

We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

5,500

Open access books available

136,000

International authors and editors

170M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com



Chapter

Powerful Multilevel Simulation Tool for HiL Analysis of Urban Electric vehicle's Propulsion Systems

Raul Octavian Nemeş, Mircea Ruba, Sorina Maria Ciornei and Raluca Maria Raia

Abstract

The general focus of the proposed chapter is to describe a complex yet transparent solution for advanced simulation analysis of urban electric vehicles propulsion unit. As general rule, precise and realistic results are obtained only when performing real-time simulations, engaging dedicated software for such applications. Hence, simulation of an electric vehicle as a complete solution can become rather difficult. The authors targeted advanced analysis of the propulsion unit, including the motor, the battery, the power converter, and its control. These are designed using multilevel models in Matlab/Simulink, referring to different complexity levels of each assembly. Another feature of the models is their organization, based on Energetic Macroscopic Representation (EMR), this easing the process of inter-connecting models correctly. Nevertheless, the mechanical, aerodynamical and road profile details are included using Amesim Software. All the simulations are performed on a real-time target, using a National Instruments PXIe embedded controller. The latter runs NI VeriStand software, allowing real-time communication between Amesim and Simulink offering in the same time possibility to read/write analog/digital IOs for external communication. This feature in fact is used when passing from modeling to Hardware in the Loop (HiL) analysis, replacing the simulated assembly with the actual one.

Keywords: real-time, Hardware in the Loop, EMR, multilevel, electric vehicle

1. Introduction

Electrification of urban transportation services, from high power solutions, such as busses and trains, to low power ones, such as small cargo delivery vehicles became highly important. The pressure of developing such solutions is continuously increasing due to the highly crowded urban scenarios in which merchandise and goods must be delivered as fast as possible. Investigating, researching and creating such solutions challenges many universities and OEM engineers. Nowadays, taking advantage of the achievements in the field of computer aided design, any serious study requires advanced simulations, fusing measured data, estimated

and computed ones into powerful computation models, able to replicate behavior of the actual hardware. It is important to mention that at a certain level of complexity, studies can become rather difficult by means of large required computation time, powerful computers to perform and many model's implementations time.

In order to achieve high precision and accurate phenomenon simulation, one must use several software interconnected, each performing analysis in their field of expertise. For example, when simulation of an urban electric vehicle (UEV), all the mechanical, aerodynamical and road conditioning factors can be simulated in Amesim Software, while all the power electronics, electrical drive assemblies and their advanced control can be simulated in Matlab/Simulink. Such a solution ensures high fidelity and correct analysis approach of each assembly of the UEV. However, a co-simulation of this magnitude becomes rather complicated, time and resource demanding, returning a non-real-time behavioral screening. Hence, one solution is to engage a third software, National Instruments VeriStand. This was created to integrate exported model from many simulation software and create one complex simulation solution. Another hands-on advantage is that using a PXIe embedded controller, VeriStand can upload the model and perform the simulation in real-time, allowing the user to observe the system behavior just like in the real life. The simulation time can be from seconds elongated to tens of hours. The data recording can be streamed and decimated as desired. Such a solution will be presented in this chapter.

Another important feature of this tool is given by the real-time target's nature. This allows external communication via analog and digital I/O s simplifying and facilitating the transition from model-based analysis to hardware in the loop (HIL) based analysis.

In order to offer the possibility of selecting the accuracy and the magnitude of the model, the authors considered developing multilevel complexity model, all organized in Energetic Macroscopic Representation (EMR). The latter ensures the user with the correct understanding of the system's power flow, creates a direct path when designing the control loops and facilitates the model exchange when passing from one level of complexity to another. It has to be mentioned that all the models presented in the following sections are designed in EMR philosophy. However, the interest of this chapter is not focused towards this special organization method, hence the reader is encouraged to lecture references [1–3].

2. Multilevel modeling of electric vehicle's propulsion unit

The multilevel approach for developing simulation models for an electric vehicle's propulsion system must be broken down into its main assemblies: the power supply (batteries), the electronic converter, the electrical machine and the general system controller (electronic control unit-ECU). To have a comprehensive analysis, each of these assemblies will be addressed separately in the following sub-sections. For each model, the corresponding energetic macroscopic representation (EMR) block will be presented, separately from its content. The latter will be in fact the modeling approach of each assembly. EMR is an energy flow based graphical organization philosophy easing the building process of complex electromechanical systems. The connection between blocks is always based on action-reaction principle according to the physical causality. In **Figure 1**, the basic elements used in EMR organization are depicted. These will be used in the following sections as some assemblies, for example, an electrical machine will include more than one such block in their graphical description.

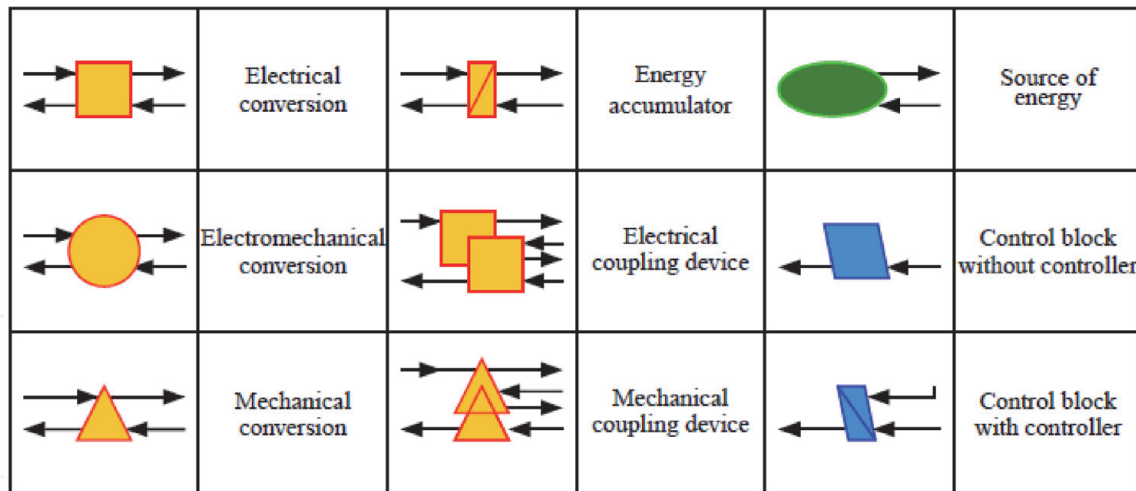


Figure 1.
Main EMR pictograms.

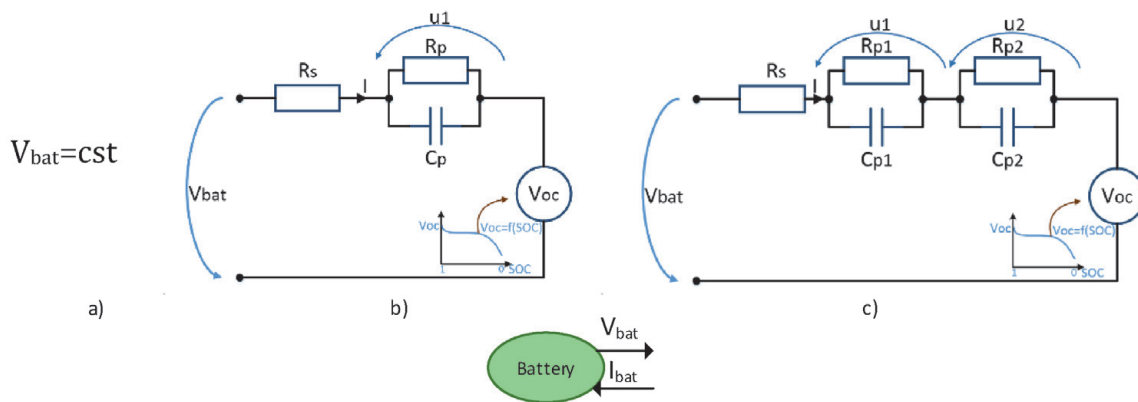


Figure 2.
First (a) and second (b) order equivalent battery circuits and their EMR pictogram (c).

Before diving in the presentation of the multilevel models, it is lucrative to explain the philosophy of building a control scheme using EMR. This is necessary because, in the present chapter, the control units will be also under discussion and their multilevel attribute will come from the level of compromises chosen when building its EMR representation. The control loop can be simply generated by inverting all the model blocks. Hence, the blue pictograms (the control) are a reversed model of the orange ones (the model). The main difference is that in the control loop, only the action path is reversed, while the reaction is neglected, as the latter one is necessary only for the model description and not for the control. In the above indicated references, more details about building EMR control loops are presented, hence the reader is encouraged to visit those.

2.1 Multilevel modeling of batteries

When it comes to model a power supply for an electric vehicle, one can choose between ideal or non-ideal approaches. The latter category can be divided into less or more complex mathematical description. In general, besides the ideal approach the non-ideal one include first and second order battery modeling [4, 5]. A source or a sink in EMR is represented by a green oval, outputting voltage, and inputting current or vice versa.

In **Figure 2** the pictogram used for the battery as electrical source in a simulation is depicted. The content of this block however, can be adjusted to the designer's

needs. In **Figure 2(a)–(c)**, the ideal battery model is depicted (where the voltage is imposed invariant of any changes in the system), the first and second order models (where the battery voltage is dependent on the sourced/sinked current and the state of charge (SOC)). The latter models are called first and second order ones due to the number of parallel RC branches that compound their circuit.

$$V_{bat} = V_{oc} - u_1 - R_s I_{bat}; \quad \frac{du_1}{dt} = -\frac{1}{R_p C_p} u_1 + \frac{1}{C_p} I_{bat} \quad (1)$$

The mathematical translation of the circuits is detailed in Eq. (1) and Eq. (2). It is important to mention that the circuits parameter's values for such models are identified using laboratory experiments and actual battery cells. Such approaches were carried out by the authors on a LG (LGABD11865) battery with a rated capacity of 3000 mAh, 3.75 V rated, 4.2 V maximum over charge voltage, 2.7 V minimum discharge voltage, 0.5–1 A charging current and 0.2–0.5 A discharging current.

$$\begin{aligned} V_{bat} &= V_{oc} - u_1 - u_2 - R_s I_{bat} \\ \frac{du_1}{dt} &= -\frac{1}{R_1 C_1} u_1 + \frac{1}{C_1} I_{bat} \\ \frac{du_2}{dt} &= -\frac{1}{R_2 C_2} u_2 + \frac{1}{C_2} I_{bat} \end{aligned} \quad (2)$$

The complete process described step-by-step to identify the parameters of the battery cell needed to run the correct model is presented in detail in [4]. Even more, not only the experimental process itself is presented, but also very important post processing steps mandatory to reach high accuracy results.

The open circuit voltage (V_{OC}) is identical for both models, being measured for the same conditions. The recorded function of the SOC is depicted in **Figure 3**.

To help the reader emphasize an image about the precision of these models, a comparative analysis was carried out, superimposing their results for the same testing conditions. After the parameter's identification process [5–8], both models

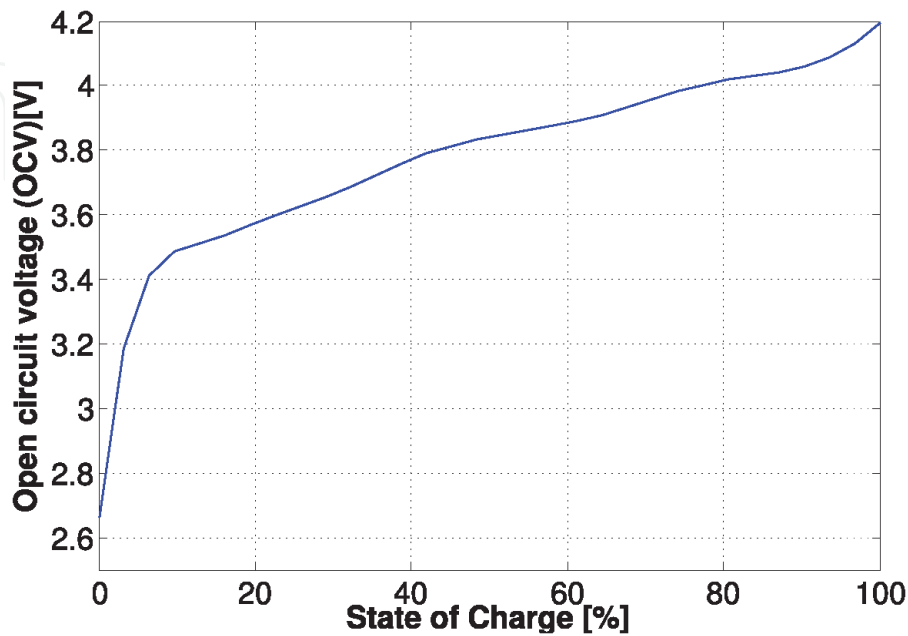


Figure 3.
The open circuit cell voltage vs. SOC.

presented in Eq. (1) and Eq. (2) were built in Matlab/Simulink, as depicted in **Figure 4**-left.

The considered test was to apply equal length current pulses with equal length relaxation time between them, discharging the cell from 100–10%. Initially, this test was applied to the actual cell, the results being also used to identify the necessary data for the simulation models. Imposing the same current variation to the two models, and comparing their voltage calculations with the measured one, one can conclude that the second order model reaches higher accuracy than the first order one. In the zoomed plot from **Figure 4**-right, it can be observed that the second order and measured voltages are in best agreement, while the first order's one marks a slight deviation.

To quantify this difference to a certain palpable value, the root mean square error (RMSE) is computed using Eq. (3) and the instantaneous error between the two signals.

$$RMSE = \sqrt{\frac{1}{T} \cdot \int_0^T [V_{measured}(t) - V_{model}(t)]^2 dt} \quad (3)$$

The comparative results of the RMSE and the instantaneous error are depicted in **Figure 5** (left and right). In both cases one can observe that the second order model yields lower disturbance compared to the first order one. More, in **Figure 5**-left

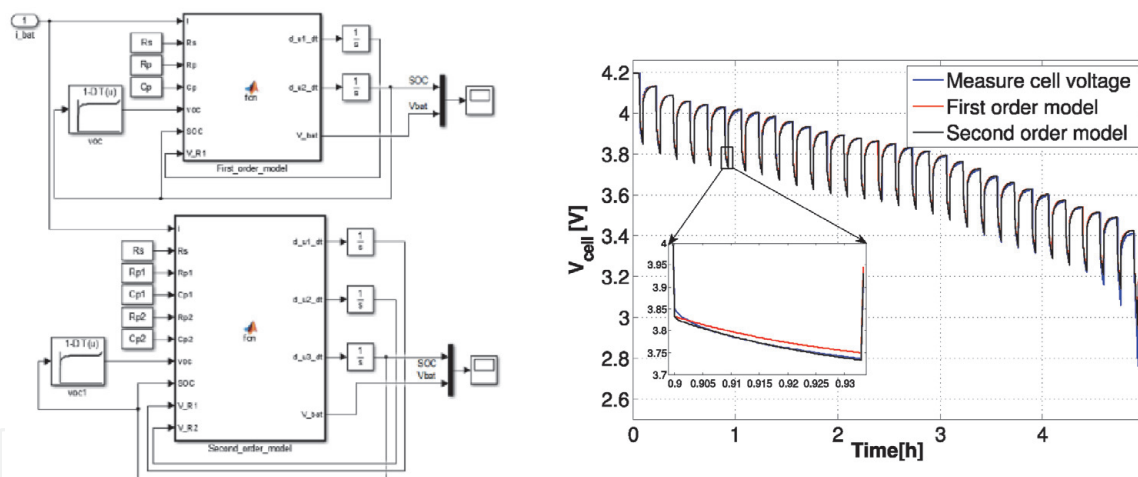


Figure 4. The Matlab/Simulink cell models (left) and the comparative analysis (right).

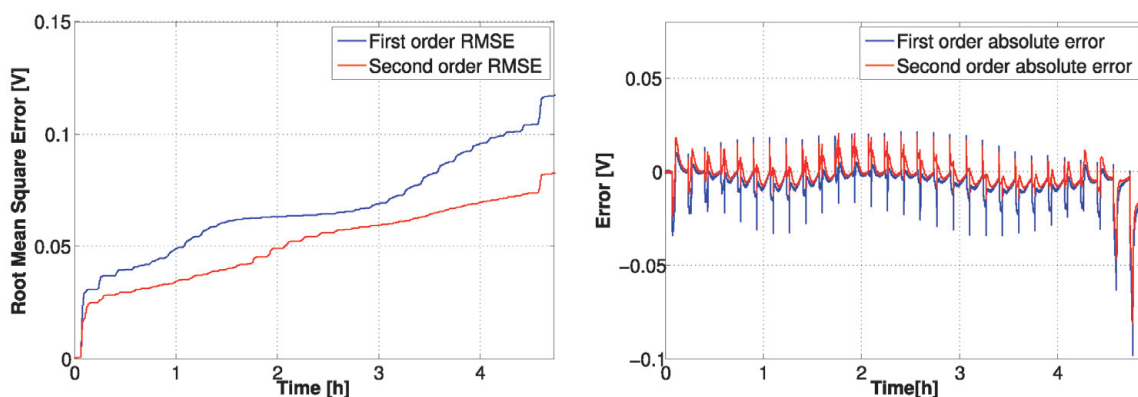


Figure 5. The RMSE (left) and the instantaneous error (right) of the battery models.

shows that during the ongoing discharging process, the RMSE for the first order model increases gradually faster than the one of the second order one.

To conclude this section, one can understand that increasing the rank of the battery model, the gap between real and simulated results is mitigated. The cost of this is the increased complexity of models, time-consuming parameters identification and the necessity of more data post processing operations. On the contrary, the first order model is quite simple to design, its parameters are simple to identify with classical calculations and there is no need for values post processing. If the compromise of accepting a small error in the battery model with the gain of simplicity, the first order model is the hands-on solution.

2.2 Multilevel modeling of power electronics' losses

The general approach when dealing simulation approaches of electronic converters regard conventional ideal ones, with or without switching, so called smooth or switched models. However, none of these consider the losses that occur in a non-ideal converter. The largest losses are due to the switching components (power transistors and/or diodes), while the rest of the losses inside a converter can be neglected (those on the driver and measuring circuits). One reason why the losses are generally neglected overall is because their actual measurement is rather difficult and requires dedicated equipment. In doing so, one simple method is to consider a general efficiency of the converter and add it into its calculations. So, already some straightforward solutions for modeling electronic converters were mentioned, solutions that are available in nearly any reference the reader would search. To add value to the research and to the present chapter, the authors introduce 2 different approaches to compute the losses that occur on one MOSFET transistor. Of course, these can be then extrapolated towards the total number of power switches.

In **Figure 6**-bottom the EMR pictogram used for any electronic converter is presented. It inputs the DC link voltage (V_{dc}), the load currents (i_{load}) and the duty cycle (m_s). After the internal calculations, the converter outputs the resulting voltage (u) to be supplied to the load and its output currents (i_{conv}). The inputs and outputs do not change with the chosen method of internal calculations. Either is

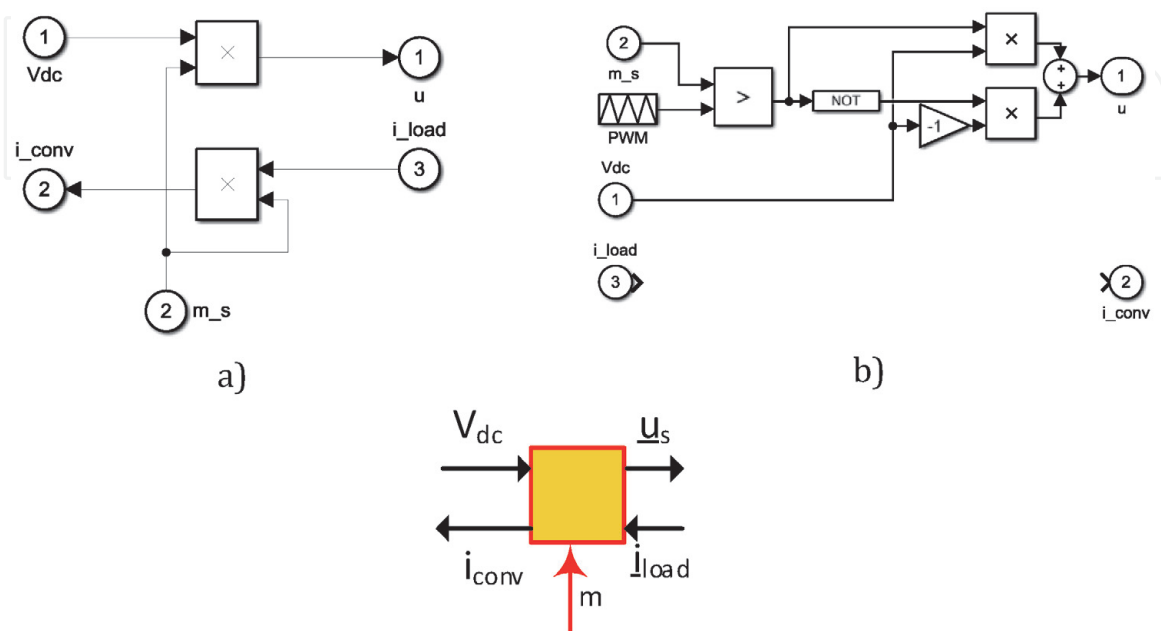


Figure 6. Smooth (a) and switched (b) converter architectures and their EMR pictogram (c).

a smooth or switched model, these remain the same and valid even if we consider DC or AC converters.

In addition to the above depicted models, this block can include also the calculus for the converter losses. Even if in the standard pictogram there are no input/output ports considered for this, the information, being additional, can be manipulated to/from the block via wireless labels.

Taking into consideration of these losses is mandatory when one is interested in reaching optimal converter architectures [9, 10]. The following calculus can be adapted to any power transistor, the authors using the IRFP4668PBF MOSFET, having its specifications listed in **Table 1**.

The losses computed in any active switching electronic component can be broken down into: conduction losses, switching losses and blocking (leakage) losses, the later normally being neglected, as expressed in Eq. (4).

$$P_{lossT} = P_{conductionT} + P_{switchingT} \quad (4)$$

As mentioned before, two approaches will be presented in detail. The first one computes the power losses during conduction and switching the MOSFET while the second one computes losses during switch-on and switch-off transients.

In the first approach, the conduction and switching losses are computed based on Eqs. (5) and (6), with regards to the semiconductor's parameters and its operation. The latter is described via the duty cycle (D), the switching frequency (f_{sw}) and the transistor's output capacitance (C_{oss})

Parameter	Symbol	Value	Unit
Forward voltage	V	120	V
Drain to source resistance	Rds	0.008	Ω
Switching Frequency	fsw	10000	Hz
Output Capacitance	Coss	810e-12	F
Turn off time	Toff	64e-9	s
Turn on time	Ton	41e-9	s
Forward voltage drop	Vf	1.3	V
Reverse Leakage current	Ir	0.00025	A
Reverse Recovery Time	Trr	135e-9	s
Max Recovery Current	Irrm	8.7	A
Gate Resistance	Rg	11	Ω
Plateau Voltage	Uplateau	6.1	V
Gate Capacitance	CGD1	200e-12	F
Gate Capacitance	CGD2	1200e-12	F
Reverse recovery charge	Qrr	700e-9	C
Rise Time	tr	105e-9	s
Fall time	tf	74e-9	s
Reverse Recovery Current	IDoff	8.7	A
Average diode current	IFav	20	A
Diode on-state zero-current voltage	ud0	0.5	V

Table 1.
 The main specifications for IRFP4668PbF MOSFET.

$$P_{conductionT} = I_d^2 R_{ds} D \quad (5)$$

$$P_{switchingT} = \frac{VI_d}{2} (T_{on} + T_{off}) f_{sw} + C_{oss} V^2 f_{sw} \quad (6)$$

On the other hand, the losses occurring in the diode attached to the transistor are broken down into conduction, reverse and switching ones. The total losses due to the diode's operation are:

$$P_{lossD} = P_{conductionD} + P_{reverseD} + P_{switchingD} \quad (7)$$

Each of the eq.'s (7) components are formulated in expressions (8)–(10), with parameters valued and explained in **Table 1**:

$$P_{conduction} = I_f V_f D \quad (8)$$

$$P_{reverse} = I_r V (1 - D) \quad (9)$$

$$P_{switching} = \frac{T_{rr} I_{rrm} V_r f_{sw}}{2} \quad (10)$$

This model is quite simple and represents the hands-on solution for quantifying the converter losses without engaging too many mathematical tools.

The more complex model, the conduction losses are computed function of the on-state resistance (R_{ds}) and the current passing through the component (i_d):

$$P_{cond} = R_{ds}^2 i_d \quad (11)$$

The next component of the total losses, the switching one, is broken down into two states, according to the on and off transients. There are different opinions and mathematical interpretations on how to compute the voltage fall-time (tfu) to gain accuracy. Details are presented in Ref. [10]. Engaging a straight forward approach, the fall-time is considered to be composed of two parts in accordance with the variation of the gate capacitance:

$$tfu = \frac{tfu1 + tfu2}{2}$$

$$tfu1 = (U_{DD} - R_{DS} I_d) R_G \frac{C_{GD1}}{U_{drv} - U_{plateau}} \quad (12)$$

$$tfu2 = (U_{DD} - R_{DS} I_d) R_G \frac{C_{GD2}}{U_{drv} - U_{plateau}}$$

In Eq. (12), the terms (U_{drv}) and (U_{DD}) refer to the driver voltage that excites the transistor's gate and the converter's DC link voltage, respectively. The rest of the terms are to be found in **Table 1**.

The complementary model to compute the switch-off losses differ just by subtracting from Eq. (12) the driver voltage.

$$tru = \frac{tru1 + tru2}{2}$$

$$tru1 = (U_{DD} - R_{DS} I_d) R_G \frac{C_{GD1}}{U_{plateau}} \quad (13)$$

$$tru2 = (U_{DD} - R_{DS} I_d) R_G \frac{C_{GD2}}{U_{plateau}}$$

The required energy to turn on and off the transistor is calculated in Eq. (14) and (15).

$$E_{onT} = U_{DD}I_{Don} \frac{tru + tfu}{2} + Q_{rr}U_{DD} \quad (14)$$

$$E_{offT} = U_{DD}I_{Doff} \frac{tru + tf}{2} \quad (15)$$

With the above computed ingredients, one can calculate the power losses on the transistor, Eq. (16), and diode, Eq. (17).

$$P_{swT} = (E_{onT} + E_{offT})f_{sw} \quad (16)$$

$$P_{swD} = (E_{onD} + E_{offD})f_{sw} \approx E_{onD}f_{sw} \quad (17)$$

The losses in the MOSFET and the free-wheeling diode are computed as the sum of the conduction and switching losses.

$$P_{TotalT} = P_{conductionT} + P_{switchingT}$$

$$P_{TotalT} = R_{DSon}I_{Drms}^2 + (E_{onM} + E_{offM})f_{sw} \quad (18)$$

$$P_{TotalD} = P_{ConductionD} + P_{switchingD}$$

$$P_{TotalD} = u_{D0}I_{Fav} + R_D I_{Frms}^2 + E_{onD}f_{sw} \quad (19)$$

It has to be mentioned that during operation, in one transistor or diode, the currents can be positive or negative. In an analytical approach, this would lead to decreased losses, hence, to correctly calculate those, the absolute current's value is used.

Comparing the results of the two approaches (**Figure 7**) one can conclude that the complex model reaches higher calculated values. This is more realistic as it contains more loss sources compared to the simple model. However, the simple model proves to offer a quite good quantification of the losses despite its ease in approach, reaching satisfactory results.

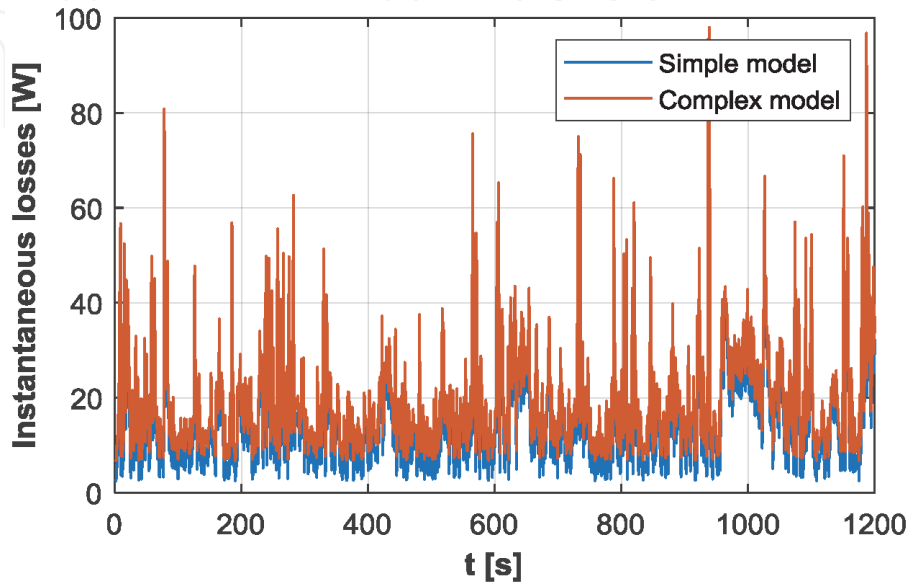


Figure 7.
 The instantaneous converter losses for the simple and complex models.

2.3 Multilevel modeling of permanent magnet synchronous machines

The general approach when modeling a permanent magnet synchronous machine (PMSM) is to use its mathematical representation in dq0 reference frame, fixed to the rotor.

$$\begin{aligned} v_{sd} - R_s i_{sd} - e_{sd} &= \frac{d}{dt} \psi_{ds} \\ v_{sq} - R_s i_{sq} - e_{sq} &= \frac{d}{dt} \psi_{qs} \end{aligned} \quad (20)$$

$$\begin{aligned} T_{pmsm} &= \frac{3}{2} p (\Psi_{ds} i_{qs} - \Psi_{qs} i_{ds}) \\ e_{sd} &= -\omega \Psi_{qs}, e_{sq} = \omega \Psi_{ds} \\ \Psi_{ds} &= \Psi_{md} + L_d I_{ds} + L_{dq} I_{sq} \\ \Psi_{qs} &= \Psi_{mqd} + L_q I_{qs} + L_{qd} I_{ds} \end{aligned} \quad (21)$$

$$J \frac{d\Omega}{dt} = T_{pmsm} - T_{load} - B\Omega \quad (22)$$

The variables in Eqs. (20) and (21) are: the phase resistance (R_s), the direct and quadrature inductances (L_d and L_q) and the cross-coupling inductances (L_{dq} and L_{qd}). The number of pole pairs is denoted with (p), while the rotational speed is marked by (ω) and the friction coefficient in marked by (B). The dq currents (I_{ds} and I_{qs}) are computed by integration of the voltage expressions.

For a PMSM, the EMR model is constructed by coupling 3 pictograms, one accumulation element (for the windings) described by Eq. (20), one electromechanical conversion (for the torque generation), described by Eq. (21) and another accumulation element (for the mechanical equilibrium), described by Eq. (22), (visit **Table 1** for clarifications). Such a model is depicted in **Figure 8**.

Returning to Eq. (21), the dq magnetic flux equations give the opportunity to choose the level of modeling complexity. In this section, 3 different such levels will be detailed: based on constant dq inductances, based on variable dq inductances and a flux-linkage based model. The latter is the most complex and precise one.

To have a comprehensive analysis, like for the battery and the power inverter, for the PMSM the model's parameters are listed in **Table 2**.

The first level of complexity consists in using constant values for the d and q axes inductances. The identification of these is a very simple process. It can be carried out in finite element analysis (FEA) or on the real machine (or in both environments). It requires only fixing the rotor in 0 and 90 electrical degrees positions and applying in both one controlled voltage pulse. The time dependent slope of the current will return after some simple calculations the global machine inductance (L). Having this value, the d and q components are easy to be computed,

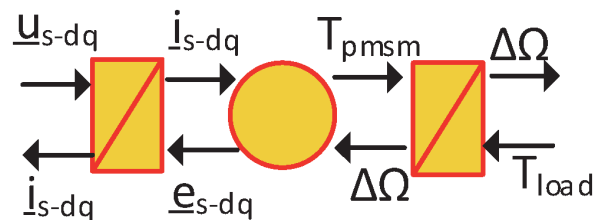


Figure 8.
The EMR for a PMSM.

Electric and geometric parameters			
	Dimension	Value	Unit
1	Pole pairs	3	—
2	Rotor speed	2000	[rot/min]
3	Rated torque	12	[Nm]
4	Rated current per phase	21	[A]
5	Mechanical power	2500	[W]
6	DC Voltage link	120	[V]

Table 2.
 The PMSM under analysis.

using Eq. (23). Details about the identification process can be selected from references [11, 12].

$$L_d = \frac{2}{3}L \quad (\theta_{el} = 0^\circ), L_q = \frac{2}{3}L \quad (\theta_{el} = 90^\circ) \quad (23)$$

The second level model regards detailed information about the machine under testing. The latter needs to be simulated in a FEA environment in order to compute the dq inductance's variations function of the machine current. This is generally ranged from $\pm 10 \times I_n$, where I_n is the machine's rated current. The method itself that allows preliminary identification of the 3-phase magnetic fluxes is the so called "frozen permeability" approach. Applying it demands to set to 0 the permanent magnet's remanent flux density, freezing its permeability. More details about this method are to be found in Ref.s [12, 13]. The resulted phase magnetic fluxes will be used to compute the dq inductances versus the machine current, applying Eq. (24).

$$\Psi_d = \frac{2}{3} \left(\Psi_a - \frac{1}{2}\Psi_b - \frac{1}{2}\Psi_c \right), \Psi_q = \frac{1}{\sqrt{3}}(\Psi_b - \Psi_c).$$

$$L_d = \frac{\Psi_d}{i_d}, L_{qd} = \frac{\Psi_q}{i_d}, L_q = \frac{\Psi_q}{i_q}, L_{dq} = \frac{\Psi_d}{i_q} \quad (24)$$

The outcome of these calculations is stored in look up tables (LUT) and introduced in the machine model. It is clear that such a procedure is impossible to be implemented in a real testbench or it would require complex and expensive hardware. One can observe that using the calculus from Eq. (23), the direct as well as the cross-coupling inductances are computed. Their variation function of the ranged current is depicted in **Figure 9**.

The last and altogether the most complex model is based on using instead of inductances, the variation of the d and q currents function of the d and q magnetic fluxes. These are 2D LUTs as depicted in **Figure 10**.

It is important to mention that building the 2D LUTs from **Figure 10** is a quite complex and challenging process. Here to the designer needs to have the FEA model of the machine under study. Several simulations need to be accomplished, fetching initially the $\Psi_d(i_d, i_q)$ and $\Psi_q(i_d, i_q)$. From this information mathematical inversion methods need to be applied to get $i_d(\Psi_d, \Psi_q)$ and $i_q(\Psi_d, \Psi_q)$. The complete process is explained in detail in [14].

In **Figure 11** a simple comparative analysis of the results obtained when simulating all the 3 models for the same conditions are depicted. Focusing the attention on the variation of the q axis magnetic flux, it can be noticed that flux linkage and

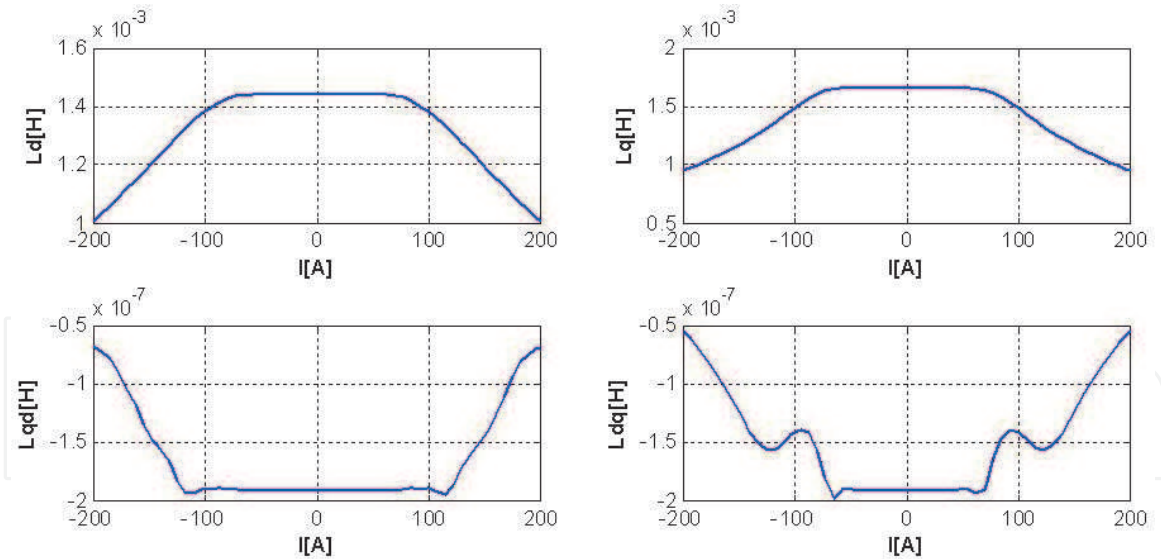


Figure 9.
The direct and cross-coupling inductances calculated with frozen permeabilities.

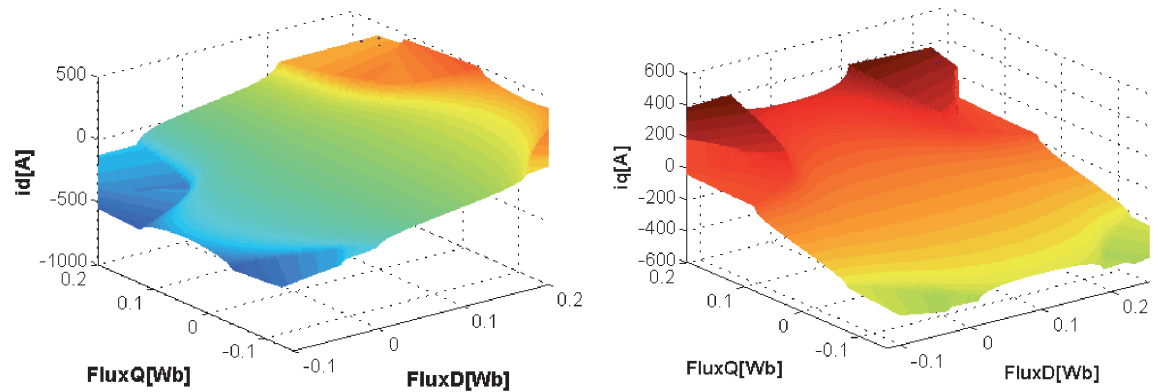


Figure 10.
The 2D LUTs of $i_d(\Psi_d, \Psi_q)$ and $i_q(\Psi_d, \Psi_q)$.

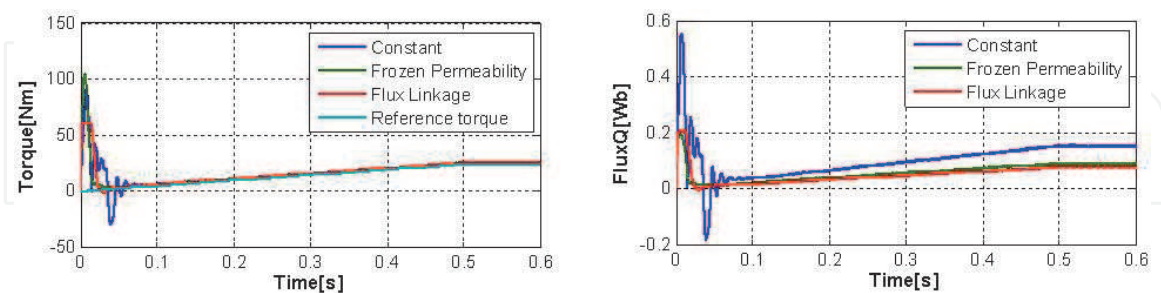


Figure 11.
The PMSM torque and Ψ_q during benchmarking the simulation levels.

frozen permeability models do saturate, while the one based on constant inductances reaches large values. The latter is ideal; hence the machine will never saturate. In this condition, one can perform simulations within rated operation. However, when attempting to go in overload region the results will not reflect the real machine operation.

The flux linkage and frozen permeabilities models are behaving quite in the same manner. However, the complexity of the flux linkage model is high, both in data post processing and design. The ultimate conclusion is that using the frozen

permeabilities model will lead to quite satisfactory results reflecting closely the actual machine behavior.

2.4 Designing the control loop

Energetic Macroscopic Representation is one organization concept that besides helping the designer to build simulation programs that reflect the actual power flow of a real testbench, it also proves to be a very lucrative solution when it comes to building control loops. To do so, one has just to inverse the simulation model and keep only the action path, discarding the reaction one.

In **Figure 12** the control loop can be observed being highlighted in light blue. Its inversed character compared to the model (in orange) is noticeable and each block contains the mathematical description of the inversed action.

The reversed accumulation block contains a regulator that outputs the reference torque to be developed function of the error between the actual and the imposed speed (see Eq. (25)).

$$\xi_{n_{PMSM}} = n_{PMSM_ref} - n_{PMSM}$$

$$\xi_{n_{PMSM}} \rightarrow PI(k_{p_n}, k_{i_n}) \rightarrow T_{PMSM} \quad (25)$$

The inversed block without controller computes the reference q axis current function of the reference torque above expressed. In the same time, the d axis current is imposed to 0, this for torque maximization purposes.

$$i_{sq_ref} = \frac{T_{PMSM}}{\frac{3}{2}p [\Psi_{PM} + (L_d - L_q)i_{sd}]}$$

$$i_{sd_ref} = 0 \quad (26)$$

This reference d and q currents are compared with the actual ones. The resulting errors are sent to regulators that output the reference d and q voltages. This process is carried out in the third control block (see **Figure 12** from right to left).

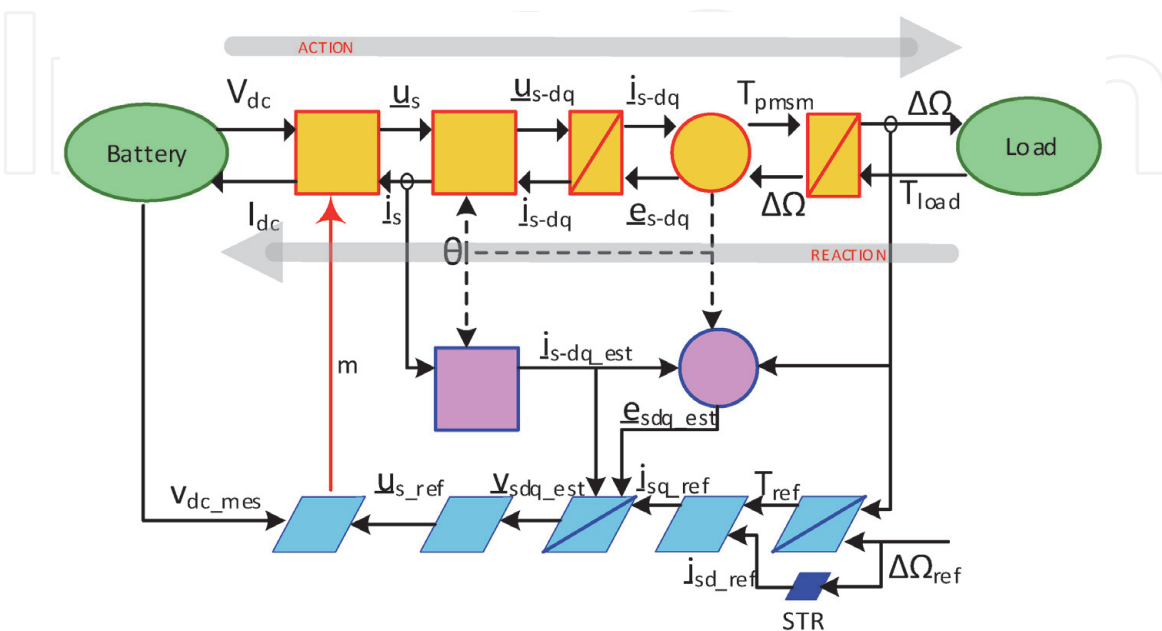


Figure 12.
 The complete PMSM simulation model in EMR organization.

$$\begin{aligned} \xi_{iqd} &= i_{sqd_ref} - i_{sqd} \\ \xi_{iqd} &\rightarrow PI(k_{p_i}, k_{i_i}) \rightarrow u_{sqd} \\ u_{sqd_est} &= u_{sqd} + e_{sqd_est} \end{aligned} \tag{27}$$

The resulted d and q voltage references are converter to 3-phase quantities and sent to the last control block, that computes the duty-cycle (m) for each phase. The duty-cycle is sent to the PWM generator (in orange) closing the model's loop. More details about generating the process model are indicated in Ref. [15].

3. Real-time model analysis

To develop an analysis tool able to replicate as close as possible the behavior of a real electric urban vehicle, it is clear that using complex simulation developments for the electrical assemblies will not be enough. However, building the mechanical models using advanced mathematical tools is highly time consuming and would require a lot of expertise and education in this direction. Hence, a wise solution is to engage a software created for such simulations, like Simcenter Amesim from Siemens.

In order to highlight the previously mentioned application with a practical example, an urban vehicle's mechanical model is developed. The vehicle is an electrical tricycle used for cargo delivery purposes. Its main specifications are presented in **Table 3**.

In **Figure 13**, the Amesim model is depicted, observing that none electrical assemblies are included in the model. Those are modeled in Matlab/Simulink as it will be presented in the following sections. The link between the two models will be performed using National Instruments VeriStand software, able to create an integrated project that runs a co-simulation on a real-time processor for the complete vehicle.

The Matlab/Simulink model for the electrical assemblies is built putting together all the components detailed in Section 2 based on the hierarchy depicted in **Figure 12**. It has to be mentioned that the pictograms cannot be exactly replicated in Simulink as the inputs are on the left and the outputs always on the right. However, the organization is the same as it can be observed in **Figure 14**.

Having all the models above presented, the goal is to create a virtual simulation platform that allows simple transition hardware in the loop (HiL) testing. The latter

Parameter	Symbol	Value
Maximum mass	m	250 kg
Maximum speed	v	10 m/s
Wheel radius	Rw	12"
Maximum road slope	α	15%
Rolling resistance	f	0.017
Front area	A	1.2 m ²
Air drag coefficient	Cs	0.36
Gear ratio	Gr	5

Table 3.
Tricycle specifications.

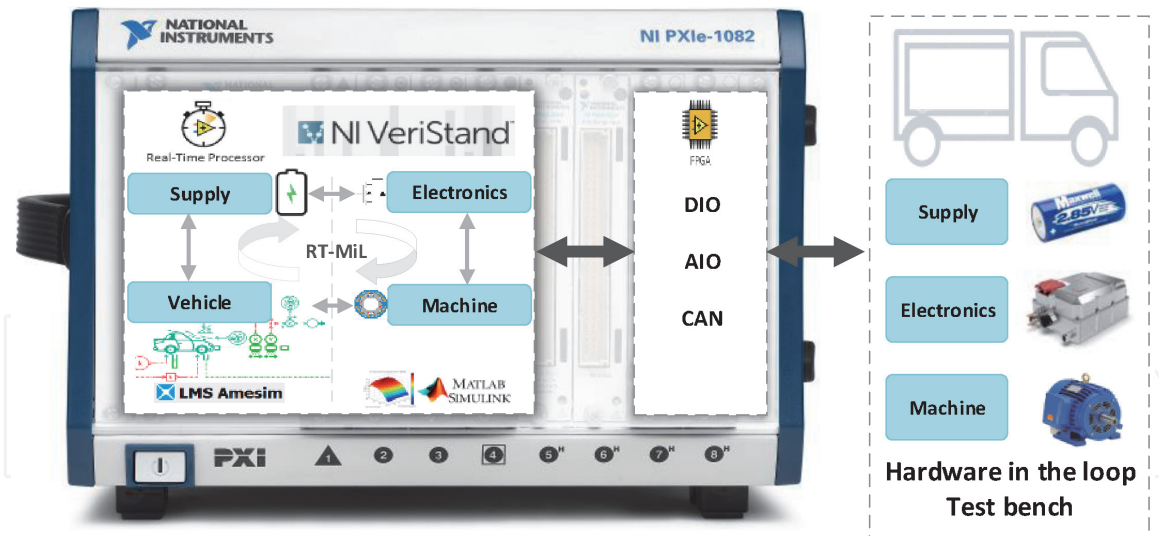


Figure 15.
The platform's hardware/software architecture.

hardware. The user has the possibility to run the virtual model and in parallel the actual hardware in a bidirectional communication network, performing as an entire entity. The real-time processor also features the possibility to run only the simulation of the entire vehicle, without any hardware connected to it. The difference between such a simulation and one running on a PC is that the latter would take tens of hours for a road cycle of 30 minutes while on the real-time computer it will take the elapsed 30 minutes.

Both scenarios will be presented in detail in the following section, comparing the results and discussing them.

4. System validation via urban scenario

Taking advantage of the flexibility of the above presented analysis platform one can perform a study using only simulation, hence a virtual vehicle or can combine virtual elements (mechanical assemblies) with real ones (the propulsion motor). For a robust simulation model, it is important to ensure that using it as reference will return results that mimic closely the actual hardware. In doing so, the platform depicted in **Figure 15** was initially tested only for simulation, the complete system being entirely virtual and running on the PXIe under VeriStand software. The latter ensured the continuous communication of the Simulink model (running all electrical assemblies) with Amesim (running all mechanical systems).

It has to be mentioned that the chosen simulation used the most complex model assemblies from those presented in Section 2. In order to avoid redundancy, results for the rest of the levels are not presented in the chapter, however during the presentations in Section 2, the main differences between the models were already presented.

The second step was to keep in the virtual level the mechanical assemblies (in Amesim) and the battery model (in Simulink). The rest of the virtual electrical assemblies (the inverter, the motor and the controller) were replaced with their laboratory homologs. These remained however connected to the simulator via the analog/digital IOs presented earlier.

The outcome of this comparative analysis was more than satisfactory. Firstly, the reference torque was plotted versus the simulated and the measured ones. It can be observed that the agreement is very good proving the accuracy of the simulator.

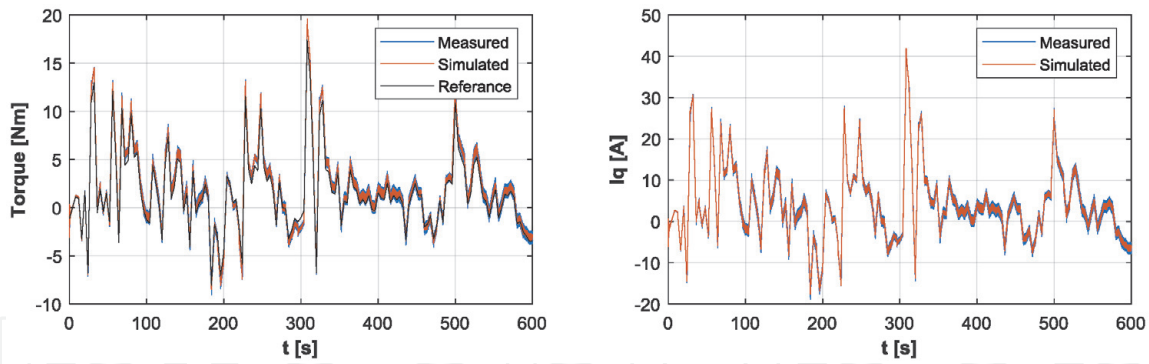


Figure 16.
The comparative analysis of the torque (right) and q current component (left).

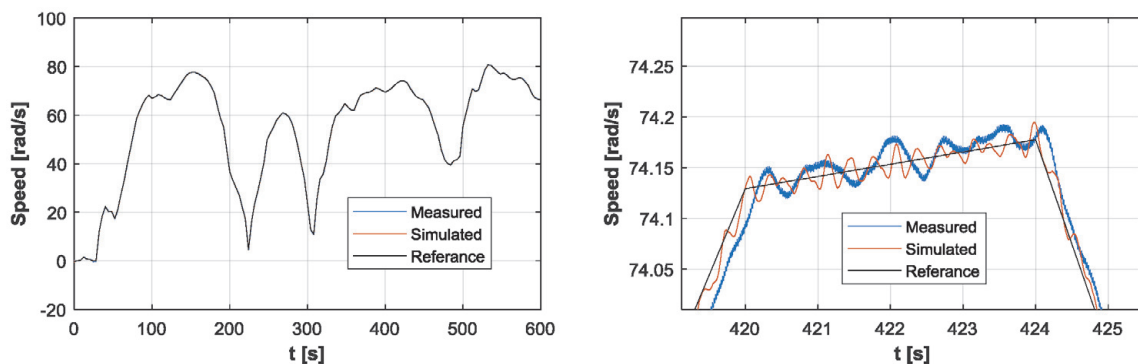


Figure 17.
The comparative analysis of the speed (right) and its zoomed section (left).

Knowing that in a PMSM the q current component is responsible for the torque production, for both measured and simulated cases this was recorded and depicted in **Figure 16**-right. The same conclusions as for the comparative analysis of the torque characteristics can be considered. In order to avoid redundancy, the d currents were not depicted as those values are forced to 0 at all times.

The slight differences between the values plotted in **Figure 16** are more due to measurements error and noise. No filtering was used what's so ever in order to avoid any unnecessary postprocessing (**Figure 17**).

The slight differences between the values plotted in **Figure 16** are more due to measurements error and noise. No filtering was used what's so ever in order to avoid any possible influence over the quality results.

Smoother results can be reached if certain data sampling time or other such procedure is considered. However, the interest in this entire work was to prove that one can reach very accurate results when building responsible simulation programs. Also using a real-time processor can benefit the study with fast analysis and reliable results as well.

5. Conclusions

When it comes to develop simulation tools wise enough to reach the expectations of the fast-growing industry in the field of engineering, one needs to consider solutions that become hands-on. These must be flexible to changes, simple to implement and accurate when it comes to results. Using real-time processors to run these simulations offers reduced analysis time, accurate results and close to real

study. The same processors when engaged, ease the transition from simulation to HiL testing, simply by replacing virtual components with real ones. Choosing between the complexity levels of the virtual models, allows the user to select the accuracy and the necessary time invested in the development. The system under analysis being compound of several assemblies, more complex models can be considered for those that are of interest, while the rest can be ideal ones.

In the present chapter the main assemblies of an electrical urban vehicle's traction system are presented, offering the choice of complexity, mathematical description and EMR organization. The latter is introduced as graphical method for representing the elements of any simulation program by respecting the actual action/reaction physical and natural laws.

It has to be mentioned that the nature of the present chapter is more towards a review than of an academic lecture, hence the reader is encouraged to consult the indicated references that are guidance and complementary information to the one described in the previous pages.

Acknowledgements

This work was supported by a grant of the Ministry of Research, Innovation and Digitization, CNCS/CCCDI – UEFISCDI, project number PN-III-P2-2.1-PED-2019-4384, within PNCDI III.

This work was supported by a grant of the Ministry of Research, Innovation and Digitization, CNCS/CCCDI – UEFISCDI, project number PN-III-P1-1.1-TE-2019-0411, within PNCDI III.


IntechOpen

Author details

Raul Octavian Nemeş, Mircea Ruba*, Sorina Maria Giornei and Raluca Maria Raia
Technical University of Cluj Napoca, Romania

*Address all correspondence to: mircea.ruba@emd.utcluj.ro

IntechOpen

© 2021 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/3.0>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited. 

References

- [1] A. Bouscayrol, R. Schoenfeld, G. Dauphin-Tanguy, G.-H. Geitner, X. Guillaud, A. Pennamen, and J. Hautier, "Different Energetic Descriptions for Electromechanical Systems," in Proc. of the European Conference on Power Electronics and Applications (EPE), Sep. 2005, Dresden, Germany.
- [2] K. Chen, A. Bouscayrol, and W. Lhomme, "Energetic macroscopic representation and inversion-based control: Application to an electric vehicle with an electrical differential," *Journal of Asian Electric Vehicles*, vol. 6, no. 1, pp. 1097–1102, Jun. 2008.
- [3] A. Bouscayrol, X. Guillaud, P. Delarue, and B. Lemaire-Semail, "Energetic macroscopic representation and inversion-based control illustrated on a wind-energy conversion system using hardware-in-the-loop simulation," *IEEE Transactions on Industrial Electronics*, vol. 56, no. 12, pp. 4826–4835, Dec. 2009.
- [4] Nemes, R.O., Maria Ciornei, S., Ruba, M., Martis, C., "Parameters Identification Using Experimental Measurements for Equivalent Circuit Lithium-Ion Cell Models", 11th International Symposium on Advanced Topics in Electrical Engineering, ATEE 2019; Bucharest; Romania; 28 March 2019 through 30 March 2019; DOI: 10.1109/ATEE.2019.8724878
- [5] Ruba, M., Ciornei, S., Nemeş, R., Martis, C., "Detailed Design of Second Order Model of Lithium-Ion Battery Simulator Based on Experimental Measurements", 11th International Symposium on Advanced Topics in Electrical Engineering, ATEE 2019; Bucharest; Romania; 28 March 2019 through 30 March 2019; DOI: 10.1109/ATEE.2019.8724878
- [6] A. Nikolian, Firouz, Y. Gopalakrishnan, R. Timmermans, J. M. Timmermans, N. Omar, P. van den Bossche, J. van Mierlo, "Lithium ion batteries Development of advanced electrical equivalent circuit models for nickel manganese cobalt lithium-ion," *Energies* 2016.
- [7] X. Zhang, J. Lu, S. Yuan, J. Yang, X. Zhou, "A novel method for identification of lithium-ion battery equivalent circuit model parameters considering electrochemical properties," *J. Power Sources* 2017.
- [8] M. Mathew, Q. H. Kong, J. McGrory, M. Fowler, Simulation of lithium ion battery replacement in a battery pack for application in electric vehicles, *J. Power Sources* 2017.
- [9] D. Graovac and M. Pürschel, "MOSFET power losses calculation using the data-sheet Parameters," 2006. <https://www.infineon.com>.
- [10] S. Shuvo, E. Hossain, T. Islam, A. Akib, S. Padmanaban and M. Z. R. Khan, "Design and hardware implementation considerations of modified multilevel cascaded H-bridge inverter for photovoltaic system," in *IEEE Access*, vol. 7, pp. 16504-16524, 2019, doi: 10.1109/ACCESS.2019.2894757.
- [11] D. E. Pinto, A. C. Pop, J. Kempkes, J. Gyselinck, "dq0-modeling of interior permanent-magnet synchronous Machines for High-Fidelity Model Order Reduction, Optim & Intl Aegean Conference on Electrical Machines and Power Electronics, May, 2017.
- [12] D. Zarko, D. Ban, R. Klaric, "Finite element approach to calculation of parameters of an interior permanent magnet motor", *Automatika* 46, 2005
- [13] J. Wu, J. Wang, C. Gan, Q. Sun and W. Kong, "Efficiency optimization of

PMSM drives using field-circuit coupled FEM for EV/HEV applications," in IEEE Access, vol. 6, pp. 15192-15201, 2018, doi: 10.1109/ACCESS.2018.2813987.

[14] S. M. Ciornei, M. Ruba, R. O. Nemeş and C. Martiş, "Multi-level models for a light electric vehicle propulsion system using EMR organisation," 2020 International Symposium on Power Electronics, Electrical Drives, Automation and Motion (SPEEDAM), 2020, pp. 357-362, doi: 10.1109/SPEEDAM48782.2020.9161855.

[15] M. Ruba, R. O. Nemes, S. M. Ciornei, C. Martis, A. Bouscayrol and H. Hedesiu, "Digital twin real-time FPGA implementation for light electric vehicle propulsion system using EMR organization," 2019 IEEE Vehicle Power and Propulsion Conference (VPPC), 2019, pp. 1-6, doi: 10.1109/VPPC46532.2019.8952428.