High–Fidelity Simulation of Electrical Machines: A Comparison of Modeling Techniques



(Article begins on next page)

# High–Fidelity Simulation of Electrical Machines: A Comparison of Modeling Techniques

Filip Jukić*, Simone Ferrari†, Paolo Pescetto†, Kristijan Gaži*, Stjepan Stipetić*, Gianmario Pellegrino†
* University of Zagreb Faculty of Electrical Engineering and Computing, Zagreb, Croatia
† Energy Department "Galileo Ferraris", Politecnico di Torino, Turin, Italy
filip.jukic@fer.hr, simone.ferrari@polito.it, paolo.pescetto@polito.it, kristijan.gazi@fer.hr, stjepan.stipetic@fer.hr, gianmario.pellegrino@polito.it

Abstract—High–fidelity models of electrical machines are essential for accurate simulations in modern applications. These models enable realistic analysis, control system design and optimization prior to hardware implementation, thus accelerating the validation process. In simulation environments, both the circuital and non–circuital approaches are in use for high–fidelity modeling. To capture key nonlinearities such as magnetic saturation, cross-coupling and spatial harmonics both approaches rely on look-up tables (LUTs) derived from finite element analysis (FEA). These LUTs also help model various loss mechanisms, including iron, magnet and copper losses. The circuital approach represents the machine by equivalent electrical circuits, while the non–circuital approach directly implements the equations governing the behavior of the machine without explicit representation of the circuits. This paper presents a comparative analysis of these modeling approaches, focusing on their implementation in MATLAB/Simulink using the FEA dataset of a permanent magnet machine within the open–source tool SyR–e. The study evaluates key performance criteria such as accuracy and computational efficiency, while the results provide valuable insights for the selection of a suitable modeling strategy based on application-specific requirements, e.g. non–circuital models offering higher computational speed and reduced software dependency, whereas circuital models enable more detailed operational analysis.

Index Terms—Electric machines, permanent magnet machines, high–fidelity modeling, lookup tables, computer simulation.

## I. Introduction

The increasing demand for electrification, particularly in the automotive domain [1], is pushing for the rapid development of control algorithms while reducing associated testing costs. Model–Based Design (MBD) has proven to be an effective method for overcoming these challenges by using a structured V–model framework [2]. This framework encompasses the Model–in–the–Loop (MiL), Software–in–the–Loop (SiL), Processor–in–the–Loop (PiL) and Hardware–in–the–Loop (HiL) phases, enabling a systematic transition from high–level simulations to implementation on physical hardware.

To ensure accurate and robust control development within this framework, high–fidelity electrical machine models are essential, especially for the widespread class of three–phase AC machines used in power generation, automation and electric transportation [3], [4]. However, these machines – notably magnetically anisotropic designs such as permanent magnet synchronous machines (PMSMs), synchronous reluctance machines (SynRMs) and permanent magnet–assisted synchronous reluctance machines (PM–SynRMs) – exhibit strong nonlinearities, including magnetic saturation, cross–coupling, spatial harmonics and temperature–dependent effects, which conventional linearized $dq$ frame models, such as current–based (CMs) or flux–based models (FMs), cannot adequately capture [5], [6].

Therefore, to account for these nonlinearities, CMs and FMs are typically supplemented by maps, i.e. lookup tables (LUTs), derived from finite element analysis (FEA). Two–dimensional $dq$ LUTs are used to capture magnetic saturation and cross–coupling, where CMs typically use apparent and incremental inductance maps, $\boldsymbol{L}_{dq}(\boldsymbol{i}_{dq})$ and $\boldsymbol{l}_{dq}(\boldsymbol{i}_{dq})$, while FMs employ inverse flux linkage map, $\boldsymbol{i}_{dq}(\boldsymbol{\psi}_{dq})$ [5]. Consequently, the electromagnetic torque can be calculated using the current–flux cross product or a dedicated torque map, $\boldsymbol{T}_e(\boldsymbol{i}_{dq})$, can be used instead. For further modeling of the spatial harmonics, three–dimensional $dq\theta$ LUTs such as $\boldsymbol{i}_{dq}(\boldsymbol{\psi}_{dq}, \theta_{re})$, $\boldsymbol{L}_{dq}(\boldsymbol{i}_{dq}, \theta_{re})$, $\boldsymbol{l}_{dq}(\boldsymbol{i}_{dq}, \theta_{re})$ and $\boldsymbol{T}_e(\boldsymbol{i}_{dq}, \theta_{re})$ are usually employed [6]–[9], while four–dimensional LUTs such as $\boldsymbol{i}_{dq}(\boldsymbol{\psi}_{dq}, \theta_{re}, \Theta)$ extend this approach to also capture temperature effects [10]–[12]. Finally, loss mechanisms, including iron core, permanent magnet (PM) and AC copper losses, can be included via analytical expressions or loss–specific LUTs to enhance model accuracy over a wide operating range [13], [14].

The implementation of such high–fidelity electrical machine models in simulation environments generally follows either a circuital or non–circuital approach. Circuital models are based on equivalent electrical circuits assembled from components available in specialized toolboxes and libraries of simulation platforms, whereas non–circuital models implement CMs or FMs using differential equations and transfer functions. Among simulation platforms, MATLAB/Simulink is the most widely adopted in published studies, though only a few explicitly state the modeling approach, e.g., the non–circuital strategies reported in [8], [12]. However, since most studies do not explicitly reference MATLAB/Simulink circuital toolboxes such as Simscape Electrical, it can be concluded that non–

circuital approaches are more widespread in academic research. In contrast, commercial simulation tools, including MATLAB/Simulink, but also dedicated platforms such as PLECS, tend to favor circuital modeling, supported by extensive electric component libraries and application examples. Within MATLAB/Simulink, Simscape Electrical, for instance, offers a comprehensive set of machine blocks, including the FEM–Parameterized PMSM model [15], which supports the use of LUTs to account for a wide range of nonlinearities. Nevertheless, these blocks restrict the user's access to the internal model structure, thereby limiting transparency. PLECS, on the other hand, grants full model access and supports both the Rotor Reference Frame (RRF) and the Voltage Behind Reactance (VBR) formulation, but in the current release only the VBR formulation integrates LUTs to represent the nonlinear machine behavior [16].

Besides studies on non–circuital modeling, the only detailed performance evaluation of circuital models known to the authors is presented in [13], where the SyR–e framework [17], [18] was used to compare PMSM circuital models within MATLAB/Simulink and PLECS. In this work, the VBR formulation from PLECS was re–implemented in MATLAB/Simulink, while a Controlled Current Generator (CCG) model, functionally equivalent to the RRF implementation in PLECS, was developed from scratch and deployed in both environments. The comparative analysis highlighted the trade–offs between using $dq$ and $dq\theta$ LUTs and showed that the CCG model consistently outperformed the VBR approach on both platforms in terms of computational efficiency, with PLECS platform offering a higher simulation speed overall. Based on these results, the CCG model was selected as the benchmark for comparison with experimental results under both healthy and faulty operating conditions, showing strong agreement with the measured data.

Despite the aforementioned contributions, comprehensive comparative analysis between circuital and non–circuital high–fidelity modeling approaches are still limited. This paper fills this gap by evaluating circuital and non–circuital PMSM models within MATLAB/Simulink using the SyR-e simulation framework, which natively supports the CCG, VBR and Simscape FEM–Parameterized PMSM models. To enable a direct comparison, a high–fidelity non–circuital FM model was additionally developed and integrated into the SyR–e framework. The CCG block was chosen as a circuital reference due to its favorable ratio between accuracy and computational cost, as reported in [13], while the FEM–Parameterized block served as a library–based high–fidelity benchmark. With all models coexisting in a unified environment, this study, therefore, presents a systematic evaluation of different modeling strategies under identical conditions, focusing on dynamic accuracy, modeling flexibility and computational efficiency.

## II. High–fidelity model of the PMSM

### A. Nonlinear FM of PMSM

The nonlinear FM of a PMSM in the rotating $dq$ reference frame, accounting for magnetic saturation, cross–coupling effects and spatial harmonics can be expressed as follows [7]:

1) Electrical model:

$$\boldsymbol{u}_{dq} = R_s \boldsymbol{i}_{dq} + \frac{d\boldsymbol{\psi}_{dq}}{dt} + \omega_{re} \boldsymbol{J} \boldsymbol{\psi}_{dq}, \tag{1}$$

$$\boldsymbol{\psi}_{dq} = f(\boldsymbol{i}_{dq}, \theta_{re}), \tag{2}$$

2) Mechanical model:

$$\boldsymbol{T}_e = f(\boldsymbol{\psi}_{dq}, \theta_{re}), \tag{3}$$

$$\frac{d\omega_{re}}{dt} = p(T_e - T_l), \tag{4}$$

where $\boldsymbol{u}_{dq} = [u_d \ u_q]^{\mathrm{T}}$ is the stator voltage, $\boldsymbol{i}_{dq} = [i_d \ i_q]^{\mathrm{T}}$ is the stator current, $\boldsymbol{\psi}_{dq} = [\psi_d \ \psi_q]^{\mathrm{T}}$ is the stator flux, $R_s$ is the stator resistance, $p$ are the pole pairs, $\omega_{re}$ is the electrical rotor speed, $\theta_{re}$ is the electrical rotor position, $T_e$ is the electromagnetic torque, $T_l$ is the load torque and $\boldsymbol{J} = \left[\begin{smallmatrix} 0 & -1 \\ 1 & 0 \end{smallmatrix}\right]$ is the rotation matrix.

### B. Nonlinear FM of PMSM including losses

To achieve the desired mechanical output power at the shaft, the electrical input power must also take into account the internal losses, including copper, iron and PM losses, as well as the mechanical losses:

$$P_{el}^{in} = P_{shaft}^{out} + \underbrace{P_{Cu} + P_{Fe} + P_{PM} + P_{mech}}_{P_{loss}}. \tag{5}$$

Excluding mechanical losses, the electrical part of the FM under steady–state conditions, i.e. $d\boldsymbol{\psi}_{dq}/dt = 0$, can be represented by the equivalent circuit diagram in Fig. 1. Compared to the equations (1) and (2), this circuit contains an additional resistance branch, denoted by $R_{Fe}$, to model the iron and PM losses, while the copper losses are captured by the resistance $R_s$. Both $R_s$ and $R_{Fe}$ are treated as variable resistors, as the associated losses depend on operating conditions such as temperature and electrical speed. Under this representation, the stator current is divided into two components:

$$\boldsymbol{i}_{dq} = \boldsymbol{i}_{dq}^{Fe} + \boldsymbol{i}_{dq}^{m}, \tag{6}$$

where $\boldsymbol{i}_{dq}^{Fe} = [i_d^{Fe} \ i_q^{Fe}]^{\mathrm{T}}$ is the equivalent loss current and $\boldsymbol{i}_{dq}^{m} = [i_d^{m} \ i_q^{m}]^{\mathrm{T}}$ is the magnetizing current.

Accordingly, the losses are calculated as:

$$P_{Cu} = \frac{3}{2} R_s |\boldsymbol{i}_{dq}|^2, \tag{7}$$

$$P_{Fe} + P_{PM} = \frac{3}{2} \omega_{re} \big( \boldsymbol{i}_{dq}^{Fe^T} \boldsymbol{J} \boldsymbol{\psi}_{dq}^{*} \big), \tag{8}$$

where $\boldsymbol{\psi}_{dq}^{*} = [\psi_d \ \ -\psi_g]^T$ is the complex–conjugate of the stator flux.

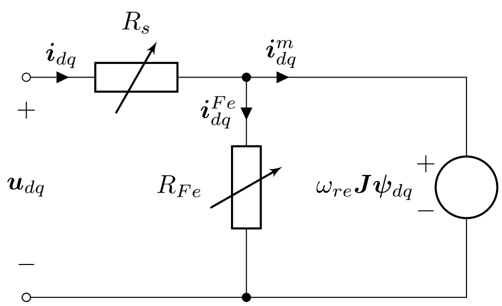


Fig. 1: Steady–state equivalent circuit of PMSM

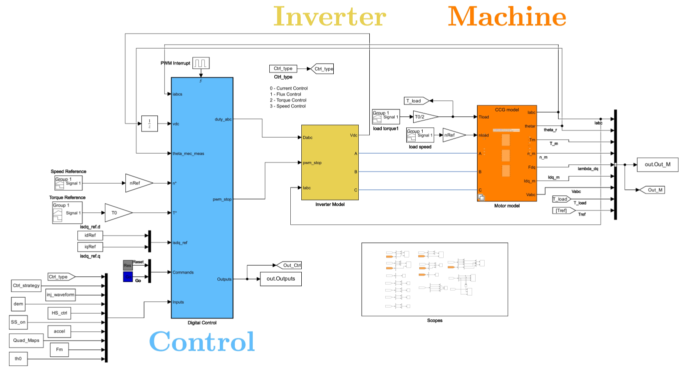


Fig. 2: SyR–e default simulation framework in MATLAB/ Simulink.

## III. SyR–e simulation framework

### A. Default SyR–e simulation framework

The high–fidelity PMSM model introduced in Section II which incorporates the principal nonlinearities and loss mechanisms, is supported by SyR–e tool. The latest SyR–e release can automatically generate models for PMSMs, SynRMs and PM-SynRMs in both MATLAB/Simulink and PLECS, with user–selectable complexity and accuracy levels. In MATLAB/Simulink, SyR–e adopts the standardized architecture shown in Fig. 2, comprising three core subsystems: Control, Inverter and Machine. The Control block is implemented as a C–based S–function and supports sensored or sensorless torque, current or speed control using either field–oriented control (FOC) or direct–flux vector control (DFVC). By default, the Inverter and Machine blocks follow a circuital approach. The block Inverter is created using the Simscape Electrical component Converter (Three-Phase) and supports two configurations: Average Switch and Ideal Semiconductor Switch. The former calculates average voltages over PWM duty cycles, while the latter emulates ideal IGBT switching behavior. The Machine block allows for the selection among CCG, VBR or Simscape FEM–Parameterized PMSM models with support for $dq$ or $dq\theta$ LUTs and optional inclusion of iron, PM and AC copper losses.

### B. Utilized SyR–e simulation framework configurations

Considering the configurability of the SyR–e simulation framework, Fig. 3 provides a detailed overview of the specific setups adopted in this study, complementing the standard structure outlined above. While the Control block remains unchanged in all configurations using a sensored FOC for torque control, the Inverter and Machine blocks are varied based on three configuration switches:

- SW1 – toggles between circuital and non–circuital modeling of Inverter and Machine blocks,
- SW2 – selects between the CCG and Simscape FEM–Parameterized PMSM for circuital machine modeling,
- SW3 – chooses between average and instantaneous modeling of the Inverter block.

When the SW1 switch is set to CIRC, the default SyR–e circuital modeling approach is applied to both the Inverter block and the Machine block, using the Simscape Electrical components highlighted in blue. The voltage $\boldsymbol{u}_{abc}$ is generated by the Converter block and applied to the machine model, which is selected via the SW2 switch between the CCG and the Simscape FEM–Parameterized PMSM model.

On the other hand, setting the switch SW1 to NON–CIRC enables the non–circuital modeling approach, which is not inherently included in the SyR–e framework, but was developed and integrated as part of this study. In this configuration, all Simscape Electrical components are excluded, and both the Inverter and Machine blocks are implemented directly applying mathematical models. Consequently, the Inverter calculates the output voltage analytically based on the following expressions:

- Average mode:
$$\boldsymbol{u}_{abc} = \boldsymbol{V}\boldsymbol{d}_{abc}^{*}, \tag{9}$$
- Instantaneous mode:
$$\boldsymbol{u}_{abc} = \boldsymbol{V}\boldsymbol{s}_{abc}^{*}, \tag{10}$$

where $\boldsymbol{d}_{abc}^{*} = [d_a^* \ d_b^* \ d_c^*]^T$ denotes the duty cycle including the dead time, $\boldsymbol{s}_{abc}^{*} = [s_a^* \ s_b^* \ s_c^*]^T$ represents the PWM switching signals for the upper IGBTs, with the lower IGBTs operating in a complementary manner, also including dead time. The matrix $\boldsymbol{V}$, assuming a floating neutral point of the machine, is

$$\boldsymbol{V} = \frac{2U_{dc}}{3}\begin{bmatrix} 1 & -1/2 & -1/2 \\ -1/2 & 1 & -1/2 \\ -1/2 & -1/2 & 1 \end{bmatrix}, \tag{11}$$

where $U_{dc}$ is the DC–link voltage.

Finally, the selection between applying an average or a PWM–generated voltage to the machine model is determined by the switch SW3 and is available in both circuital and non–circuital modeling modes. In the non–circuital approach, the voltage is computed analytically according to (9) and (10), while in the circuital approach, as stated earlier, the voltage is generated by the Converter block, which directly receives either the duty–cycle $\boldsymbol{d}_{abc}^{*}$ or the PWM switching signals $\boldsymbol{s}_{abc}^{*}$ as inputs.

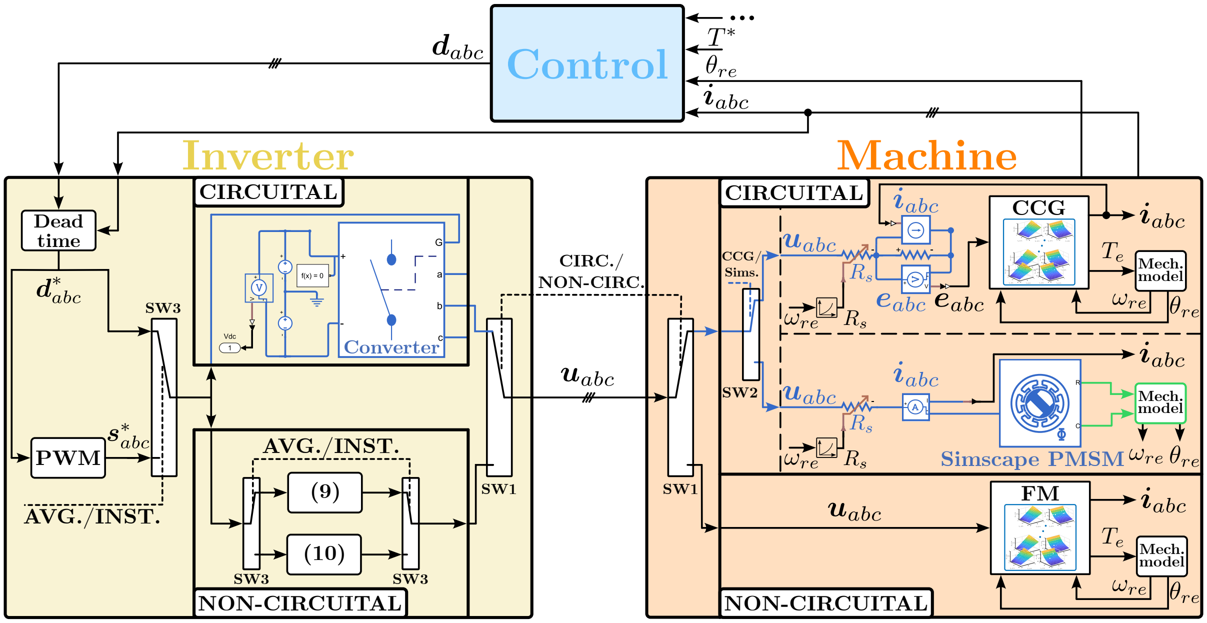

Fig. 3: Configurations utilized within the SyR–e simulation framework.

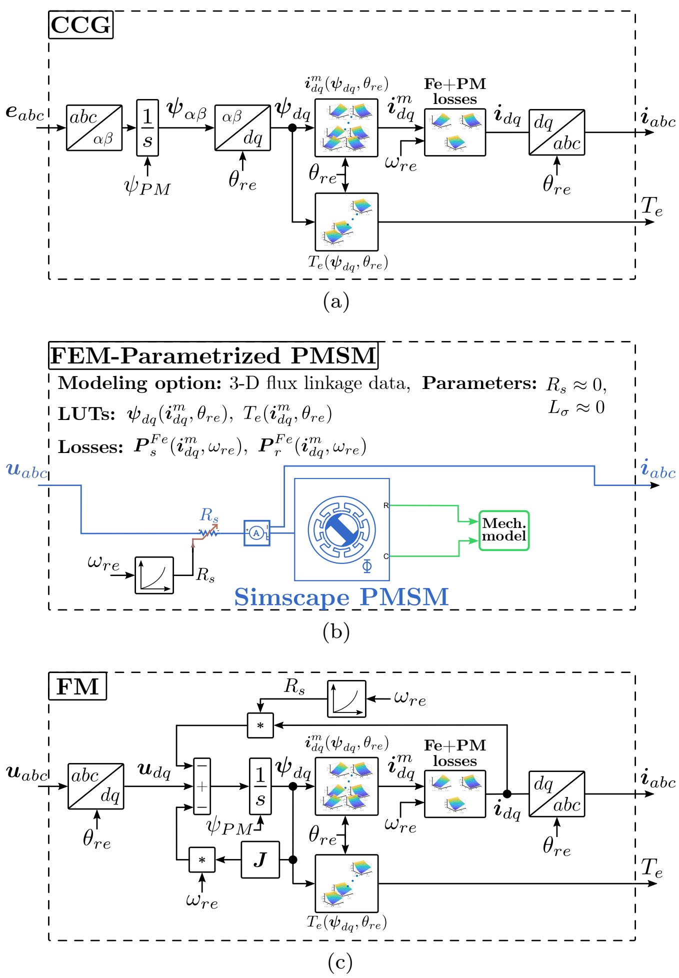

Fig. 4: PMSM models implemented within SyR–e simulation framework. (a) CCG. (b) Simscape PMSM. (c) FM.

### C. High–fidelity PMSM models within SyR–e framework

To accurately capture nonlinear relationships such as (2) and (3), high–fidelity PMSM models in simulation environments generally rely on $dq\theta$ LUTs that are precalculated using FEA [19]. However, in consideration of the iron and PM core losses, the stator phase current and magnetizing current need to be treated separately. As described by (6), these currents are distinct, and consequently, the flux map defined by (2) should be computed with respect to the magnetizing current plane, $(i^m_d, i^m_q)$, rather than the $(i_d, i_q)$ current plane, i.e.:

$$\boldsymbol{\psi}_{dq} = f(\boldsymbol{i}^m_{dq}, \theta_{re}). \tag{12}$$

Similarly, to separate iron and PM losses and further decompose iron losses into hysteresis and eddy current components, these loss contributions can also be precalculated individually over the entire $(i^m_d, i^m_q)$ current plane at a constant frequency using FEA, rather than relying solely on the analytical model (8). These loss LUTs can then be extrapolated to variable–speed conditions using Steinmetz–based scaling laws, as follows [13], [19]:

$$\boldsymbol{P}_{Fe} = \boldsymbol{P}_{h0}\left(\frac{\omega_{re}}{\omega_{re,0}}\right)^{\alpha} + \boldsymbol{P}_{e0}\left(\frac{\omega_{re}}{\omega_{re,0}}\right)^{2}, \tag{13}$$

$$\boldsymbol{P}_{PM} = \boldsymbol{P}_{PM0}\left(\frac{\omega_{re}}{\omega_{re,0}}\right)^{2}, \tag{14}$$

where $\boldsymbol{P}_{h0} = f(\boldsymbol{i}^m_{dq})$ are the hysteresis losses, $\alpha$ is the material-dependent Steinmetz exponent, $\boldsymbol{P}_{e0} = f(\boldsymbol{i}^m_{dq})$ are the eddy current losses and $\boldsymbol{P}_{PM0} = f(\boldsymbol{i}^m_{dq})$ are the PM losses, which are all evaluated with the reference electrical velocity $\omega_{re,0}$.

Eventually, utilizing (13) and (14) in combination with (8), the equivalent loss current component can be expressed as:

$$\boldsymbol{i}_{dq}^{Fe} = \frac{2}{3}\left(\boldsymbol{P}_{Fe} + \boldsymbol{P}_{PM}\right) \oslash \left(\omega_{re}\boldsymbol{J}\boldsymbol{\psi}_{dq}^{*}\right). \quad (15)$$

Accordingly, total stator current $\boldsymbol{i}_{dq}$ can be then calculated combining (6) and (15).

Knowing the total stator current after accounting for core losses, the copper losses can be computed using (7). However, to capture the influence of the AC component on copper losses, a one–dimensional LUT is utilized to model the stator resistance as a function of the electrical rotor speed. Therefore, (7) is modified as:

$$P_{Cu} = \frac{3}{2}\boldsymbol{R}_s|\boldsymbol{i}_{dq}|^2, \quad (16)$$

where $\boldsymbol{R}_s = f(\omega_{re})$ represents the speed–dependent stator resistance.

Using the LUT–based methodology described above, the high–fidelity PMSM models employed in this study are implemented within the SyR–e framework as follows:

1) CCG: As shown in Fig. 3, the three–phase voltage vector $\boldsymbol{u}_{abc}$ generated by the Converter block is applied to the CCG machine model. Each phase is represented by a variable stator resistor $R_s$ according to (16), a controlled current source and a large shunt resistor that aids numerical convergence. The resulting shunt voltage drops, which correspond to the phase back–EMFs, form the vector $\boldsymbol{e}_{abc}$. According to Fig. 4a, this vector is first mapped to the $\alpha\beta$ frame using the Clarke transformation and then integrated to obtain the flux vector $\boldsymbol{\psi}_{\alpha\beta}$. A subsequent Park transformation yields the flux linkage $\boldsymbol{\psi}_{dq}$, which together with the electrical angle $\theta_{re}$ addresses the inverse flux $dq\theta$ LUT, i.e. $\boldsymbol{i}_{dq}^{m} = f(\boldsymbol{\psi}_{dq}, \theta_{re})$, which is obtained by numerical inversion of (12). In conjunction with the electrical speed $\omega_{re}$, this magnetizing current $\boldsymbol{i}_{dq}^{m}$ feeds the iron and PM loss maps of (13) and (14). The resulting loss current $\boldsymbol{i}_{dq}^{Fe}$ is calculated via (15) and added to $\boldsymbol{i}_{dq}^{m}$ to obtain the total stator current $\boldsymbol{i}_{dq}$ according to (6). An inverse Park transformation then returns $\boldsymbol{i}_{dq}$ to the instantaneous current vector $\boldsymbol{i}_{abc}$ which is fed back to the controlled current sources. At the same time, the electromagnetic torque, which is obtained from the $dq\theta$ torque LUT as in (3), is transferred to the mechanical model of (4).

2) Simscape FEM–Parameterized PMSM: The Simscape FEM–Parameterized model of the PMSM is based on the corresponding Simscape Electrical block, which accepts the stator terminal voltages as inputs and returns the electromagnetic torque. The machine model thus corresponds to that of the physical device, whereby the current is measured at the same terminal where the voltage is applied, as shown in Fig. 4b, while the calculated torque is passed to a Simscape Mechanical block that captures the rotor dynamics. Although multiple levels of accuracy and modeling are available, SyR–e automatically instantiates the configuration shown in Fig. 4b which uses $dq\theta$ LUTs for the flux linkage as in (12), for electromagnetic torque as $\boldsymbol{T}_e(\boldsymbol{i}_{dq}^{m}, \theta_{re})$ and for iron/PM losses as in (13) and (14). Since the Simscape FEM–Parameterized block does not inherently support LUTs for AC copper losses, a speed–dependent variable resistance as in (16) was added in series with the stator windings to the original model created by SyR–e to account for these losses and to ensure consistence comparison with the other two models used in this study.

3) FM: The non–circuital FM of the PMSM is implemented by directly embedding the equations (1)–(3) into the SyR–e simulation framework, as shown in the block diagram in Fig. 4c, with addition of variable resistor as in (16). The input voltage $\boldsymbol{u}_{abc}$ is first transformed into the $dq$ reference frame, yielding $\boldsymbol{u}_{dq}$, which serves as input for the voltage equation (1). This equation is then integrated to obtain the state variable, i.e. the flux $\boldsymbol{\psi}_{dq}$, from which the electromagnetic torque and magnetizing current $\boldsymbol{i}_{dq}^{m}$ are calculated using corresponding $dq\theta$ LUTs, (3) and inverse of (12). As in the CCG model, the torque is passed to the mechanical dynamics block described by (4), while the magnetizing current $\boldsymbol{i}_{dq}^{m}$ along with the electrical speed $\omega_{re}$ is processed through the loss LUTs (13) and (14) to evaluate the total stator current $\boldsymbol{i}_{dq}$ as in (6) and (15). Finally, this current is then transformed back into the $abc$ frame.

## IV. Simulation

To ensure a consistent basis for comparison, each model within the SyR–e simulation framework, specifically the CCG, the FEM–Parameterized PMSM and the FM, was configured to represent the same interior PMSM, characterized by the following rated data:

PMSM: $P_n = 195$ kW, $T_n = 430$ Nm, $n_n = 4330$ rpm, $I_n = 565$ A,

DC-link: $U_{dc} = 400$ V,

while the corresponding LUTs used for modeling electromagnetic characteristics and losses are shown in Figs. 5 and 6, respectively.

For visualization purposes, the electromagnetic LUTs are only shown at a fixed electrical rotor angle $\theta_{re} = 0°$, although full $dq\theta$ LUTs are employed during the simulation. All electromagnetic data are computed assuming a PM temperature $\Theta = 80$ °C. In addition, the iron loss, which are composed of hysteresis and eddy current components, and the PM loss LUTs are shown for three representative operating speeds scaled by formulas defined in (13) and (14). To model the AC copper losses the resistance is treated as a function of both frequency and temperature. However, for the purpose of this study, a fixed temperature of $\Theta = 120$ °C is assumed so that only the frequency dependence is retained in the simulation model, according to (16).

### A. Results

Figure 7 presents the simulation results obtained from the CCG, Simscape FEM–Parameterized PMSM and

FM–based models, corresponding to the configurations illustrated in Figs. 3 and 4.

The default SyR–e torque control sequence used in all simulations is shown in Fig. 7a. Initially, half of the nominal torque setpoint, $T^* = 215$ Nm, is applied at standstill at $t = 0.15$ s. Subsequently, at $t = 0.2$ s the rotor speed is ramped up to the nominal value $n_r = 4330$ rpm, followed by a torque reversal at $t = 0.85$ s.

Accordingly, Figs. 7b and 7c show the electromagnetic torque and stator current of the PMSM for all three high–fidelity modeling approaches for both AVG and INST inverter configurations. Following the control sequence, the torque reference is processed by the Control block, which consequently generates the corresponding $d$ and $q$–axis current references.

In addition to the dynamic validation, the simulation performance of the three PMSM models was evaluated under both inverter modes. The evaluation, summarized in Fig. 8, includes three key performance metrics: elapsed simulation time, CPU utilization and RAM consumption, while all simulations were run on a workstation equipped with 32 GB RAM and a 12th Gen Intel(R) Core(TM) i7-1260P processor with 2.1 GHz, 12 cores and 16 logical threads.

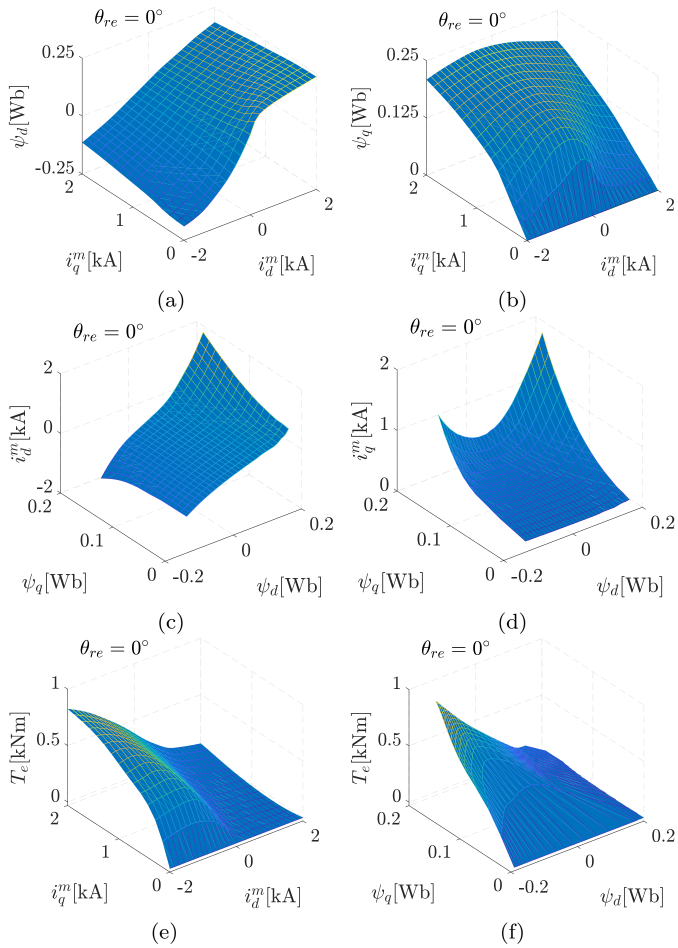


Fig. 5: LUTs utilized in the SyR–e simulation framework for modeling electromagnetic characteristics. (a) $\boldsymbol{\psi}_d(i_d^m, i_q^m, \theta_{re})$. (b) $\boldsymbol{\psi}_q(i_d^m, i_q^m, \theta_{re})$. (c) $\boldsymbol{i}_d^m(\psi_d, \psi_q, \theta_{re})$. (d) $\boldsymbol{i}_q^m(\psi_d, \psi_q, \theta_{re})$. (e) $\boldsymbol{T}_e(i_d^m, i_q^m, \theta_{re})$. (f) $\boldsymbol{T}_e(\psi_d, \psi_q, \theta_{re})$.

## B. Discussion on the results

Analyzing the simulation results in Figs. 7b and 7c, particularly the responses of the electromagnetic torque and the stator current, some general observations can be made that apply to all three PMSM modeling approaches. To ensure a consistent basis for comparison, the stator fluxes were excluded from the results and analysis, as the Simscape FEM–Parameterized block does not provide direct access to the flux variables during the simulation.

The first observation that applies to both the AVG and INST inverter configurations is the presence of increased ripple in both the torque and current waveforms during rotor speed ramp-up and steady state, i.e. after $t = 0.2$ s. This behavior, which is consistently observed across all models, results from the use of $dq\theta$ LUTs for electromagnetic characteristics that include rotor position as a variable. The spatial harmonics introduced by rotor angle variation are thus inherently captured, demonstrating one of a key advantage of $dq\theta$ LUT–based modeling, namely the ability to replicate more realistic machine behavior, including spatial non–idealities.

Furthermore, the comparison between AVG and INST inverter configurations reveals the expected differences in waveform smoothness. The INST model, which explicitly includes PWM, exhibits slightly higher ripple in the stator current and torque waveforms than the AVG model, where the ripple is naturally averaged.

However, despite these inverter–dependent differences, all three machine modeling approaches generally maintain stable behavior throughout the torque control sequence and achieve comparable dynamic performance in both inverter configurations. As expected for the interior PMSM topology under consideration, the $d$–axis current reference remains negative for all torque values, while the $q$–axis current reference follows the torque. For a positive torque setpoint at standstill, all models exhibit consistent dynamic behavior, i.e. after a short transient phase, the currents and torque converge to steady–state values and stable operation is maintained. Furthermore, when the torque is reversed at the nominal speed, the current and torque curves stabilize again for all models, retaining stability. Nevertheless, there is an additional noticeable discrepancy in the $d$–axis current response for the Simscape FEM–Parameterized model after a torque reversal compared to the other two models. This transition is short–lived, disappearing after $t = 0.86$ s, and may result from the use of different LUTs, as the Simscape FEM–Parameterized block requires direct flux LUTs rather than the inverse flux LUTs as used in the CCG and FM models. However, since the internal structure of the Simscape FEM–Parameterized

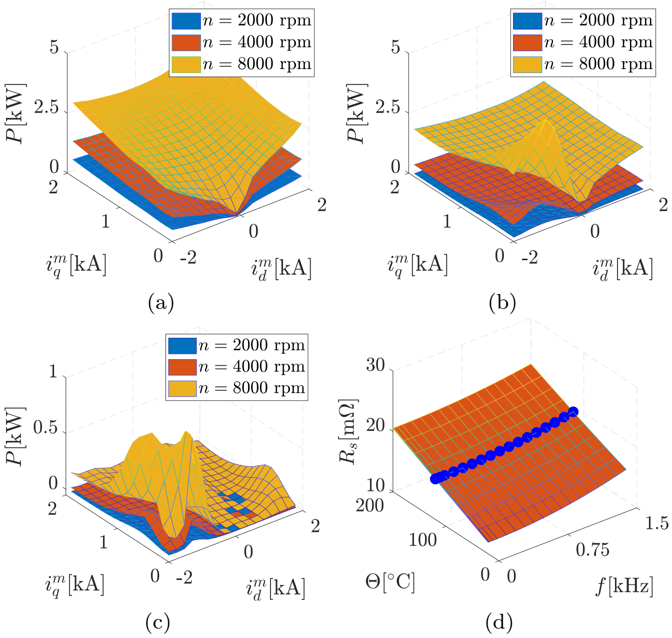


Fig. 6: LUTs utilized in the SyR–e simulation framework for loss modeling. (a) $\boldsymbol{P}_{h0}(i_d^m, i_q^m)$. (b) $\boldsymbol{P}_{ec0}(i_d^m, i_q^m)$. (c) $\boldsymbol{P}_{PM0}(i_d^m, i_q^m)$. (d) $\boldsymbol{R}_s(f, 120\ ^\circ\mathrm{C})$.

block is not accessible, it is difficult to determine the exact cause of this discrepancy.

Regarding the losses, the influence of iron, PM and AC copper losses on the torque and current dynamics was not directly analyzed in this study, as this topic has already been addressed in detail in [13], [14]. Instead, the present analysis focuses on how the inclusion of these losses affects the simulation performance, as shown in Fig. 8. In terms of simulation time, summarized in Fig. 8a, it can be seen that the simulations with the INST inverter configuration are consistently more time–consuming than their AVG counterparts, regardless of the PMSM modeling approach. Moreover, the inclusion of iron, PM and AC copper losses generally increases the simulation time further, regardless of the inverter mode, as additional LUT interpolations are required during execution. This effect is particularly pronounced in the INST configuration, where the inclusion of losses leads to a significant increase in simulation time, especially in the FEM–Parameterized and CCG approaches, with the latter showing an approximate 25-fold increase. On the other hand, the FM model shows the most consistent and on average lowest simulation time for both inverter types and all modeling approaches. In addition, the CPU utilization values shown in Fig. 8b demonstrate minimal fluctuations in all cases. The CPU utilization remains relatively constant at around 6.5%, indicating that the choice of PMSM modeling approach does not significantly affect the processor load. Similarly, the RAM consumption shown in Fig. 8c also remains stable across different models and configurations. Although the RAM usage more or less follows the pattern of time consumption from Fig. 8a, the average memory usage remains constant at around 20% in all cases, with no significant differences between the modeling approaches.

## V. Conclusion

The comparative analysis performed in this study shows that all three high–fidelity PMSM models, i.e. CCG, Simscape FEM–Parameterized and FM, provide consistent and comparable dynamic behavior of the PMSM. Despite the differences in internal structure, each model reliably reproduces the machine dynamics for both AVG and INST inverter modes. On the other hand, however, the results also reveal clear distinctions between the modeling approaches in terms of computational efficiency, modeling flexibility and practical application.

With regard to simulation performance, Tab. I summarizes that although all models exhibit similar CPU and RAM usage, the non–circuital FM approach is the fastest, with relatively constant simulation times across all scenarios, even when losses are modeled. In terms of modeling flexibility, all models accept machine–specific LUTs, but the Simscape FEM–Parameterized block has two major limitations: the variable phase resistance must be emulated with an external resistor and the stator flux variables are not accessible during simulation, limiting tasks such as flux observer validation or diagnostic studies. Moreover, even though the fault scenarios have not been explicitly tested, all models appear to be able to represent some basic fault conditions, such as phase–to–phase faults, but the circuital models are generally more versatile as they can represent both inverter– and machine–side open– and short–circuit conditions. In addition, the CCG model exposes its internal equivalent circuit, making it more suitable for detailed analysis than the Simscape FEM–Parameterized block, whose internal structure is not accessible. The explicit implementation of the inverter in the circuital models also enables a straightforward calculation of the switching and conduction losses, whereas the FM model describes the inverter analytically and requires additional sub–models for the loss calculation. However, the FM model has the advantage that it is independent of the Simscape Electrical toolbox, which reduces the software dependencies and costs for additional licenses.

Based on the results of this study, the non–circuital FM approach therefore seems to be preferable when computational speed and independence from the additional licenses are of primary importance. Conversely, circuital-based models, especially CCG, are better suited for detailed analysis, e.g. of fault conditions, inverter losses, etc. Overall, by evaluating these PMSM modeling approaches within a unified simulation framework, this study offers guidance for selecting the most appropriate strategy based on the

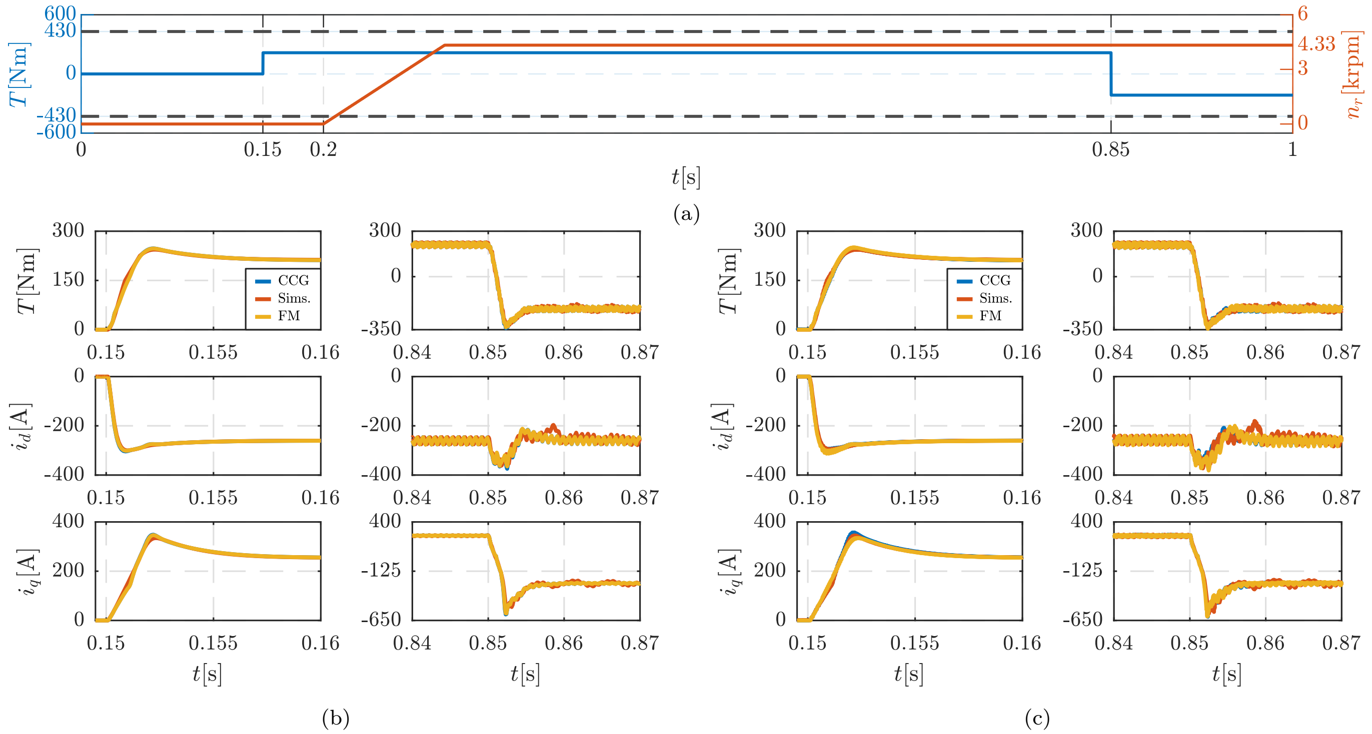


Fig. 7: Simulation results. (a) Default SyR–e torque control sequence. (b) Torque and currents with AVG mode. (c) Torque and currents with INST mode.

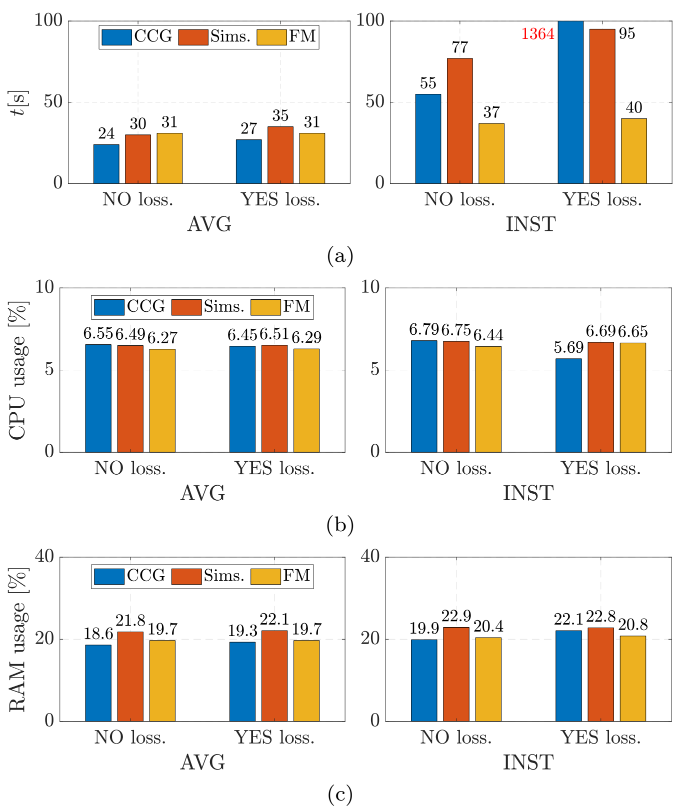


Fig. 8: Simulation performance. (a) Elapsed time. (b) CPU usage. (c) RAM usage.

TABLE I: Comparison of the high–fidelity modeling approaches

| | Circuital | | Non–circuital |
|---|---|---|---|
| | CCG | Simscape FEM-Param. | FM |
| **Simulation performance** | | | |
| Computational time | High | Medium | Low |
| CPU usage | Low | Low | Low |
| RAM usage | Medium | Medium | Medium |
| **Modeling** | | | |
| AC copper losses | ✓ | ✓/✗ | ✓ |
| Iron losses | ✓ | ✓ | ✓ |
| Flux linkage available | ✓ | ✗ | ✓ |
| Faults | ✓ | ✓/✗ | ✓/✗ |
| Inverter losses | ✓ | ✓ | ✗ |
| **Licenses** | | | |
| MATLAB/Simulink | ✓ | ✓ | ✓ |
| Simscape Electrical | ✓ | ✓ | ✗ |

desired balance between accuracy, computational efficiency and practical constraints.

## References

[1] A. T. de Almeida, F. J. T. E. Ferreira, and J. Fong, "Perspectives on electric motor market transformation for a net zero carbon economy," Energies, vol. 16, no. 3, 2023. [Online]. Available: https://www.mdpi.com/1996-1073/16/3/1248

[2] R. M. Llorente, Practical control of electric machines: Model-based design and simulation. Springer Nature, 2020.

[3] C. Liu, "Emerging electric machines and drives—an overview," IEEE Transactions on Energy Conversion, vol. 33, no. 4, pp. 2270–2280, 2018.

[4] C. Liu, K. Chau, C. H. Lee, and Z. Song, "A critical review of advanced electric machines and control strategies for electric vehicles," Proceedings of the IEEE, vol. 109, no. 6, pp. 1004–1028, 2020.

[5] M. Pizzuto, A. Balamurali, H. Mohammadi, and N. C. Kar, "Impact of model fidelity on control and performance analysis of interior permanent magnet synchronous machines," in 2024 International Conference on Electrical Machines (ICEM). IEEE, 2024, pp. 1–6.
[6] M. Wang, "A novel mathematical nonlinear pmsm realization method for electric machine emulator," IEEE Journal of Emerging and Selected Topics in Power Electronics, vol. 10, no. 4, pp. 4171–4181, 2022.
[7] X. Chen, J. Wang, B. Sen, P. Lazari, and T. Sun, "A high-fidelity and computationally efficient model for interior permanent-magnet machines considering the magnetic saturation, spatial harmonics, and iron loss effect," IEEE Transactions on Industrial Electronics, vol. 62, no. 7, pp. 4044–4055, 2015.
[8] C. Kukura, J. Apsley, and S. Djurovic, "Computational features of electric machine hybrid fea-dq dynamic models," IEEE Access, 2024.
[9] A. Di Tommaso, R. Miceli, C. Nevoloso, G. Scaglione, and G. Schettino, "Improved high-fidelity ipmsm mathematical model including saturation, cross-coupling, torque ripple and iron loss effects," in 2022 International Conference on Electrical Machines (ICEM). IEEE, 2022, pp. 21–27.
[10] X. Chen, J. Wang, and A. Griffo, "A high-fidelity and computationally efficient electrothermally coupled model for interior permanent-magnet machines in electric vehicle traction applications," IEEE Transactions on Transportation Electrification, vol. 1, no. 4, pp. 336–347, 2015.
[11] S. Li, D. Han, and B. Sarlioglu, "Modeling of interior permanent magnet machine considering saturation, cross coupling, spatial harmonics, and temperature effects," IEEE Transactions on Transportation Electrification, vol. 3, no. 3, pp. 682–693, 2017.
[12] H. Cai and D. Hu, "On PMSM model fidelity and its implementation in simulation," in 2018 IEEE Energy Conversion Congress and Exposition (ECCE). IEEE, 2018, pp. 1674–1681.
[13] A. Bojoi, S. Ferrari, P. Pescetto, and G. Pellegrino, "Advanced circuital model for e-drive simulation, including harmonic effects and fault scenarios," in PCIM Europe 2023; International Exhibition and Conference for Power Electronics, Intelligent Motion, Renewable Energy and Energy Management. VDE, 2023, pp. 1–10.
[14] S. Ferrari, P. Ragazzo, G. Dilevrano, and G. Pellegrino, "Flux and loss map based evaluation of the efficiency map of synchronous machines," IEEE Transactions on Industry Applications, vol. 59, no. 2, pp. 1500–1509, 2022.
[15] The MathWorks Inc., "Fem-parameterized pmsm," Natick, Massachusetts, United States, 2025. [Online]. Available: https://www.mathworks.com/help/sps/ref/femparameterizedpmsm.html
[16] Plexim GmbH, "Plecs," Zurich, Switzerland, 2025. [Online]. Available: https://www.plexim.com/products/plecs
[17] A. Varatharajan, D. Brunelli, S. Ferrari, P. Pescetto, and G. Pellegrino, "syredrive: Automated sensorless control code generation for synchronous reluctance motor drives," in 2021 IEEE Workshop on Electrical Machines Design, Control and Diagnosis (WEMDCD). IEEE, 2021, pp. 192–197.
[18] "SyR-e: Synchronous Reluctance - evolution." [Online]. Available: www.github.com/SyR-e
[19] S. Ferrari, G. Dilevrano, P. Ragazzo, and G. Pellegrino, "The dq-theta flux map model of synchronous machines," in 2021 IEEE Energy Conversion Congress and Exposition (ECCE). IEEE, 2021, pp. 3716–3723.