Modeling and simulation aspects of AC machines

MICHAEL POPP, PATRICK LAZA, WOLFGANG MATHIS

Leibniz Universität Hannover
Institute of Theoretical Electrical Engineering
Appelstraße 9A, 30167 Hannover, Germany
e-mail: popp@tet.uni-hannover.de

Abstract: In the field of power and drive systems, electrical AC machines are mostly modeled using a set of explicit ordinary differential equations in a state space representation. It is shown, that by using other equation types for simulation, algebraic constraints arising from aggregating several machines to a more complex system can directly be considered. The effects of different model variants on numerical ODE/DAE solvers are investigated in the focus of this work in order perform efficient simulations of larger systems possessing electrical AC machines.
Key words: State Space Modeling, AC Machines, ODE and DAE Solvers, MATLAB

1. Introduction

State space models of electrical single- and multiphase machines are widely used in the analysis, design and control of power and drive systems. Following standard textbooks like [1-3] among many others, the derivation of such simulation models follows similar procedures, even though the resulting model equations may differ with respect to their form and structure. These differences in various model representations for one and the same machine gave rise to the question, if the choice of a specific representation can be advantageous over others for the numerical simulation.

Starting with a short exposure of the general modeling principles of electrical machines, the model equations of an induction machine are derived and simulated numerically for an example parameter set. Thereafter, several variants of this model representation are derived systematically with respect to the basic structure of the resulting differential equations, the set of state variables, the underlying coordinate system as well as the coupling between the electrical and the mechanical subsystem. In order to find the most appropriate model representation, each variant had been simulated with several numerical solvers for ordinary differential equations (ODE) and differential algebraic equations (DAE) in the problem solving environment MATLAB [4, 5]. Based on the structural properties of these models numerical aspects will be discussed in detail aiming towards an optimized combination of model and numerical solver in order to reduce the computational effort for simulation.

2. General machine modeling

The behavior of any electrical machine can completely be described by the following three equations [3]:

$$\mathbf{u} = \mathbf{Ri} + \frac{\mathrm{d}}{\mathrm{d}t}\mathbf{\Psi} \, , \tag{1}$$

$$\mathbf{\Psi} = \mathbf{L}(\theta)\mathbf{i} \, , \tag{2}$$

$$J\frac{\mathrm{d}}{\mathrm{d}t}\omega = -D\omega + m_1(\mathbf{\Psi}, \mathbf{i}, \theta) + m_e \qquad \text{with} \qquad \omega = \frac{\mathrm{d}}{\mathrm{d}t}\theta \, . \tag{3}$$

Therein, the components of the vectors **u, i** and $\mathbf{\Psi}$ represent the voltages, currents and flux linkages per phase, respectively. The self- and mutual magnetic coupling of the phase windings in the case of linear magnetic materials is expressed by the inductance matrix $\mathbf{L}(\theta)$, which is in general dependent from the rotor position angle θ. The mechanical subsystem of the rotor is described by (3), in which J is the moment of inertia, D is the mechanical damping constant, m_e is the external driving torque and m_i is the torque developed by the machines' electromagnetic subsystem.

In order to model a specific type of electrical machine, i.e. a synchronous or an induction machine, the matrices **R** and $\mathbf{L}(\theta)$ have to be determined under a given set of assumptions under which the derived model shall be valid. Such an assumption could be the restriction to consider only the spatial fundamental of the magnetic field in the air gap of the modeled machine. The question how to calculate the self- and mutual inductances contained in $\mathbf{L}(\theta)$ under such an assumption is covered in detail in most of the standard textbooks, i.e. [1, 2], and is not subject of this paper. Here these inductances are treated as known parameters of the system under examination. It has also to be mentioned, that when specializing the matrices in (1) - (3) in the sense mentioned above, the question of the underlying coordinate system, in which the quantities summarized in the vectors **u, i** and $\mathbf{\Psi}$ are expressed, arises. This aspect and its influence on the numerical efficiency is covered in detail in section 5.

Even though (1) - (3) can be directly used for numerical simulation purposes by utilizing an adequate solver for systems of semi-explicit differential-algebraic equations (DAE), the further degrees of freedom concerning the basic equation structure itself as well as the selection of possible state variables as mentioned in the introduction have to be considered. In the following all these aspects are discussed in detail based on an example simulation scenario which is interpreted with respect to an efficient numerical simulation.

3. Induction machine model

The general machine Equations (1) - (3) are specialized in this section in order to model a wound rotor induction machine following the basic principles in [1, 2]. As mentioned above, the task is to find appropriate matrices **R** and $\mathbf{L}(\theta)$ based on the machines cross-section in
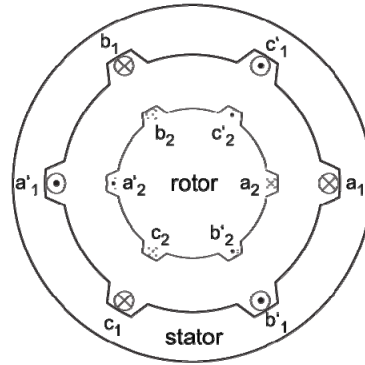
Fig. 1. Cross section of a wound rotor induction machine

Fig. 1. It is convenient to exploit the natural partitioning of the complete machine in subsystems, e.g. the stator subsystem (index 1) and the rotor subsystem (index 2). $\mathbf{R}$ and $\mathbf{L}(\theta)$ then read as follows:

$$\mathbf{R} = \begin{bmatrix} \mathbf{R}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{R}_2 \end{bmatrix}, \tag{4}$$

$$\mathbf{L}(\theta) = \begin{bmatrix} \mathbf{L}_{11} & \mathbf{L}_{12}(\theta) \\ \mathbf{L}_{12}^{\mathrm{T}}(\theta) & \mathbf{L}_{22} \end{bmatrix}, \tag{5}$$

with $\mathbf{0}$ denoting the zero matrix of appropriate dimension and $\mathbf{R}_i = \mathrm{diag}(R_i)$, $i \in \{1,2\}$, expressing, that for each phase only the wire resistance of the coil itself causes an ohmic voltage drop with respect to (1). The inductance submatrices $\mathbf{L}_{ii}, i \in \{1,2\}$, describe the self- and mutual inductive coupling of the phases within the subsystem denoted by the index i and are independent from the rotor position angle θ. Defining the phase's self-inductance as L_{li} and the mutual phase-to-phase inductance as L_{mi}, $\mathbf{L}_{ii}$ reads

$$\mathbf{L}_{ii=} \begin{bmatrix} L_{mi} + L_{li} & -\frac{1}{2}L_{mi} & -\frac{1}{2}L_{mi} \\ -\frac{1}{2}L_{mi} & L_{mi} + L_{li} & -\frac{1}{2}L_{mi} \\ -\frac{1}{2}L_{mi} & -\frac{1}{2}L_{mi} & L_{mi} + L_{li} \end{bmatrix}. \tag{6}$$

The angular dependent submatrix $\mathbf{L}_{12}(\theta)$ expresses the magnetic coupling between rotor and stator. Using Fig. 1,

$$\mathbf{L}_{12}(\theta) = L_{m12} \begin{bmatrix} \cos\theta & \cos\left(\theta+\frac{2\pi}{3}\right) & \cos\left(\theta-\frac{2\pi}{3}\right) \\ \cos\left(\theta-\frac{2\pi}{3}\right) & \cos\theta & \cos\left(\theta+\frac{2\pi}{3}\right) \\ \left(\theta+\frac{2\pi}{3}\right) & \cos\left(\theta-\frac{2\pi}{3}\right) & \cos\theta \end{bmatrix} \tag{7}$$

can directly be read off with $\mathbf{L}_{m12}$ as maximal stator-rotor coupling inductance. Having determined all block matrices of $\mathbf{L}(\theta)$ the coupling between the electromagnetic subsystems with the mechanics of the rotor in (3) can be formulated. One efficient way in doing so is based on the change of magnetic energy or co-energy, respectively, stored in the air gap of the modeled machine and which is converted from (generator operation) or to (motor operation) mechanical energy [3]. Using the co-energy approach,

$$m_i = \mathbf{i}^{\mathrm{T}} \frac{\mathrm{d}\mathbf{L}(\theta)}{\mathrm{d}\theta} \mathbf{i} \tag{8}$$

can be derived and the energy approach gives

$$m_i = -\mathbf{\Psi}^{\mathrm{T}} \frac{\mathrm{d}\left(\mathbf{L}^{-1}\right)^{\mathrm{T}}(\theta)}{\mathrm{d}\theta} \mathbf{\Psi} \tag{9}$$

as an alternative expression for the so called inner torque of the modeled machine.

After selecting one of the various possible model representations, the simulation can be performed. Fig. 2 and 3 show the simulation results of the fly-wheel start-up of an 2,2 kW (3 hp) induction machine operated as asynchronous motor which is supplied by an ideal three-phase symmetric voltage source.
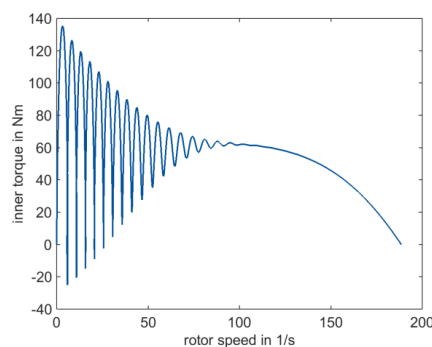

Fig. 2. Transient torque over rotor speed during fly wheel start-up of the example machine
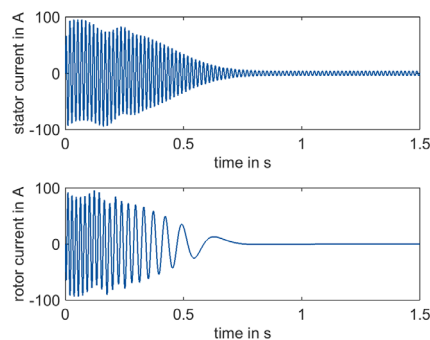

Fig. 3. Stator and rotor currents during fly wheel start-up of the example machine

The parameters of this benchmark example used throughout this paper are taken from [1]. The resulting dynamic torque-speed characteristic is shown in Fig. 2 and the corresponding current waveforms are depicted in Fig. 3.

4. State variables and electromechanical coupling

Introducing a state vector $\mathbf{x}$, which collects the electrical and mechanical state variables of a machine, its electrical subvectors can be partitioned so, that they are compatible to the block partitioning of the matrices $\mathbf{R}$ and $\mathbf{L}(\theta)$:

$\quad$ $\mathbf{x}_1$: stator related currents or flux-linkages, respectively;

$\quad$ $\mathbf{x}_2$: rotor related currents or flux-linkages, respectively;

$\quad$ $\mathbf{x}_m$: mechanical state variables (θ and ω).

With the definition above, one of the following four combinations of electrical state variables $\mathbf{x}_e = (\mathbf{x}_1, \mathbf{x}_2)^{\mathrm{T}}$ can be used to formulate simulation models:

$$\mathbf{x}_e = \begin{bmatrix} \mathbf{i}_1 \\ \mathbf{i}_2 \end{bmatrix}, \quad \mathbf{x}_e = \begin{bmatrix} \mathbf{\Psi}_1 \\ \mathbf{\Psi}_2 \end{bmatrix}, \tag{10}$$

$$\mathbf{x}_e = \begin{bmatrix} \mathbf{i}_1 \\ \mathbf{\Psi}_2 \end{bmatrix} \quad \text{or} \quad \mathbf{x}_e = \begin{bmatrix} \mathbf{\Psi}_1 \\ \mathbf{i}_2 \end{bmatrix}. \tag{11}$$

First order ODE model variants with the state variable sets in (10) can be derived from the DAE description given in (1) - (3) by substituting either $\mathbf{i}$ or $\mathbf{\Psi}$ in the voltage Equation (1) by using the relation given in the flux linkage Equation (2), provided that the inverse inductance matrix $\mathbf{L}^{-1}(\theta)$ exists, which is fulfilled for the induction machine model considered here. Written in explicit form, this yields

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{\Psi} = -\mathbf{R}\mathbf{L}^{-1}(\theta)\mathbf{\Psi} + \mathbf{u} , \tag{12}$$

or with currents $\mathbf{i}$ as state vector

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{i} = -\mathbf{L}^{-1}(\theta)\big(\mathbf{R} + \omega\frac{\mathrm{d}}{\mathrm{d}\theta}\mathbf{L}(\theta)\big)\mathbf{i} + \mathbf{L}^{-1}(\theta)\mathbf{u} , \tag{13}$$

respectively. Model variants with the hybrid state vector configurations in (11) can be derived by substituting either $\mathbf{i}$ in (13) or $\mathbf{\Psi}$ in (12) with one of the following transformation relationships.

$$\begin{bmatrix} \mathbf{i}_1 \\ \mathbf{i}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{1} & \mathbf{0} \\ -\mathbf{L}_{22}^{-1}\mathbf{L}_{12}^{\mathrm{T}} & -\mathbf{L}_{22}^{-1} \end{bmatrix} \begin{bmatrix} \mathbf{i}_1 \\ \mathbf{\Psi}_2 \end{bmatrix}, \tag{14}$$

$$\begin{bmatrix} \mathbf{i}_1 \\ \mathbf{i}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{L}_{11}^{-1} & -\mathbf{L}_{11}^{-1}\mathbf{L}_{12}^{\mathrm{T}} \\ \mathbf{0} & \mathbf{1} \end{bmatrix} \begin{bmatrix} \mathbf{\Psi}_1 \\ \mathbf{i}_2 \end{bmatrix}, \tag{15}$$

$$\begin{bmatrix} \mathbf{\Psi}_1 \\ \mathbf{\Psi}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{L}_{11} - \mathbf{L}_{12}\mathbf{L}_{22}^{-1}\mathbf{L}_{12}^{\mathrm{T}} & \mathbf{L}_{12}\mathbf{L}_{22}^{-1} \\ \mathbf{0} & \mathbf{1} \end{bmatrix} \begin{bmatrix} \mathbf{i}_1 \\ \mathbf{\Psi}_2 \end{bmatrix}, \tag{16}$$

$$\begin{bmatrix} \mathbf{\Psi}_1 \\ \mathbf{\Psi}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{1} & \mathbf{0} \\ \mathbf{L}_{12}^{\mathrm{T}}\mathbf{L}_{11}^{-1} & \mathbf{L}_{22} - \mathbf{L}_{12}^{\mathrm{T}}\mathbf{L}_{11}^{-1}\mathbf{L}_{12} \end{bmatrix} \begin{bmatrix} \mathbf{\Psi}_1 \\ \mathbf{i}_2 \end{bmatrix}. \tag{17}$$

Even though the transformation relationships (14-17) are derived from the flux linkage Equation (2), all variants result in different models with different properties concerning the numerical solution process, as seen below. Furthermore, it has to be mentioned, that additional transformation relationships can be derived in a similar way to that in (14-17) wherein the electrical state vector takes the form $(\mathbf{i}_1, \mathbf{\Psi}_1)^{\mathrm{T}}$ or $(\mathbf{i}_2, \mathbf{\Psi}_2)^{\mathrm{T}}$ respectively. In these cases, the state vector would only carry the information of either the stator or the rotor subsystem, which contradicts the modeling of the whole machine. For the induction machine

$$\det(\mathbf{L}_{12}(\theta)) = 0 , \tag{18}$$

for all values of θ can be found. This relationship prohibits the existence of such state vector variants as stated above, because the corresponding transformation matrices would contain the inverse of $\mathbf{L}_{12}(\theta)$ which does not exist.
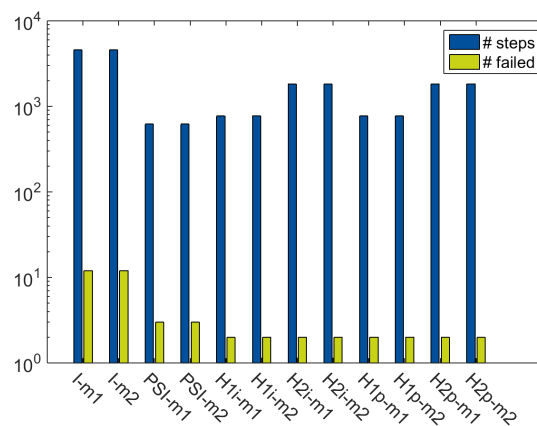

Fig. 4. Solver statistics for different simulation runs of the example machine represented as explicit ODE and solved with MATLAB's ode45 for different sets of state variables (I: currents as state variables, PSI: flux linkages as state variables, H1i, H1p, H2i, H2p: hybrid state vectors as defined in (14) - (17)) and electromechanical coupling variants (m1: co-energy based formulation (8), m2: energy based formulation (9))

Changing the set of state variables in the differential equations of the electromagnetic subsystem, the same change of variables has also to be applied to the equation of the mechanical subsystem; especially to the term describing the inner torque. Referring to section 3, the two variants of the electromechanical coupling term result in

$$m_i = (\mathbf{T}\mathbf{x}_e)^{\mathrm{T}} \frac{\mathrm{d}\mathbf{L}(\theta)}{\mathrm{d}°\theta} \mathbf{T}\mathbf{x}_e \qquad (19)$$

and

$$m_i = -(\mathbf{T}\mathbf{x}_e)^{\mathrm{T}} \frac{\mathrm{d}(\mathbf{L^{-1}})^{\mathrm{T}}(\theta)}{\mathrm{d}°\theta} \mathbf{T}\mathbf{x}_e \,, \qquad (20)$$

wherein **T** denotes the corresponding transformation matrix from (14-17).

In order to compare the influence of different choices of state vector variants as well as electromechanical coupling terms on the time it takes to simulate the benchmark example introduced in section 3 the twelve resulting combinations had been computed with the same ODE solver (MATLAB's ode45). The results of this study are shown in Fig. 4 in form of a graphical representation of the solver statistics. Three main results can be found by interpreting Fig. 4. First, the choice of the electromechanical coupling term, either the co-energy based formulation (8), (19) or the energy based formulation (9), (20), does not have any influence on the simulation performance. Both variants, which are marked with "m1" and "m2" in Fig. 4, need exactly the same number of time steps for each state vector variant. Second, the model variants with flux linkages **Ψ** as electrical state variables (marked with "PSI" in Fig. 4) need nearly one order of magnitude less time steps than the variant with currents **i** in (13) during the numerical solving process. As reason for these remarkable differences, the term

$$\omega \frac{\mathrm{d}}{\mathrm{d}\theta} \mathbf{L}(\theta)$$

in (13) had been identified. The additional state dependency brought into the electrical equations in form of ω makes it difficult for the step size controller to find an appropriate step size; especially in a highly transient operational state of the simulated machine like the startup used here intentionally as a benchmark. Third, Fig. 4 also reveals, that the model variant formulated with a state vector in hybrid form after (14) or (16), marked with "H1i" or "H1p", respectively, indeed need more time steps for obtaining an solution, but less failed steps, in which the given standard tolerances for solving the ODEs could not be met.

5. Reference frames

Besides the question, which set of state variables is used for simulation, further model variants arise when taking different reference frames (coordinate systems) into account in which these state variables are expressed. When reading off the machine's describing matrices **R** and $\mathbf{L}(\theta)$ the submatrices of the stator are naturally described in a stationary planar three-axis reference frame in which the axis a1, b1 and c1 correspond to the actual stator windings and its angular velocity equals zero; cf. Fig. 5.
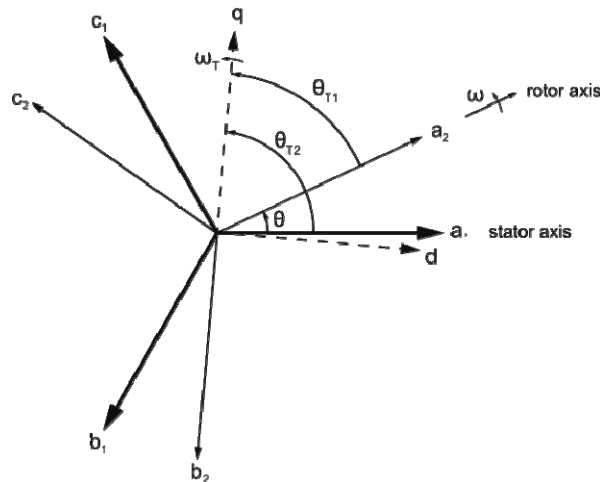
Fig. 5. Coordinate systems and axis definitions for an induction machine

This reference frame will be denoted as (abc)0 in the following. The natural submatrices of the rotor are also described in a planar three-axis reference frame which is fixed to the rotor, and therefore rotating with the angular frequency ω. It will be denoted as (abc)r in the following. For that reason, the resistances and inductances of the rotor itself appear independent from the rotor displacement angle θ. Only the magnetic coupling described by the block matrix $\mathbf{L}_{12}(\theta)$ becomes angular dependent in the choice of this natural reference frames.

It is common in the field of machine modeling and analysis to change the reference frames of the stator and the rotor and as a consequence of this of the coupling between both. Krause [1] gives an overview of common variants and their historical development. Staying with the nomenclature introduced above, each of both natural reference frames (abc)0 and (abc)r can be transformed into a orthogonal two-axis reference frame with an additional third axis covering unbalanced conditions. The most commonly used two-axis reference frames are:

      (dq0)0: stationary two-axis frame,

      (dq0)r: two-axis frame attached to the rotor,

      (dq0)s: two-axis frame rotating at synchronous speed.

A change of the reference frame can be applied by making use of the relationships (21)-(24) and their inverses:

$$\mathbf{g}_{dq0} = \mathbf{T}_{abc \to dq0}(\theta_T)\mathbf{g}_{abc}, \tag{21}$$

with

$$\theta_T = \theta_{abc} - \theta_{dq0}, \tag{22}$$

$$\omega_T + \frac{d}{dt}\theta_T, \tag{23}$$

$$\mathbf{T}_{\mathrm{abc}\to\mathrm{dq0}}(\theta_{\mathrm{T}}) = \sqrt{\frac{2}{3}} \begin{bmatrix} \cos\theta_{\mathrm{T}} & \cos\left(\theta_{\mathrm{T}} - \dfrac{2\pi}{3}\right) & \cos\left(\theta_{\mathrm{T}} + \dfrac{2\pi}{3}\right) \\ \sin\theta_{\mathrm{T}} & \sin\left(\theta_{\mathrm{T}} - \dfrac{2\pi}{3}\right) & \sin\left(\theta_{\mathrm{T}} + \dfrac{2\pi}{3}\right) \\ \sqrt{1/2} & \sqrt{1/2} & \sqrt{1/2} \end{bmatrix} \tag{24}$$

and **g** a vector containing arbitrary electrical or magnetic quantities. It has to be mentioned, that it is possible without restrictions to use different reference frames for the stator and the rotor.

In order to discover the effect of the choice of a specific coordinate system on the simulation efficiency the benchmark example had been simulated with different coordinate system combinations. The results of this study are summarized in Fig. 6 in form of the resulting solver statistics of MATLAB's ode45. Overall it can be seen that the differences in the computational effort for solving the benchmark problem are not as remarkable as when using different state variable sets as presented in section 4. Furthermore, it turns out that describing both electromechanical subsystems in the rotor-fixed two phase reference frame, denoted with "(dq0)r/r" in Fig. 6, yields the lowest number of steps. A model described completely in the synchronous reference frame, denoted with "(dq0)s/s" in Fig. 6, needs a similar number of steps for being solved. However, the number of failed solver attempts increases by more than a factor of 15 which indicates a minor suitability of the model variant for numerical treatment.
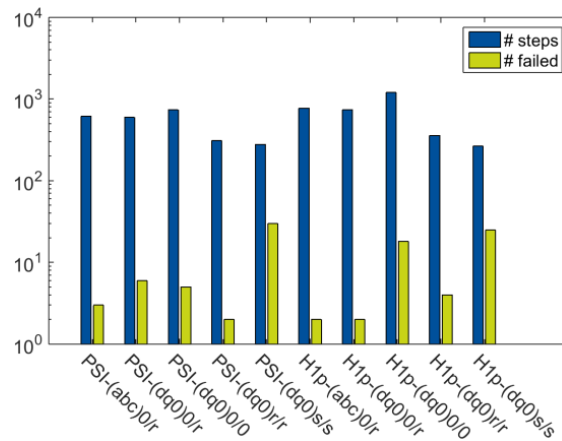

Fig. 6. Solver statistics for different simulation runs of the example machine represented as explicit ODE and solved with MATLAB's ode45 for different sets of state variables (PSI: flux linkages as state variables, H1p: hybrid state vector as defined in (16)) and coordinate systems ((abc): three phase coordinate system, (dq0): orthogonal two phase coordinate system; 0: stator fixed, r: rotor fixed, s: synchronous angular velocity with respect to stator/rotor quantities)
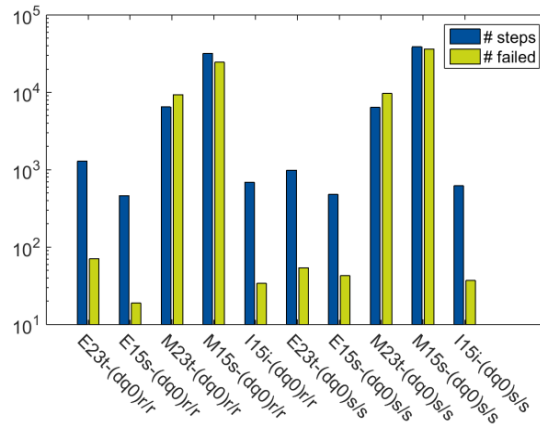
Fig. 7. Solver statistics for different simulation runs of the example machine with flux-linkages as state variables in different ODE representations (E: explicit, M: with mass matrix, I: implicit), ODE solvers (ode23t, ode15s and ode15i) and coordinate systems ((dq0)r/r: stator and rotor described in a rotor fixed two phase reference frame, (dq0)s/s: stator and rotor described in a synchronously rotating two phase reference frame)
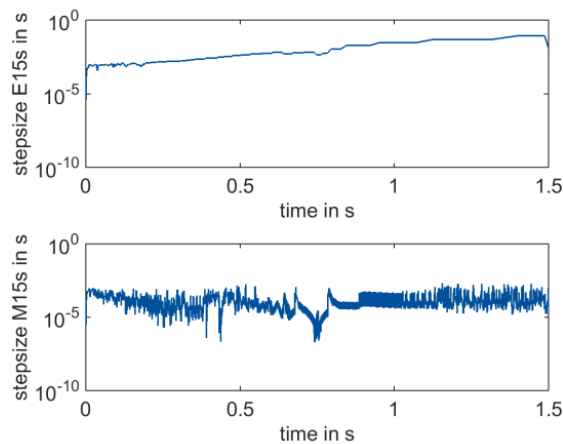

Fig. 8. Comparison of the ODE solver step sizes during the simulation of the example machine with flux-linkages as state variables for an explicit (E) and a mass matrix (M) ODE representation

6. Equation structures and numerical ODE solvers

In this section finally different combinations of basic equation structures and standard ODE solvers are studied with respect to computational efficiency. The presented results in the sections 4 and 5 were achieved by formulating the model's state space representation in form of an explicit ODE $\left(\dot{\mathbf{x}} = \mathbf{F}(\mathbf{x}, t)\right)$ which had been solved with MATLAB's standard ode45 solver. This approach is the first choice, when simulating the behavior of a single machine.

Are there more machines to be simulated in an aggregated model, other basic equation structures like a mass matrix formulation $\left(\mathbf{M}(\mathbf{x},t)\dot{\mathbf{x}} = \mathbf{F}(\mathbf{x},t)\right)$ or an implicit formulation $\left(\mathbf{0} = \mathbf{F}(\mathbf{x},\dot{\mathbf{x}},t)\right)$ have to be used in order to consider the additional algebraic constraints due to the aggregation. Appropriate standard MATLAB solvers for this equation structures are ode15s and ode23t for the mass matrix case and ode15i for the implicit case. For more details on the solvers, cf. [4, 5].

As in the sections 4 and 5, all possible combinations of equation structure, ODE solver and coordinate system had been simulated and selected results of the solver's performance are shown in Fig. 7. It turns out that explicit model representations, marked with "E" in Fig. 7, perform well with all investigated ODE solvers. The results of implicit representations, marked with "I" in Fig. 7, are similar besides the slightly higher number of failed steps whereas the mass matrix representations ("M") need about 84 times more steps and – even worse – the number of failed steps nearly equals the number valid steps. The reason for these remarkable differences can be found in the control schemes for the step size within the solver algorithm. In the case of solving an explicit model equation with the ode15s solver, the step sizes behave quite smoothly with a gradual rise towards the periodic steady state solution (Fig. 8, upper panel). In Fig. 8, lower panel, a very different behavior of the step size control can be observed. There a mass matrix representation had been simulated with the same solver, which needs to correct the step size at nearly every integration step.

7. Conclusion

In this paper, the numerical efficiency of different model representations of electrical machines is studied in an example oriented manner. It turns out, that the choice of the basic model equation structure, the set of state variables and the reference frame reveal a strong impact on the total number of time steps, which are needed by the numerical ODE solver to obtain the simulation result for given tolerances. Therefore, it is possible to save simulation time by utilizing an appropriate model representation.

In future work it is tried to figure out a systematic procedure for finding the most practicable model representation rather than benchmarking all possible variants as done for this study presented here.

References

[1] Krause, P.C., Wasynczuk, O., Sudhoff, S.D., Pekarek, S., *Analysis of Electric Machinery and Drive Systems*, 3rd edition, Wiley, IEEE Press (2013).

[2] Ong, C.M., *Dynamic Simulation of Electric Machinery*, Prentice Hall (1998).
[3] Müller, G., Ponick, B., *Theory of Electrical Machines (in German)*, Wiley–VCH (2009).
[4] Shampine, L.F. et al., *Solving ODEs with MATLAB*, Cambridge University Press (2003).
[5] Shampine, L.F., *Solving 0 = F(t,y(t),y'(t)) in Matlab*, Journal of Numerical Mathematics, 10(4): 29-310 (2002).