresponding flux linkage. For phase U the flux linkage is as follows:

$$\Psi_U = w \cdot \Phi_U \quad (15)$$

Here, w is the number of turns. The flux linkage is evaluated for different mechanical rotation angles φ_{mech} of the rotor. Because of the periodicity of machines with $p > 1$, the waveforms are plotted over a full electrical period with $\varphi_{\text{el}} = ps^2 \varphi_{\text{mech}}$. Note, that these (temporal) angles can be converted to corresponding time values based on the electrical frequency f_1 of the machine. An example solution for all three phases can be seen in Figure 5 at the bottom left.

Next, the back-EMF u_i of a phase can be derived based on Faraday’s law of induction (Pyrhönen *et al.*, 2013):

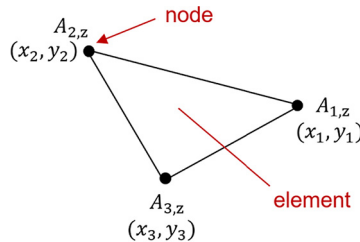


Figure 4.
First-order triangular
element

Source: Figure created by authors

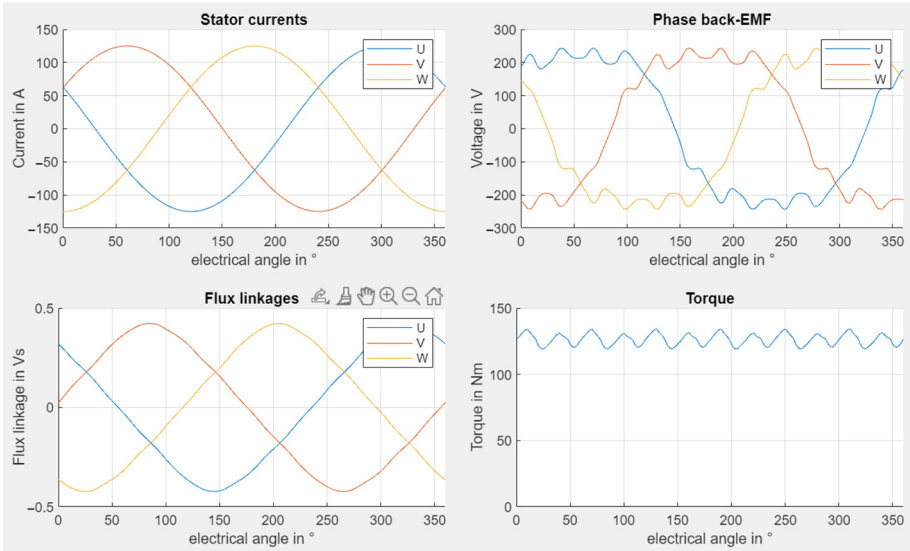


Figure 5. Result plots for the Toyota Prius 2004 with 125A at 1,540 rpm

Source: Figure created by authors

$$u_i = -\frac{d\Psi}{dt} = -2\pi \cdot n \cdot p \cdot \frac{d\Psi}{d\varphi_{el}} \quad (16)$$

The speed of the rotor n is used. Figure 5 shows example waveforms at the upper-right.

To calculate the torque of the machine, Maxwell's stress tensor for the two-dimensional case in cylindrical coordinates (r, φ, z) is used:

$$\mathbf{T}_m = \frac{1}{2\mu} \cdot \begin{pmatrix} B_r^2 - B_\varphi^2 & 2 \cdot B_r \cdot B_\varphi \\ 2 \cdot B_r \cdot B_\varphi & B_\varphi^2 - B_r^2 \end{pmatrix} \quad (17)$$

This term is evaluated for every element inside the airgap. Then, the force vector can be calculated by integration over the rotor volume V_r or the rotor surface A_r and using the outer radius of the rotor $r_{o,r}$ (Binder, 2012):

$$\vec{F} = \begin{pmatrix} F_r \\ F_\varphi \end{pmatrix} = \iiint_{V_r} \text{div} \mathbf{T}_m \cdot dV = \oiint_{\partial V_r} \mathbf{T}_m \cdot d\vec{A}_r = \iint_{A_r} \frac{1}{2\mu} \begin{pmatrix} B_r^2 - B_\varphi^2 \\ 2 \cdot B_r \cdot B_\varphi \end{pmatrix} \cdot r_{o,r} \, d\varphi \, dz \quad (18)$$

The torque has only a z -component that can be derived from the given force:

$$\vec{T} = \vec{r} \times \vec{F} = \begin{pmatrix} r_{o,r} \\ 0 \\ 0 \end{pmatrix} \times \begin{pmatrix} F_r \\ F_\varphi \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ r_{o,r} \cdot F_\varphi \end{pmatrix} \quad (19)$$

The torque waveform for the example geometry is shown in [Figure 5](#) at the bottomright. The maximum electrical angle for the simulation and the overall number of points per simulation are input values inside the GUI. A preview plot shows the expected waveforms of the current with one period and the torque with ν periods over 360° electrically, where:

$$\nu = \frac{\text{lcm}(2p, N)}{p} \quad (20)$$

(Müller *et al.*, 2008; Pyrhönen *et al.*, 2013).

A core loss calculation based on loss separation (Bertotti, 1988; Zhou and Bowen, 2020) is implemented. The user can specify the regression coefficients C_h , C_e , C_{ex} , n_h and n_{ex} for the used material data. The flux density is evaluated in every mesh element over the simulation steps. Then, the flux density waves are decomposed into their harmonic content. The specific losses are calculated based on the harmonic flux density amplitude of every element and the corresponding harmonic frequency f_k :

$$p_{c,k}^e = C_h \cdot f_k \cdot (B^e)^{n_h} + C_e \cdot f_k^2 \cdot (B^e)^2 + C_{ex} \cdot (f_k \cdot B^e)^{n_{ex}} \quad (21)$$

For every element, the harmonic losses are summed. Afterward, a summation of all element losses multiplied with the element volume V^e is performed:

$$P_c = \sum_e \left(\left(\sum_k p_{c,k}^e \right) \cdot V^e \right) \quad (22)$$

2.5 Coordinate transformation

It is possible to change the amplitude and phase shift for the three-phase current. Alternatively, d - and q -currents can be set directly. The following transformation formula (Park, 1929) is used:

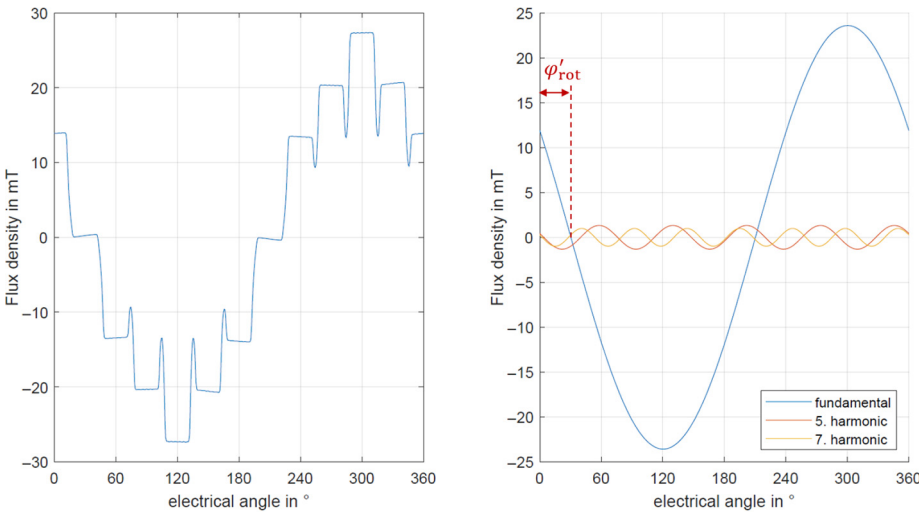
$$\begin{pmatrix} i_d \\ i_q \end{pmatrix} = \frac{2}{3} \begin{pmatrix} \cos(\varphi) & \cos\left(\varphi - \frac{2\pi}{3}\right) & \cos\left(\varphi - \frac{4\pi}{3}\right) \\ -\sin(\varphi) & -\sin\left(\varphi - \frac{2\pi}{3}\right) & -\sin\left(\varphi - \frac{4\pi}{3}\right) \end{pmatrix} \begin{pmatrix} i_U \\ i_V \\ i_W \end{pmatrix} \quad (23)$$

Therefore, the initial rotor position as a reference must be found. At first, the normal airgap flux density is calculated for a model without permanent magnets (material replaced with air) over the spatial angle. The phase shift of the fundamental component φ'_{rot} is evaluated using a Fourier decomposition of the waveform. The corresponding plots can be seen in [Figure 6](#).

The mechanical rotation angle for the initial position can then be calculated using the number of pole pairs:

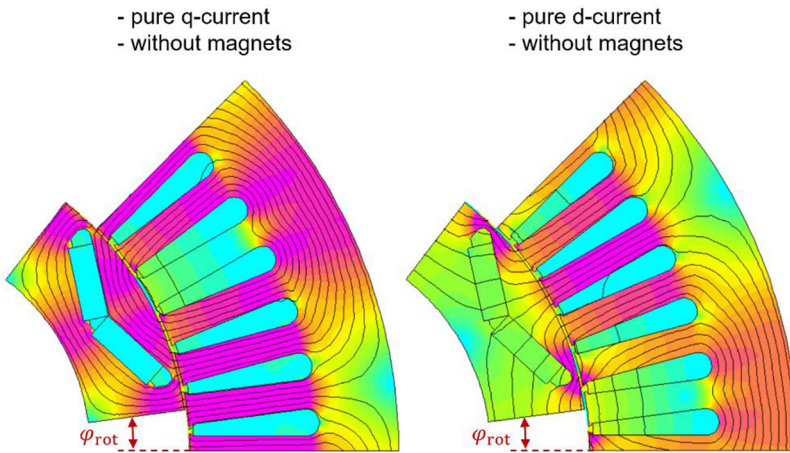
$$\varphi_{rot} = \frac{90^\circ - \varphi'_{rot}}{p} \quad (24)$$

An angle of 90° is used to put the initial position on the q -axis. [Figure 7](#) shows an example of pure d - or q -currents without permanent magnets. The pure q -current is generated with a current phase shift of 0° . A phase shift of -90° is needed for a pure positive d -current.



Source: Figure created by authors

Figure 6. Airgap flux density plot and harmonic waves (fundamental, 5. and 7.)



Source: Figure created by authors

Figure 7. Magnetic flux lines for a pure q -current (left) and a pure d -current (right)

Then, the flux linkages and inductances in dq -coordinates can be calculated. The values can also be plotted over different d - and q -current pairs. Starting from this point, a torque-speed-diagram can be generated.

3. Tool validation

The electrical motor from the Toyota Prius 2004 is used as a reference machine to validate the functionalities of the presented tool. The necessary data are taken from

Burruss *et al.* (2008) and Hsu *et al.* (2004). Figure 8 shows the motor geometry that is used inside of FEMM. Due to its symmetry, 1/8 of the machine is enough to characterize the magnetic field. Therefore, symmetry conditions must be defined. Here, only one pole is modeled, so an antiperiodic condition is needed. Dirichlet boundaries ($A = 0$) are used for the inner and outer radius of the motor.

After simulation, the results are compared to measured locked-rotor torque curves. The plot for a peak current of 250A is displayed in Figure 9. It can be shown that the results obtained using the tool qualitatively agree with the comparison values. In consequence, they can be considered valid. Quantitative deviations can be attributed to unknown measurement conditions and inaccuracies. The procedure for obtaining the locked-rotor simulation results is described in Katona and Orosz (2022).

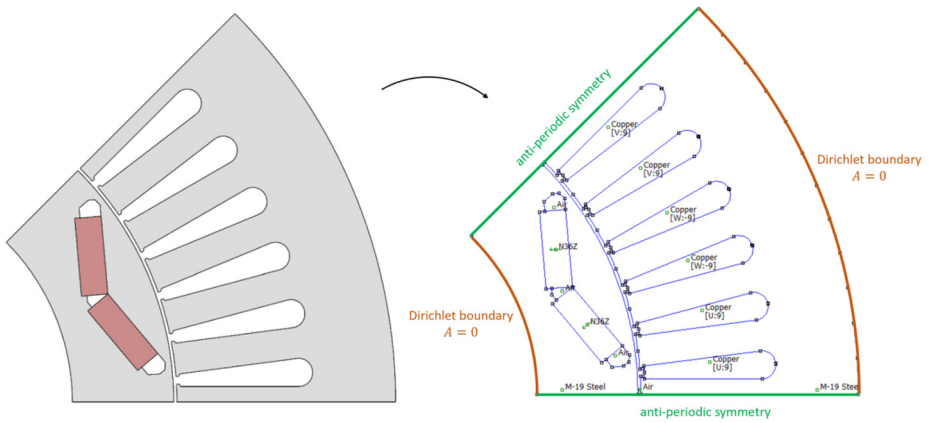


Figure 8.
Toyota Prius 2004
motor model (left:
gray = electrical
steel, red = magnets |
right: model in
FEMM)

Source: Figure created by authors

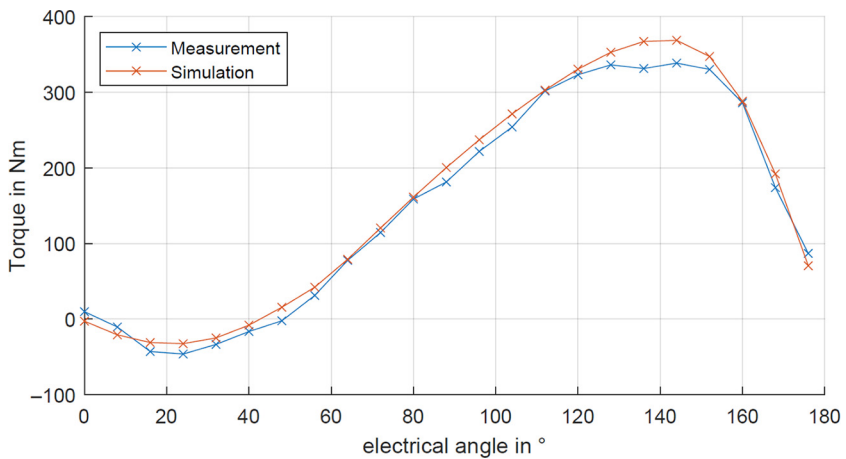


Figure 9.
Locked rotor torque
of the Prius 2004 for
250A

Source: Figure created by authors

4. Educational purposes

The presented tool offers a variety of possible applications for educational purposes in the field of electrical machines, which is supported by its modular and transparent structure. This allows different levels of detail that can be used for teaching (see Figure 10) to achieve a better understanding of electrical machines and the underlying theory (Stoev *et al.*, 2017) and thus educational success:

- Level 1: Using the GUI to simulate and design PMSM. This includes, for example, the usage of parameter studies and to learn about winding design.
- Level 2: Looking inside the underlying MATLAB scripts. The students learn how to run FEMM and how the solution can be used to calculate the back-EMF, torque, *dq*-inductances, torque-speed-diagrams and efficiency maps.
- Level 3: Learning about the basic principles of FEM. It can be taught how FEMM calculates all field quantities based on the solution of a PDE. This knowledge can also be applied to other disciplines, like mechanical or thermal simulations.

The desired degree of detail can be chosen together with the students. This provides an opportunity to adapt the teaching to the interests of the students. In addition, different types of tasks for lectures, student projects or theses are possible. This includes a complete motor design using the GUI, an extension of the MATLAB scripts with new functionalities or the modeling of real motors and the comparison of simulation results with measurement data. In contrast to many other projects, students who are involved in expanding the GUI can be certain that their results will find application over the long term, for example, in the context

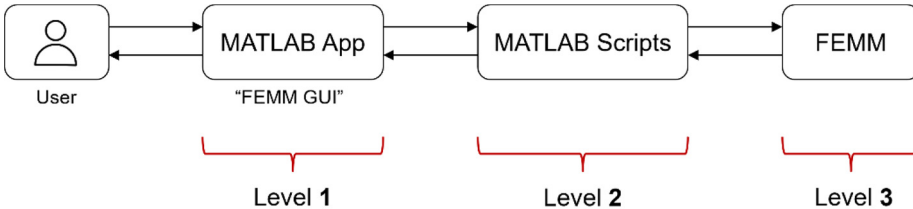


Figure 10. Educational levels of detail using the presented tool

Source: Figure created by authors

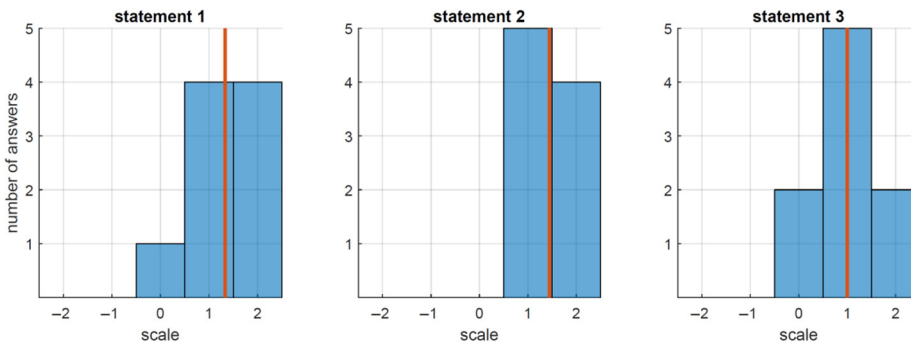


Figure 11. Results of the student survey (scale: -2 = totally disagree, -1 = rather disagree, 0 = neutral, 1 = rather agree and 2 = totally agree)

Source: Figure created by authors

of lectures and research tasks. Overall, this can contribute to greater motivation among the students. Several theses have already been assigned to extend the simulation tool, e.g. the addition of parametrized hairpin models. A survey was conducted with students after using the program for motor simulations. The following three statements were evaluated:

- statement 1: “My motivation for the topics is increased by using the program”;
- statement 2: “I would like to use the program in more lectures or projects”; and
- statement 3: “I would like to learn more about how the program works”.

The answers are based on a scale from -2 (= totally disagree) to 2 (= totally agree). [Figure 11](#) shows the results for a sample size of nine students. The mean values are marked with red lines.

Due to the small number of participants, the survey is not representative. Because the participants were all electrical engineering students with a clear interest in the subject of electrical machines, it is still possible to derive trends from the results. Looking at the answers to the first statement, it is noticeable that the motivation of most participants in relation to the topic has increased using the program. For the second statement, it is remarkable that all students surveyed would like to use the software in future courses. Regarding the answers to the third statement, some students are very interested in learning how exactly the program works, others are not particularly concerned. Overall, using the tool in the context of teaching is very well received by the students surveyed, but not everyone necessarily wants to understand exactly how the underlying functions and calculations work.

5. Summary and outlook

This paper provides a comprehensive overview of the FEMM GUI, an easy-to-use tool for the simulation of electrical machines, which was developed primarily to be used in the context of education. Although a large variety of functions and models are already included, there is still much room for improvement. The full support of electrically excited synchronous machines and induction machines could be a possible extension in the future. Also, thermal simulations in FEMM and mechanical simulations using other free software could be included and linked to the already existing models. The presented tool will continue to offer students the opportunity to incorporate their own ideas.

References

- Aliprantis, D. and Wasynczuk, O. (2023), *Electric Machines. Theory and Analysis Using the Finite Element Method*, Cambridge University Press, Cambridge.
- Bertotti, G. (1988), “General properties of power losses in soft ferromagnetic material”, *IEEE Transactions on Magnetics*, Vol. 24 No. 1, pp. 621-630.
- Binder, A. (2012), *Elektrische Maschinen Und Antriebe: Grundlagen, Betriebsverhalten*, Springer, Heidelberg.
- Bonneel, P., Le Besnerais, J., Devillers, E., Marinel, C. and Pile, R. (2018), “Pyleecan: an open-source python object-oriented software for the multiphysics design optimization of electrical machines”, 2018 XIII International Conference on Electric Machines (ICEM), IEEE, *Alexandroupoli, Greece*.
- Burress, T.A., Coomer, C.L., Campbell, S.L., Seiber, L.E., Marlino, L.D., Staunton, R.H. and Cunningham, J.P. (2008), “Evaluation of the 2007 toyota camry hybrid synergy drive system”, Oak Ridge National Laboratory, TN.

- de Andrade, K.M., Santos, H.E., Vilela, W.M., de Almeida, T.E.P. and de Paula, G.T. (2019), "PeMSyn: a free software to assist the design and performance assessment of permanent magnets synchronous machines", 2019 IEEE 15th Brazilian Power Electronics Conference and 5th IEEE Southern Power Electronics Conference (COBEP/SPEC), IEEE, Santos, Brazil, pp. 1-6.
- Hsu, J.S., Ayers, C.W. and Coomer, C.L. (2004), "Report on Toyota/Prius Motor Design and Manufacturing Assessment", Oak Ridge National Laboratory, TN.
- Katona, M. and Orosz, T. (2022), "Locked-rotor analysis of a prius 2004 IPMSM motor with Digital-twin-distiller", 2022 IEEE 20th International Power Electronics and Motion Control Conference (PEMC), IEEE, Brasov, Romania.
- Kuptsov, V., Fajri, P., Trzynadlowski, A., Zhang, G. and Magdaleno-Adame, S. (2019), "Electromagnetic analysis and design methodology for permanent magnet motors using motoranalysis-PM software", *Machines*, Vol. 7 No. 4, p. 75.
- Lehikoinen, A., Davidsson, T., Arkkio, A. and Belahcen, A. (2018), "A High-Performance Open-Source finite element analysis library for magnetics in MATLAB", 2018 XIII International Conference on Electrical Machines (ICEM), IEEE, Alexandroupoli, Greece.
- Meeker, D.C. (2019), "Finite element method magnetics", Version 4.2 (21 April 2019 Build), available at: www.femm.info (accessed 16 August 2023).
- Müller, G., Vogt, K. and Ponick, B. (2008), "*Berechnung Elektrischer Maschinen*", WILEY-VCH, Weinheim.
- Park, R.H. (1929), "Two-Reaction theory of synchronous machines. Generalized method of analysis-part I", *Transactions of the American Institute of Electrical Engineers*, Vol. 48 No. 3, pp. 716-727.
- Polycarpou, A.C. (2006), *Introduction to the Finite Element Method in Electromagnetics*, Springer, Cham.
- Pyrhönen, J., Jokinen, T. and Hrabovcová, V. (2013), "*Design of Rotating Electrical Machines*", 2nd ed. WILEY-VCH, Weinheim.
- Stoev, B., Todorov, G., Rizov, P., Pagiatakis, G. and Dritsas, L. (2017), "Finite element analysis of rotating electrical machines – an educational approach", 2017 IEEE Global Engineering Education Conference (EDUCON), IEEE, Athens, Greece.
- Zhou, J. and Bowen, H. (2020), "Summary of calculation methods of PMSM iron loss based on finite element analysis", IOP Conference Series: materials Science and Engineering, Vol. 768, Shanghai.

Corresponding author

Alexander Schugardt can be contacted at: schugardt@tu-berlin.de

For instructions on how to order reprints of this article, please visit our website:

www.emeraldgroupublishing.com/licensing/reprints.htm

Or contact us for further details: permissions@emeraldinsight.com