Energy Preserving Methods and Torque Computation From Energy Balance in Electrical Machine Simulations

Energy Preserving Methods and Torque Computation From Energy Balance in Electrical Machine Simulations

Lauri Perkkiö [1,2], Paavo Rasilo[2,3], Bishal Silwal[2], Antti Hannukainen[1], Antero Arkkio[2], and Timo Eirola[1]
[1]Department of Mathematics and Systems Analysis, Aalto University, PO Box 11100, FI-00076 AALTO, Finland
[2]Department of Electrical Engineering and Automation, Aalto University, PO Box 13000, FI-00076 AALTO, Finland
[3]Department of Electrical Engineering, Tampere University of Technology, P.O. Box 692, FI-33101 TAMPERE, Finland

Finite element analysis for the simulation of magnetic fields in electrical machines leads to an index-1 differential algebraic equation (as opposed to a conventional ordinary differential equation), because the electrical conductivity can be zero in certain regions. First, we construct a differential-algebraic equation-compatible time integration scheme which is energy-balanced, meaning that in a linear system the input, stored and lost powers sum exactly to zero. Second, we use a method based on the energy balance to compute torque. We show that the energy balance method approaches the virtual work principle applied at remeshing layer as the time step is refined. A similar result holds also if the rotation of the rotor is implemented by Nitsche's method, which is an instance of so-called mortar methods.

***Index Terms*—electric machine, energy balance, non-matching mesh, torque.**

I. Introduction

FORCES and torques can be computed by several different methods in finite element (FE) analysis of electromagnetic problems, especially when rotating electric machines are simulated [1]–[5]. Even if the methods give equal results for exact, non-discretized fields, at the finite element level there might be a significant difference. In this paper we analyze one proposed method where the torque of an electrical machine is computed from the power balance equation

$$P_{\mathrm{in}} = \tfrac{d}{dt} W_{\mathrm{stored}} + P_{\mathrm{loss}} + P_{\mathrm{torq}}, \tag{1}$$

where $P_{\mathrm{torq}} = \omega_m T$, mechanical angular speed ω_m times torque T. Thus, T could be in principle solved if all the other terms are known [6]–[8].

First, we introduce a differential algebraic equation (DAE)-compatible time stepping scheme, which either conserves energy if a conservative system is studied, or gives energy consistent losses for a dissipative system. The motivation for this method is to ensure that the time integration method itself does not cause numerical energy losses in (1). When, for instance, an induction machine is modeled, the non-conducting domains will lead to a FE-discretized equation that is an index - 1 DAE [9]. An ordinary differential equation (ODE) governs at the conducting areas, whereas (nonlinear) constraint equations hold at the nonconducting ones. The proposed time integration scheme is based on Gaussian collocation methods [10], a subfamily of implicit Runge-Kutta methods, where the DAE-constraint equations are required to hold at each time step, and if 4th or higher order scheme is used, constraints are required as certain averages of the Gaussian quadrature points [11]. The simplest of these methods can be stated as "Solve ODE by Implicit Midpoint method, but require the constraints to hold at the endpoint". The computational cost and solution error of this approach are very close to the conventional Crank-Nicolson method but the advantage is that the numerical energy balance is exactly zero for a linear problem, and closer to zero for a non-linear problem. Another advantage is that one obtains a polynomial representation for the solution in a consistent manner on each time step which can be used for interpolating the solution between time steps.

Second, we show that the torque computation method based on energy balance is actually a difference approximation to virtual work principle applied in the remeshing layer. This is illustrated by comparing Coulomb's method [4] (an implementation of virtual work principle) to torque by energy balance method using very short time steps. Similar results are obtained if Nitsche's method [12] is used instead of remeshing. Nitsche's method is an instance of discontinuous Galerkin methods, where the field is allowed to be discontinuous over element interfaces. In this case it allows a non-matching mesh in the airgap, which is convenient when rotor motion is modeled [13], [14]. Some problems emerge if the time step is very small. First, using the remeshing layer for torque integration will give oscillating results, and the energy balance method has the same problem. This oscillation is a numerical artefact, which occurs as the nodes slide past each other at the air gap interface. Second, we do not get the exact value (in temporal sense) of the virtual work principle, but effectively a difference approximation to it.

These findings are illustrated with numerical examples for a three-phase squirrel cage motor in 2D [15] , but the principle is the same for 3D simulations or different machine types, for example, permanent magnet machines.

II. Differential Algebraic Equation

Throughout the paper we use a standard two dimensional vector potential formulation for an electric machine [15], which leads to a PDE which is parabolic (like the heat equation) in conducting regions, and elliptic (like the Laplace equation) in non-conducting ones. The magnetic field $\boldsymbol{B}$ lies on the

Manuscript received December 11, 2111; revised December 12, 2112. Corresponding author: Lauri Perkkiö (email: lauri.perkkio at aalto.fi).

xy-plane and the currents $\boldsymbol{J} = J\boldsymbol{e}_z$ flow in the z-direction. By using a potential $A(x, y)$ such that $\boldsymbol{B} = \nabla \times (A\ \boldsymbol{e}_z)$, $\boldsymbol{E} = -\frac{\partial}{\partial t} A\boldsymbol{e}_z$, we are left with a PDE

$$-\nabla \cdot (\nu \nabla A) + \sigma \frac{\partial A}{\partial t} = J, \qquad (2)$$

where the reluctivity ν is a piecewise constant or a (non-hysteretic) function of the field $\boldsymbol{B}$ (or equivalently A), electric conductivity σ is zero in some regions, and the current density J in wires can be coupled to circuit equations, which in our test case model a squirrel cage induction motor. Zero Dirichlet boundary condition is imposed on the outer boundary of the stator. The initial state A_0 may be zero or, e.g., come from a magnetostatic or time-harmonic solution of (2).

The weak form of (2) is to find field A such that

$$\int_\Omega \nu \nabla A \cdot \nabla \phi + \int_\Omega \sigma \frac{\partial A}{\partial t} \phi = \int_\Omega J\phi \qquad (3)$$

for all test fields ϕ. After a standard finite element discretization, e.g., quadratic elements for A in 2D, the spatial discretization of PDE (3) is as follows. Some basis functions are completely in non-conducting ($\sigma = 0$) areas; denote the coefficients of these functions by $\boldsymbol{v} \in \mathbb{R}^M$. The (at least partially) conducting ones are labeled $\boldsymbol{u} \in \mathbb{R}^N$. Then the problem is to solve $(\boldsymbol{u}, \boldsymbol{v}) \in \mathbb{R}^{N+M}$ from the equation

$$\boldsymbol{k}_1(\boldsymbol{u}, \boldsymbol{v}) + \boldsymbol{M}\dot{\boldsymbol{u}} = \boldsymbol{j}(t) \qquad (4)$$

$$\boldsymbol{k}_2(\boldsymbol{u}, \boldsymbol{v}) = 0, \qquad (5)$$

where the mass matrix $\boldsymbol{M} \in \mathbb{R}^{N\times N}$ represents the conductivity term and $\boldsymbol{k}_1 : \mathbb{R}^{N+M} \to \mathbb{R}^N$ and $\boldsymbol{k}_2 : \mathbb{R}^{N+M} \to \mathbb{R}^M$ are nonlinear if the reluctivity ν depends on the field, i.e. we have nonlinear magnetic materials. In the linear case we simply have $\boldsymbol{k}_1 = \boldsymbol{K}_{uu}\boldsymbol{u} + \boldsymbol{K}_{uv}\boldsymbol{v}$ and $\boldsymbol{k}_2 = \boldsymbol{K}_{vu}\boldsymbol{u} + \boldsymbol{K}_{vv}\boldsymbol{v}$, where $\boldsymbol{K}_{ij}$ are fixed matrices. Equation (4) is an ODE, whereas (5) is an algebraic equation (AE), sometimes called constraint equation, which correspond to areas where the conductivity σ is zero. The variables $\boldsymbol{u}$, whose derivative appears in the Equation (4), are called differential variables, whereas the rest ($\boldsymbol{v}$) are called algebraic variables. The system of equations (4-5), with the assumption that $\boldsymbol{v}$ can be solved for any given $\boldsymbol{u}$ from (5), is called a semi-explicit index-1 DAE [10]. More generally, we write such DAE system as

$$\boldsymbol{M}\dot{\boldsymbol{u}} = \boldsymbol{f}(\boldsymbol{u}, \boldsymbol{v}, t) \qquad (6)$$

$$0 = \boldsymbol{g}(\boldsymbol{u}, \boldsymbol{v}). \qquad (7)$$

For some time integration methods the initial state $(\boldsymbol{u}_0, \boldsymbol{v}_0)$ has to be consistent, which means that $(\boldsymbol{u}_0, \boldsymbol{v}_0)$ has to satisfy (7) (and possibly some so-called hidden constraints if higher than index-1 DAE is considered). Otherwise, the solution might oscillate when Crank-Nicolson (trapezoidal rule) is used for time stepping. If the initial state is obtained as the real part of a non-linear harmonic approximation, then it is inconsistent in general. Therefore, one could solve $\boldsymbol{v}$ with respect to given $\boldsymbol{u}_0$ obtained from the harmonic approximation, and use that as the initial state $\boldsymbol{v}_0$.

The two conventional time integration methods for solving the system (4-5) are Implicit (or Backward) Euler, written as

$$\boldsymbol{M}\boldsymbol{u}_{i+1} = \boldsymbol{M}\boldsymbol{u}_i + h\boldsymbol{f}_{i+1} \qquad (8)$$

$$0 = \boldsymbol{g}(\boldsymbol{u}_{i+1}, \boldsymbol{v}_{i+1}), \qquad (9)$$

and Crank-Nicolson (or Trapezoidal) method

$$\boldsymbol{M}\boldsymbol{u}_{i+1} = \boldsymbol{M}\boldsymbol{u}_i + \tfrac{h}{2}(\boldsymbol{f}_i + \boldsymbol{f}_{i+1}) \qquad (10)$$

$$0 = \boldsymbol{g}(\boldsymbol{u}_{i+1}, \boldsymbol{v}_{i+1}), \qquad (11)$$

where h is the time step length, $\boldsymbol{f}_i = \boldsymbol{f}(\boldsymbol{u}_i, \boldsymbol{v}_i, t_i)$ and $\boldsymbol{f}_{i+1} = \boldsymbol{f}(\boldsymbol{u}_{i+1}, \boldsymbol{v}_{i+1}, t_i + h)$. These methods are otherwise suitable for solving the given DAE, but if one wishes to have an exact energy balance at each time step (at least for a linear system), then slightly different methods are required. Suitable methods are presented in Section IV.

III. Energy

For simplicity, we assume a stationary rotor and current-fed system. This can be generalized to coupled circuit equations, as their energy and loss terms have a similar form. Rotor motion, and the relation to torque, is more delicate and it will be discussed later.

For convenience, we denote $\boldsymbol{x} = [\boldsymbol{u} \quad \boldsymbol{v}]^{\mathrm{T}}$ and write the DAE (4-5) shortly as

$$\boldsymbol{k}(\boldsymbol{x}) + \boldsymbol{M}\dot{\boldsymbol{x}} = \boldsymbol{j}(t), \qquad (12)$$

where $\boldsymbol{k} = [\boldsymbol{k}_1 \quad \boldsymbol{k}_2]^{\mathrm{T}}$ and $\boldsymbol{M}$ has the obvious zero rows and columns. Multiplying (12) by $\dot{\boldsymbol{x}}^{\mathrm{T}}$ we obtain

$$\dot{\boldsymbol{x}}^{\mathrm{T}}\boldsymbol{k}(\boldsymbol{x}) + \dot{\boldsymbol{x}}^{\mathrm{T}}\boldsymbol{M}\dot{\boldsymbol{x}} = \dot{\boldsymbol{x}}^{\mathrm{T}}\boldsymbol{j}(t), \qquad (13)$$

which is the discrete form of the equation

$$\int_\Omega \nu(\tfrac{\partial}{\partial t}\boldsymbol{B}) \cdot \boldsymbol{B} + \int_\Omega \sigma |\boldsymbol{E}|^2 = -\int_\Omega \boldsymbol{E} \cdot \boldsymbol{J}.$$

These quantities are, respectively, rate of change of magnetic energy, heat loss by Ohmic currents, and input power. Let us label them respectively

$$\tfrac{d}{dt}E_{\mathrm{mag}} + P_{\mathrm{loss}} = P_{\mathrm{in}}.$$

Accumulated energies are their time integrals

$$E_{\mathrm{mag}}(t) - E_{\mathrm{mag}}(0) + \int_0^t \dot{\boldsymbol{x}}^{\mathrm{T}}\boldsymbol{M}\dot{\boldsymbol{x}}\, dt = \int_0^t \dot{\boldsymbol{x}}^{\mathrm{T}}\boldsymbol{j}(t)\, dt, \qquad (14)$$

which we denote

$$\Delta E_{\mathrm{mag}} + \Delta E_{\mathrm{loss}} = \Delta E_{\mathrm{in}}. \qquad (15)$$

Only E_{mag} is an explicit function of $\boldsymbol{x}$, for example $E_{\mathrm{mag}} = \frac{1}{2}\boldsymbol{x}^{\mathrm{T}}\boldsymbol{K}\boldsymbol{x}$ in the linear case. After time discretization (15) may not exactly hold, so we call its residual the energy imbalance

$$\Delta E_{\mathrm{bal}} = \Delta E_{\mathrm{mag}} + \Delta E_{\mathrm{loss}} - \Delta E_{\mathrm{in}}. \qquad (16)$$

It is well-known that time integration by, e.g., Implicit Euler leads to $\Delta E_{\mathrm{bal}} \neq 0$. Implicit trapezoidal rule (sometimes known as Crank-Nicolson) gives smaller, but non-zero, error. Note that $\Delta E_{\mathrm{bal}} \neq 0$ is caused solely by the time discretization, even though the accuracy of right-hand side energy terms is affected by the error of spatial discretization.

IV. Energy Preserving DAE Integrator

A. Time Integration Method

The given DAE (6-7) is index-1, which is the simplest type of DAEs [10]. For this problem one can construct an energy preserving integrator by the so-called Gaussian collocation methods. These modified Runge-Kutta methods are discussed, e.g., in [11]. The variable $\boldsymbol{x}$ is considered to be an nth degree polynomial

$$\boldsymbol{x} = \begin{bmatrix} \boldsymbol{u} \\ \boldsymbol{v} \end{bmatrix} = \boldsymbol{p}(t) = \sum_{i=0}^{n} t^i \boldsymbol{c}_i \tag{17}$$

in each time step and we assume that the initial value $\boldsymbol{c}_0 = \boldsymbol{x}_0$ is given (from the initial state or previous time step), and that $\boldsymbol{x}_0$ satisfies the AE (5), i.e. it is consistent. For clarity, assume that the time step is in the interval $0 \le t \le h$.

The remaining coefficients $\boldsymbol{c}_1, \dots, \boldsymbol{c}_n$ are determined by the following procedure. For ODE we choose n nodes τ_i, $0 < \tau_1 < \dots < \tau_n < h$, to be the Gaussian quadrature points (giving the most accurate quadrature for a given n), and we integrate ODE (6) by using this quadrature. Algebraic equation (7) is assumed to be satisfied at $t = 0$, and it is required to hold at $t = h$. If $n > 1$, AE is in addition required to hold as certain weighted averages of the Gaussian points. This construction avoids oscillation in the algebraic component $\boldsymbol{v}$ that may occur when Gaussian points are used for both ODE and AE. For the methods considered in this paper the quadrature points are

$$\begin{aligned} n = 1: &\quad \tau_1 = \frac{1}{2} \\ n = 2: &\quad \tau_1 = \frac{1}{2} - \frac{\sqrt{3}}{6}, \tau_2 = \frac{1}{2} + \frac{\sqrt{3}}{6}. \end{aligned}$$

By substituting $\boldsymbol{p}$ into (6-7), the system of equations to be solved w.r.t. $\boldsymbol{c}_i$ is

$$\boldsymbol{M}\dot{\boldsymbol{p}}(\tau_i) = \boldsymbol{f}(\boldsymbol{p}(\tau_i), \tau_i) \tag{18}$$

$$0 = \boldsymbol{g}(\boldsymbol{p}(h)), \tag{19}$$

$$0 = \sum_{j=1}^{n} d_j \boldsymbol{g}(\boldsymbol{p}(\tau_j)), \quad \text{if } n > 1. \tag{20}$$

For $n = 2$ the weights are simply $d_1 = d_2 = \frac{1}{2}$. See [11] for the higher order methods. These equations determine the coefficients $\boldsymbol{c}_i$ uniquely. After solving $\boldsymbol{c}_i$, $\boldsymbol{p}(h)$ is used as the initial value for the next time step. Note that Equation (19) will assert that the next time step starts from a consistent state.

The simplest method, $n = 1$, reads just as "use Implicit Midpoint method for the ODE and require the AE to hold at the endpoint". The next method, $n = 2$ is "use 2-point Gauss for the ODE, AE at endpoint and as average of Gauss points". The order of these methods is $2n$, meaning that the solution error $\|\boldsymbol{p}(t_i) - \boldsymbol{x}_{\text{exact}}(t_i)\| = \mathcal{O}(h^{2n})$, where $t_i = ih$. In principle, these methods can be used as conventional Runge-Kutta methods without considering any collocation polynomials, but this polynomial interpretation provides an easy way to understand the energy preservation in the next section.

The advantage of these methods is that in a linear problem the changes of energies will sum up exactly to zero at each time step, or in a conservative system (without losses) the method preserves energy. The powers P_{in} and P_{loss}, as defined in the previous section, involve the time derivative of $\boldsymbol{x}$, so the corresponding change of energy over one time step,

$$\Delta E_{\text{loss}} = \int_0^h P_{\text{loss}}\, dt,$$

could be considered to be an "ODE" and we use the same Gauss quadrature formula to compute the integral.

The number of unknowns and function evaluations of the modified midpoint method ($n = 1$) is essentially the same as with the popular Backward Euler and Crank-Nicolson methods (note that in Crank-Nicolson (10) the term f_i is already computed during the previous time step). The second method ($n = 2$) is essentially a 2-step Implicit Runge-Kutta method, and the system to be solved at each time step is twice as large as for the $n = 1$-methods.

B. Energy Balance Over One Time Step

We show that the methods described in Section IV.A give an exact energy balance on each time step if the problem is linear. The heuristics is as follows: if we use an energy-preserving scheme to solve the DAE and then (at post-processing) the same scheme to compute the energy changes, then the numerical scheme has an exact energy balance.

Consider a linear DAE

$$\begin{bmatrix} \boldsymbol{M} & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \dot{\boldsymbol{u}} \\ \dot{\boldsymbol{v}} \end{bmatrix} + \begin{bmatrix} \boldsymbol{K}_{uu} & \boldsymbol{K}_{uv} \\ \boldsymbol{K}_{vu} & \boldsymbol{K}_{vv} \end{bmatrix} \begin{bmatrix} \boldsymbol{u} \\ \boldsymbol{v} \end{bmatrix} = \begin{bmatrix} \boldsymbol{j} \\ 0 \end{bmatrix}. \tag{21}$$

Matrix $\boldsymbol{K}$ is symmetric, so $\boldsymbol{K}_{uv} = \boldsymbol{K}_{vu}^{\mathrm{T}}$. Formally (but not in practice for a large system) we can solve $\boldsymbol{v}$ by

$$\boldsymbol{v} = -\boldsymbol{K}_{vv}^{-1} \boldsymbol{K}_{vu} \boldsymbol{u}$$

and substitute it into the ODE to get an equation only for $\boldsymbol{u}$,

$$\boldsymbol{M}\dot{\boldsymbol{u}} + \boldsymbol{K}_{uu}\boldsymbol{u} - \boldsymbol{K}_{uv}\boldsymbol{K}_{vv}^{-1}\boldsymbol{K}_{vu}\boldsymbol{u} = \boldsymbol{f}.$$

If we denote

$$\tilde{\boldsymbol{K}} = \boldsymbol{K}_{uu} - \boldsymbol{K}_{uv}\boldsymbol{K}_{vv}^{-1}\boldsymbol{K}_{vu}, \tag{22}$$

the ODE can be written as

$$\boldsymbol{M}\dot{\boldsymbol{u}} + \tilde{\boldsymbol{K}}\boldsymbol{u} = \boldsymbol{f}. \tag{23}$$

In this notation the magnetic field energy in both conducting and non-conducting regions is the quadratic form

$$E_{\text{mag}} = \tfrac{1}{2}\boldsymbol{u}^{\mathrm{T}}\tilde{\boldsymbol{K}}\boldsymbol{u}. \tag{24}$$

Now, we solve (21) by the simplest of above constructed methods, requiring ODE at midpoint and AE at endpoint. Assume that the initial condition is consistent. Denote the known initial values by $\boldsymbol{u}_0, \boldsymbol{v}_0$ and unknown values at the next time step by $\boldsymbol{u}_1, \boldsymbol{v}_1$. The equations to be solved, (18-20), including the initial consistency, are now explicitly

$$\boldsymbol{M}\frac{\boldsymbol{u}_1 - \boldsymbol{u}_0}{h} + \boldsymbol{K}_{uu}\frac{\boldsymbol{u}_1 + \boldsymbol{u}_0}{2} + \boldsymbol{K}_{uv}\frac{\boldsymbol{v}_1 + \boldsymbol{v}_0}{2} = \boldsymbol{f}(\tfrac{h}{2}) \tag{25}$$

$$\boldsymbol{K}_{vu}\boldsymbol{u}_1 = -\boldsymbol{K}_{vv}\boldsymbol{v}_1 \tag{26}$$

$$\boldsymbol{K}_{vu}\boldsymbol{u}_0 = -\boldsymbol{K}_{vv}\boldsymbol{v}_0. \tag{27}$$

In practice we solve $(\boldsymbol{u}_1, \boldsymbol{v}_1)$ from (25-26) as such. But now, formally, the last two rows give

$$\frac{\boldsymbol{v}_1+\boldsymbol{v}_0}{2} = -\boldsymbol{K}_{vv}^{-1}\boldsymbol{K}_{vu}\frac{\boldsymbol{u}_1+\boldsymbol{u}_0}{2},$$

which we substitute into the first equation to get

$$\boldsymbol{M}\frac{\boldsymbol{u}_1-\boldsymbol{u}_0}{h} + \tilde{\boldsymbol{K}}\frac{\boldsymbol{u}_1+\boldsymbol{u}_0}{2} = \boldsymbol{f}(h/2). \tag{28}$$

At post-processing the input energy $\int_0^h \dot{\boldsymbol{u}}^{\mathrm{T}}\boldsymbol{f}\,dt$ and resistive loss $\int_0^h \dot{\boldsymbol{u}}^{\mathrm{T}}\boldsymbol{M}\dot{\boldsymbol{u}}\,dt$ are computed by the same midpoint rule. To compute the change of energies over one time step $0 \le t \le h$, we multiply (28) by $\dot{\boldsymbol{u}}^{\mathrm{T}}$ at the midpoint (which in this case is just $\frac{\boldsymbol{u}_1-\boldsymbol{u}_0}{h}$), and multiply by the interval length h

$$\begin{aligned}0 &= h\Big(\frac{\boldsymbol{u}_1^{\mathrm{T}}\text{-}\boldsymbol{u}_0^{\mathrm{T}}}{h}\boldsymbol{M}\frac{\boldsymbol{u}_1\text{-}\boldsymbol{u}_0}{h} + \frac{\boldsymbol{u}_1^{\mathrm{T}}\text{-}\boldsymbol{u}_0^{\mathrm{T}}}{h}\tilde{\boldsymbol{K}}\frac{\boldsymbol{u}_1\text{+}\boldsymbol{u}_0}{2} \\ &\quad - \frac{\boldsymbol{u}_1^{\mathrm{T}}\text{-}\boldsymbol{u}_0^{\mathrm{T}}}{h}\boldsymbol{f}(\tfrac{h}{2})\Big) \\ &= (\boldsymbol{u}_1^{\mathrm{T}}\text{-}\boldsymbol{u}_0^{\mathrm{T}})\boldsymbol{M}\frac{\boldsymbol{u}_1\text{-}\boldsymbol{u}_0}{h} + (\boldsymbol{u}_1^{\mathrm{T}}\text{-}\boldsymbol{u}_0^{\mathrm{T}})\tilde{\boldsymbol{K}}\frac{\boldsymbol{u}_1+\boldsymbol{u}_0}{2} \\ &\quad - (\boldsymbol{u}_1^{\mathrm{T}}\text{-}\boldsymbol{u}_0^{\mathrm{T}})\boldsymbol{f}(\tfrac{h}{2}) \\ &= (\boldsymbol{u}_1^{\mathrm{T}}\text{-}\boldsymbol{u}_0^{\mathrm{T}})\boldsymbol{M}\frac{\boldsymbol{u}_1\text{-}\boldsymbol{u}_0}{h} + \tfrac{1}{2}(\boldsymbol{u}_1^{\mathrm{T}}\tilde{\boldsymbol{K}}\boldsymbol{u}_1 - \boldsymbol{u}_0^{\mathrm{T}}\tilde{\boldsymbol{K}}\boldsymbol{u}_0) \\ &\quad - (\boldsymbol{u}_1^{\mathrm{T}}\text{-}\boldsymbol{u}_0^{\mathrm{T}})\boldsymbol{f}(\tfrac{h}{2}) \\ &= \Delta E_{\mathrm{loss}} + \Delta E_{\mathrm{mag}} - \Delta E_{\mathrm{in}}.\end{aligned}$$

This shows that the change of energies sum up to zero, and that the stored energy ΔE_{mag} is also integrated correctly.

The previous computation was for the simplest method having only one collocation point. The same result can be proven also for higher order Gaussian methods. This is done by considering the polynomial degree of the integrands in $\int_0^h \dot{\boldsymbol{p}}^{\mathrm{T}}\tilde{\boldsymbol{K}}\boldsymbol{p}\,dt$, $\int_0^h \dot{\boldsymbol{p}}^{\mathrm{T}}\boldsymbol{f}\,dt$ and $\int_0^h \dot{\boldsymbol{p}}^{\mathrm{T}}\boldsymbol{M}\dot{\boldsymbol{p}}\,dt$ and using the fact that n-point Gaussian quadrature integrates polynomial of degree $2n-1$ exactly. The polynomial degree of $\boldsymbol{x}$ is n, and the integrands' degree is at most $2n-1$, so the change in stored energy is computed exactly by this formula, and all the terms will sum up to zero due to collocation condition (18-19) being satisfied at the integration points.

The exact energy balance does not hold for a nonlinear magnetic material, e.g., if the iron core reluctance is a spline function of form $\nu(|\boldsymbol{B}|^2)$. Then the magnetic field energy is no more quadratic, but instead some piecewise function $E(|\boldsymbol{B}|^2)$. The reasoning in the previous paragraph will fail, regardless of the time interation method's order. Still, one may expect better balance compared to methods where even the quadratic part is not balanced.

C. Numerical Evidence

The numerical error of solution $\boldsymbol{x}$ and energy balance ΔE_{bal} are studied in test simulations with different time steps h. The proposed methods are compared to conventional Backward Euler and Crank-Nicolson methods. The simulated motor is a voltage supplied star-connected locked-rotor 37-kW induction machine. The magnetic field equation is coupled to the stator and rotor circuit equations, so in addition to finite element DoFs, we have rotor circuit voltages, and rotor and stator circuit currents as unknowns, see [15] for details. With these additional equations and variables the system remains in form (6 - 7), so it can be solved by using the previously constructed integration methods. Also, the energy balance considerations remain essentially the same with the coupled equations.

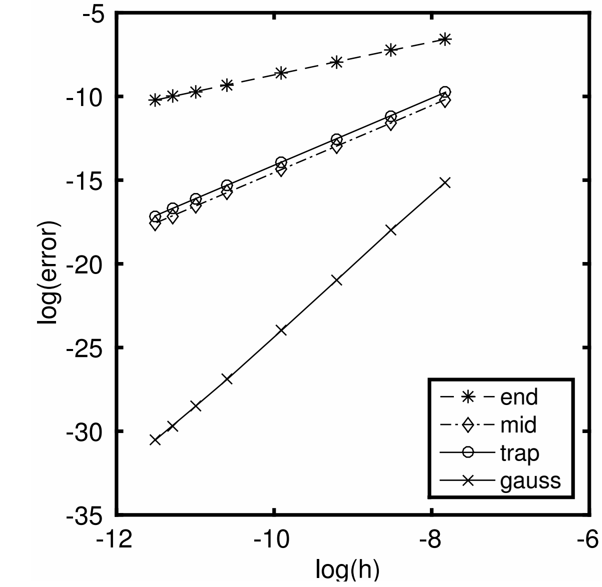

Figure 1. Linear problem, solution error vs time step length h. The legend refers to Implicit Euler, (modified) Midpoint, Trapezoidal and (modified) Gauss method, respectively. Convergence rates are h, h^2, h^2 and h^4. The errors for a nonlinear problem are very similar.

The previous reasoning applied to trapezoidal rule (Crank-Nicolson) will not give exactly $E_{\mathrm{bal}} = 0$. Instead, if we use the trapezoidal rule both for DAE-solving and power computation we will have $E_{\mathrm{bal}} = \mathcal{O}(h^3)$ over one time step, which leads to $E_{\mathrm{bal}} = \mathcal{O}(h^2)$ over a time interval $[0, Nh]$. If we integrate the energies by using backward differences, we will get only $E_{\mathrm{bal}} = \mathcal{O}(h)$. Backward Euler will give $E_{\mathrm{bal}} = \mathcal{O}(h)$, regardless of if we use mid- or endpoint for energy integration.

Fig. 1 shows how the error in the solution itself converges as the time step size h is refined. The solution error behaves as expected (Implicit Euler has order 1, midpoint/trapezoid has order 2, 2-point Gauss has order 4). Fig. 2 shows numerical evidence for the energy balance behaviour. For a nonlinear problem (in this case, spline $\nu(\boldsymbol{B})$) the energy balance is not exactly satisfied, as seen in Fig. 3. Still, the proposed methods give E_{bal} that is smaller by several orders of magnitude compared to Implicit Euler and Crank-Nicolson methods.

V. Rotation And Energy

In this section we show, both theoretically and numerically, how the torque computed from the energy balance is actually a difference approximation to the principle of virtual work applied on the remeshing layers. However, when small enough time steps are used, sliding nodes will cause oscillation and jumps in the torque obtained in this way if remeshing is used. The oscillation is a numerical artefact, as it occurs in the same frequency as the nodes slide past each other. The jumps can

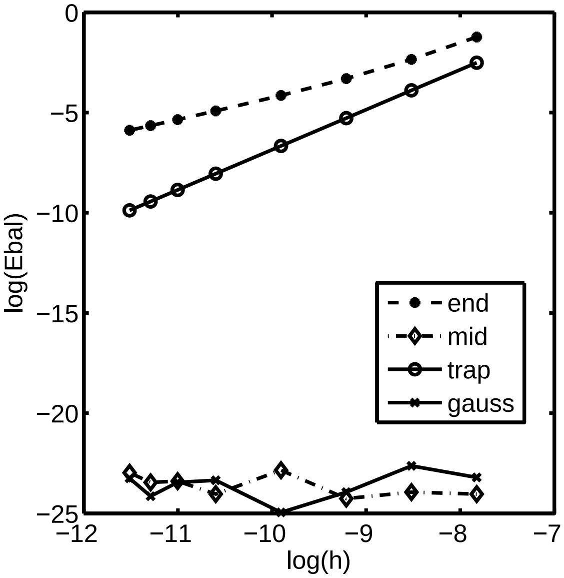

Figure 2. Linear problem, energy balance vs time step length. Midpoint and Gauss methods give practically exact balance. Trapezoid for time stepping (and energy integration) gives $E_{\mathrm{bal}} = \mathcal{O}(h^2)$. Implicit Euler for time stepping gives $E_{\mathrm{bal}} = \mathcal{O}(h)$.

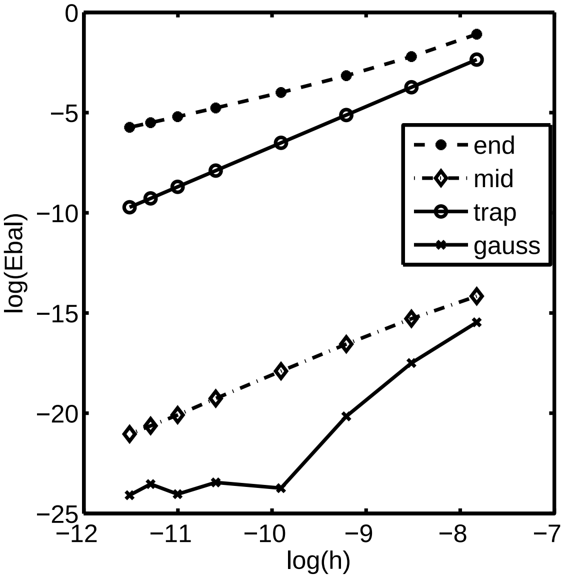

Figure 3. Nonlinear problem, energy balance vs. time step length.

be avoided if Nitsche's method is used instead of remeshing, even though the oscillation remains.

Based on numerical tests we conclude that the energy balance method shares the oscillation property of using the sliding layer for torque computation. Additionally, the method has two more error sources. First comes from the numerical energy balance, and second from the method being a difference approximation to a quantity which can be computed exactly with, e.g., Coulomb's method.

A. Remeshing and Nitsche's method

The rotor field is solved in coordinates which rotate along with the material, called Lagrangian coordinates in e.g. fluid mechanics, and the stator field is solved in fixed coordinates. Computationally one can think that we have two separate meshes (or two separate finite element problems) for rotor and stator, and then we somehow connect these two components for a given rotation angle. The air gap is governed by Laplace's equation $\Delta A = 0$, boundary conditions arising from continuity over the rotor and stator boundaries. Note that there are no time derivatives involved in the air gap, so the rotation-dependent component is contained within the constraint terms in Equation (4 - 5). The rotor and stator meshes are connected in our example either by remeshing or Nitsche's method. Remeshing is implemented by the sliding-layer technique.

Remeshing causes jumps in the stiffness matrix $\boldsymbol{K}$ when the triangle connectivity changes, and this causes problems if the torque is computed by the energy balance method, as we will see later. To avoid these jumps, we introduce Nitsche's method for non-matching grids very briefly, for a rigorous treatment see [16]. The method is an example of a mortar method, see e.g. [13], [14] for other similar methods. One can think that there are two separate meshes, one for the rotor and one for the stator, both including a part of the air gap. At a circular (or cylindrical in 3D) interface Γ, which is somewhere inside the air gap, the nodes are not necessarily matching but lie on the same circle, as in Fig. 4. FE matrix elements associated with area integrals inside rotor and stator do not depend on the rotation angle θ. Instead, we have some θ-dependent matrix which involves basis functions at triangles that have a part of Γ as their edge.

The solution of Laplace's equation w has to satisfy

$$\tilde{a}(w, v) = 0 \tag{29}$$

for each air gap basis function v, where

$$\tilde{a}(w, v) = (\nabla w, \nabla v)_{\Omega}, \tag{30}$$

$(\cdot, \cdot)_{\Omega}$ denoting the area integral over the whole domain Ω. However, we allow the field to be discontinuous on Γ, and Nitsche's method is one way to implement it. Denote the rotor and the stator side domains by Ω_1 and Ω_2, the corresponding mesh edges by G_1 and G_2, corresponding edge length by h_E (where $E \in G_i$), and the interface between the domains by Γ. We allow the mesh to be nonconforming and the solution w to be discontinuous over Γ, and using Nitsche's method the bilinear form (30) is replaced by

$$a(w, v) = \sum_{i=1,2} \left[(\nabla w, \nabla v)_{\Omega_i} + \gamma \sum_{E \in G_i} h_E^{-1} \langle [\![w]\!], [\![v]\!] \rangle_E \right] - \left\langle \left\{ \frac{\partial w}{\partial n} \right\}, [\![v]\!] \right\rangle_{\Gamma} - \left\langle \left\{ \frac{\partial v}{\partial n} \right\}, [\![w]\!] \right\rangle_{\Gamma}, \tag{31}$$

where $\langle u, v \rangle$ denotes a line integral, $[\![u]\!]$ the jump $u_1 - u_2$ of the function u over the interface, $\{u\}$ the average $\frac{1}{2}(u_1 + u_2)$ on the interface, and $\frac{\partial}{\partial n}$ the normal derivative. The parameter $\gamma > 0$ has to be chosen to be large enough to make the method stable, see Remark 2.12 in [16]. If we denote the stator side

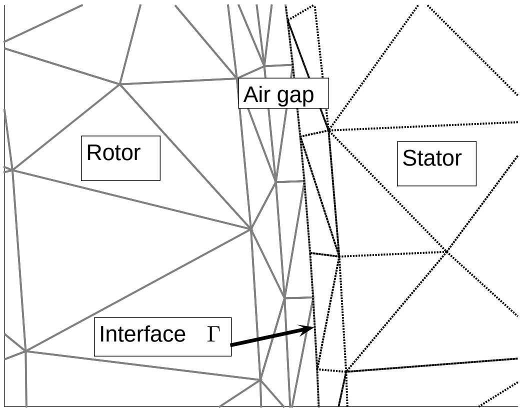

Figure 4. A nonconforming mesh in the air gap, which is used by the Nitsche's method. The interface Γ is the circle arc with hanging nodes.

basis functions by s_i and rotor side by r_i, we see that $a(s_i, s_j)$ and $a(r_i, r_j)$ do not depend on the rotor angle θ, whereas terms $a(s_i, r_j) = a(r_j, s_i)$ do depend. Let us denote this rotor angle-dependent part of the stiffness matrix by $\boldsymbol{K}_\theta$.

Nitsche's method for Laplace equation is specifically constructed such that the resulting matrix $\boldsymbol{K}_\theta$ is symmetric and positive-definite, so it preserves properties of the ambient problem. Thus we can also speak of energy E_{nit} stored in the Nitsche term,

$$E_{\mathrm{nit}} = \tfrac{1}{2}\boldsymbol{x}^{\mathrm{T}}\boldsymbol{K}_\theta\boldsymbol{x}, \qquad (32)$$

where $\boldsymbol{K}_\theta$ includes only the last three terms (edge-integrals) of (31). E_{nit} is considered to be some numerical energy and it can be shown that E_{nit} tends to zero as the mesh is refined [16], because the jump $\llbracket u \rrbracket$ tends to zero. One can also check numerically that the ratio $E_{\mathrm{nit}}/E_{\mathrm{mag}}$ is indeed small. However, if we consider numerical energy balance, the term E_{nit} should be included in the equations.

We have similar θ-dependent term when remeshing is considered,

$$E_{\mathrm{rem}} = \tfrac{1}{2}\boldsymbol{x}^{\mathrm{T}}\boldsymbol{K}_\theta\boldsymbol{x} = \tfrac{1}{2}\int_{\mathrm{rem}} \nu_0\boldsymbol{B}^2, \qquad (33)$$

but this is the actual magnetic field energy in the remeshing layer.

B. Energy Balance Method's Relation to Virtual Work Principle

We explain how torque computation by energy balance is related to the principle of virtual work. This will also show how the energy balance method is dependent on the smoothness of $\boldsymbol{K}_\theta$ with respect to θ. Let us add rotation to the linear problem and assume a constant speed ω_m. The rotor angle-dependent discretized field equation can be written as

$$\boldsymbol{K}\boldsymbol{x} + \boldsymbol{K}_\theta\boldsymbol{x} + \boldsymbol{M}\dot{\boldsymbol{x}} = \boldsymbol{f}(t),$$

where $\boldsymbol{K}_\theta$ denotes the matrix which changes with the rotor angle θ, coming from either remeshing or Nitsche's method. Assume for a while that we study small enough time interval where triangulation does not change, so $\boldsymbol{K}_\theta$ is differentiable with respect to θ. Multiplying the previous field equation by $\dot{\boldsymbol{x}}^{\mathrm{T}}$ leads to a power balance equation

$$\dot{\boldsymbol{x}}^{\mathrm{T}}\boldsymbol{K}\boldsymbol{x} + \dot{\boldsymbol{x}}^{\mathrm{T}}\boldsymbol{K}_\theta\boldsymbol{x} + \dot{\boldsymbol{x}}^{\mathrm{T}}\boldsymbol{M}\dot{\boldsymbol{x}} - \dot{\boldsymbol{x}}^{\mathrm{T}}\boldsymbol{f}(t) = 0. \qquad (34)$$

The stored energy at remeshing (or Nitsche) layer is

$$W_{\mathrm{rem}} = \tfrac{1}{2}\boldsymbol{x}^{\mathrm{T}}\boldsymbol{K}_\theta\boldsymbol{x}, \qquad (35)$$

but the rate of its change is not $\dot{\boldsymbol{x}}^{\mathrm{T}}\boldsymbol{K}_\theta\boldsymbol{x}$, because the matrix $\boldsymbol{K}_\theta$ is not constant over time (unless we have a stationary rotor).

However, we see that (note $\boldsymbol{K}_\theta = \boldsymbol{K}_\theta^{\mathrm{T}}$ if we use remeshing or Nitsche)

$$\tfrac{d}{dt}\tfrac{1}{2}\boldsymbol{x}^{\mathrm{T}}\boldsymbol{K}_\theta\boldsymbol{x} = \dot{\boldsymbol{x}}^{\mathrm{T}}\boldsymbol{K}_\theta\boldsymbol{x} + \tfrac{1}{2}\dot{\theta}\boldsymbol{x}^{\mathrm{T}}\frac{d\boldsymbol{K}_\theta}{d\theta}\boldsymbol{x}, \qquad (36)$$

where

$$T_{\mathrm{vw}} = -\tfrac{1}{2}\boldsymbol{x}^{\mathrm{T}}\frac{d\boldsymbol{K}_\theta}{d\theta}\boldsymbol{x} \qquad (37)$$

is actually the torque computed by virtual work principle at the remeshing layer, which can be computed e.g. by Coulomb's method [4]. Numerical experiments show that (37) gives a sensible estimate for the torque even if $\boldsymbol{K}_\theta$ emerges from Nitsche's method. The reason for this can be seen by computing the derivative $\dfrac{\partial}{\partial\theta}a(u,u)$ and using estimates in [16]. Then one can obtain an estimate

$$\tfrac{\partial}{\partial\theta}a(u,u) = -2\left\langle R\tfrac{\partial}{\partial\theta}u, \tfrac{\partial}{\partial r}u\right\rangle_\Gamma + \mathcal{O}(h_T^{1/2}), \qquad (38)$$

where the first term is exactly the integral of Maxwell stress tensor over Γ, and h_T denotes the mesh parameter of the triangulation. For other mortar methods a relation like (38) is not guaranteed.

The previous explains how torque is related to the power balance equation (34), which combined with (36) gives

$$\begin{aligned} &\dot{\boldsymbol{x}}^{\mathrm{T}}\boldsymbol{K}\boldsymbol{x} + \tfrac{d}{dt}\tfrac{1}{2}\boldsymbol{x}^{\mathrm{T}}\boldsymbol{K}_\theta\boldsymbol{x} - \dot{\theta}T_{\mathrm{vw}} + \dot{\boldsymbol{x}}^{\mathrm{T}}\boldsymbol{M}\dot{\boldsymbol{x}} - \dot{\boldsymbol{x}}^{\mathrm{T}}\boldsymbol{f}(t) \\ =&\tfrac{d}{dt}W_{\mathrm{mag}} + \tfrac{d}{dt}W_{\mathrm{rem}} + P_{\mathrm{torq}} + P_{\mathrm{loss}} - P_{\mathrm{in}} \\ =&0. \end{aligned} \qquad (39)$$

In other words, assuming that $\boldsymbol{K}_\theta$ is smooth enough with respect to θ, the energy balance method leads to the same value as Coulomb's method applied in remeshing layer (or its equivalent when Nitsche's method is used) as the time step is refined. Note that we assumed $\boldsymbol{K}_\theta$ to be differentiable with respect to θ. This is not true when remeshing is used, and its effect is seen in the numerical examples.

C. Computational Cost of Torque Computation

Computing the torque by Maxwell stress tensor or Coulomb's method involves integrating a quadratic function of the magnetic field $\boldsymbol{B}$ in the air gap. Because the function is quadratic, we can write the torque as a product

$$T = \boldsymbol{x}^{\mathrm{T}}\boldsymbol{V}\boldsymbol{x}, \qquad (40)$$

where the matrix $\boldsymbol{V}$ is constant throughout the simulation if we use only fixed (non-sliding) air gap layers in the integration.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMAG.2016.2537263, IEEE Transactions on Magnetics

Once the matrix $\boldsymbol{V}$ is assembled in the beginning of the simulation, the computational cost of torque computation (one vector-matrix-vector product involving only air gap elements) is very small compared to the solution of implicit equations arising from the time stepping.

The energy balance method involves computation of the terms in (39), which is more expensive, but still cheap compared to time stepping.

Assuming that the air gap elements are in circular layers and by integrating the Maxwell stress tensor over one layer [15, Eq. 124], the matrix $\boldsymbol{V}$ in (40) is

$$\boldsymbol{V}_{ij} = \frac{\nu_0}{r_0 - r_1} \int_S r(\nabla \times (v_i \boldsymbol{e}_z))_r \cdot (\nabla \times (v_j \boldsymbol{e}_z))_\phi,$$

where the integration is over an air gap layer S between radii r_0 and r_1, ν_0 is the reluctivity of air, and $(\nabla \times (v_i \boldsymbol{e}_z))_r$ and $(\nabla \times (v_j \boldsymbol{e}_z))_\phi$ are the radial and tangential components of the curls of the basis functions v_i and v_j.

For Coulomb's method [4] the corresponding matrix is

$$\boldsymbol{V}_{ij} = \nu_0 \int_S \Big[-(\nabla \times (v_i \boldsymbol{e}_z))^{\mathrm{T}} \boldsymbol{G}^{-1} \frac{\partial \boldsymbol{G}}{\partial \phi} (\nabla \times (v_j \boldsymbol{e}_z)) + \frac{1}{2} (\nabla \times (v_i \boldsymbol{e}_z)) \cdot (\nabla \times (v_j \boldsymbol{e}_z)) |\boldsymbol{G}|^{-1} \frac{\partial |\boldsymbol{G}|}{\partial \phi} \Big], \quad (41)$$

where $\boldsymbol{G}$ is the affine mapping from the mesh element into the reference element and $|\boldsymbol{G}|$ its determinant.

D. Numerical Examples

We assume linear magnetic materials and constant rotation speed in each example, as that is sufficient to show the interesting features of torque computation by energy balance method compared to other methods. These features are seen when the rotor angle change per time step is comparable to angle between nodes in the sliding layer, so a slow speed (slip $s = 0.5$) is chosen. Second order elements are used, such that one quarter of the simulated machine contains 15734 basis functions. The torque computation methods are

- T_{vw1}, Coulomb's method applied on a rotor side layer;
- T_{vw2}, Coulomb's method applied on a stator side layer;
- T_{rem}, Coulomb's method applied on the sliding layer (not applicable in Nitsche's method);
- T_{ene}, energy balance method.

In Example I, Fig. 5, we use remeshing with sliding layer technique, the modified midpoint method (Crank-Nicolson giving very similar results) and comparatively long time step, such that the supply voltage period is divided into $n_{\mathrm{per}} = 200$ steps. The difference between torques obtained from different method is up to 5 percent. In Example II, Fig. 6, we use Implicit Euler method, instead of midpoint or Crank-Nicolson, with the same time step. Energy balance method overestimates the torque because of the numerical energy loss.

Example III, Fig. 7, has the same setup as Example I, but one supply voltage period is divided into $n_{\mathrm{per}} = 1200$ steps. Now the rotor angle change per time step, $\Delta\theta$, is several times smaller than the angle between adjacent remeshing layer nodes, $\Delta\alpha$. The following observations are made:

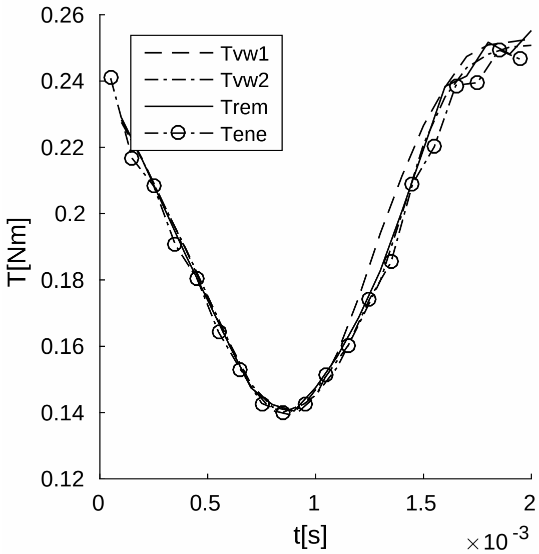

Figure 5. Example I, remeshing.

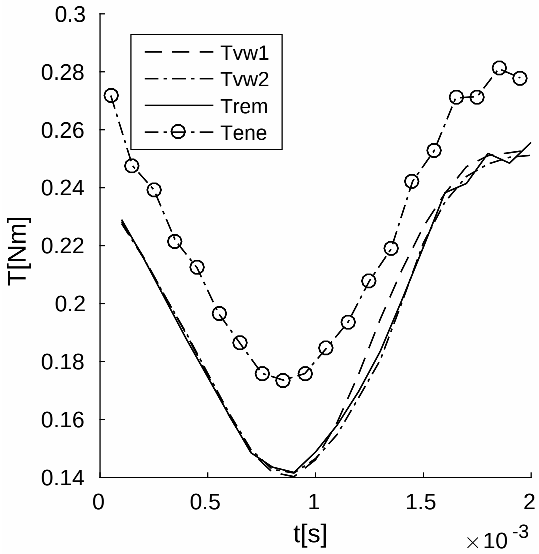

Figure 6. Example II, remeshing, Implicit Euler for time integration. Energy balance method overestimates the torque when the time step is large.

- T_{vw1} and T_{vw1} are not affected significantly by the mesh deformation.
- T_{rem} is affected by the mesh deformation and oscillates with frequency corresponding to $\Delta\alpha$.
- T_{ene} resembles T_{rem}, but has discontinuities when the mesh connectivity changes (i.e. when θ is such that $\boldsymbol{K}_\theta$ is discontinuous).

The effect of Nitsche's method in energy balance method is seen in Fig. 8. The setup is otherwise the same as in Example III, but Nitsche's method is used. Because $\boldsymbol{K}_\theta$ is smoother with respect to θ (meaning that the matrix elements change continuously when θ changes, as opposed to remeshing) the resulting torque waveform will be smoother, and the oscillation has smaller amplitude. If one uses coarser mesh or linear elements instead of quadratic ones, then the oscillation and jumps are even more prominent.

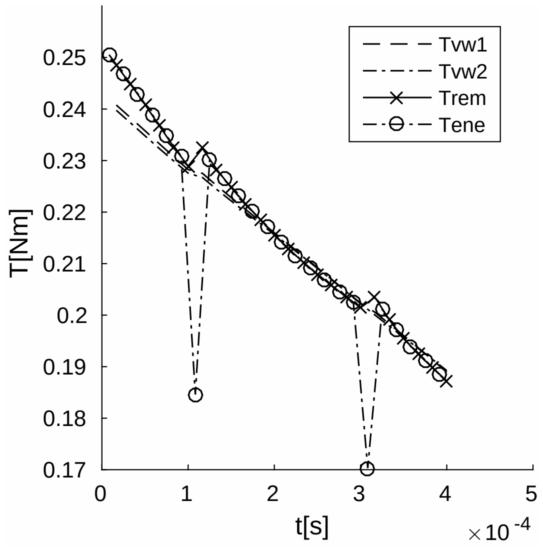

Figure 7. Example III, remeshing, smaller time step. Oscillation occurs as the nodes slide past each other.

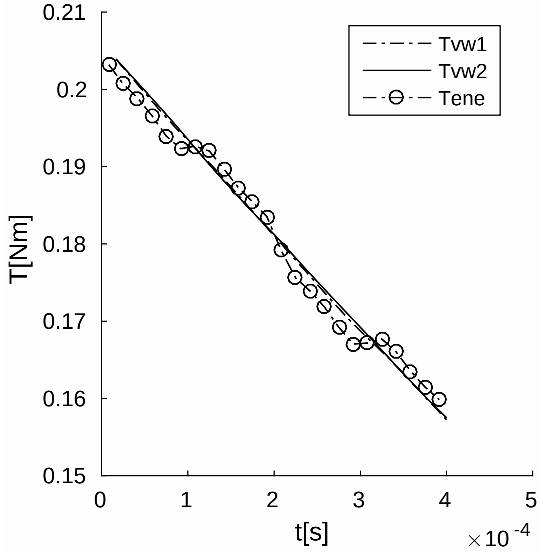

Figure 8. Example IV, Nitsche, smaller time step.

VI. Conclusion

It is possible to construct a time integration scheme that is suitable for DAEs encountered in electric machine simulations and is exactly energy-balanced at every time step, at least when a linear and non-moving system is considered. The simplest of the methods, based on Implicit Midpoint method, is as accurate as the common Crank-Nicolson method and requires a similar amount of computational effort.

The energy balance method for torque computation is shown to be a difference approximation to the virtual work principle applied on the remeshing layer. The resulting torque waveform will have larger and larger discontinuities when the time step is refined. This is due to a jump in magnetic field energy when remeshing changes the triangulation. A smoother waveform can be obtained if Nitsche's method, an instance of discontinuous Galerkin methods, is used to implement the rotor motion.

Acknowledgment

Academy of Finland is acknowledged for financial support.

References

[1] A. Bossavit, "Force-related nuts and bolts in the discretization toolkit for electromagnetics," *Magnetics, IEEE Transactions on*, vol. 44, no. 6, pp. 1158–1161, 2008.

[2] ——, "Virtual power principle and maxwell's tensor: which comes first?" *COMPEL - The International Journal for Computation and Mathematics in Electrical and Electronic Engineering*, vol. 30, no. 6, pp. 1804–1814, 2011. [Online]. Available: http://dx.doi.org/10.1108/03321641111168110

[3] F. Henrotte, G. Deliége, and K. Hameyer, "The eggshell approach for the computation of electromagnetic forces in 2d and 3d," *COMPEL - The international journal for computation and mathematics in electrical and electronic engineering*, vol. 23, no. 4, pp. 996–1005, 2004. [Online]. Available: http://dx.doi.org/10.1108/03321640410553427

[4] J. Coulomb, "A methodology for the determination of global electromechanical quantities from a finite element analysis and its application to the evaluation of magnetic forces, torques and stiffness," *Magnetics, IEEE Transactions on*, vol. 19, no. 6, pp. 2514–2519, Nov 1983.

[5] N. Sadowski, Y. Lefevre, M. Lajoie-Mazenc, and J. Cros, "Finite element torque calculation in electrical machines while considering the movement," *Magnetics, IEEE Transactions on*, vol. 28, no. 2, pp. 1410–1413, Mar 1992.

[6] D. Dorrell, M. Popescu, and M. McGilp, "Torque calculation in finite element solutions of electrical machines by consideration of stored energy," in *Magnetics Conference, 2006. INTERMAG 2006. IEEE International*, May 2006, pp. 767–767.

[7] M. Marinescu and N. Marinescu, "Numerical computation of torques in permanent magnet motors by maxwell stresses and energy method," *Magnetics, IEEE Transactions on*, vol. 24, no. 1, pp. 463–466, Jan 1988.

[8] S. Niu, S. Ho, and W. Fu, "Power balanced electromagnetic torque computation in electric machines based on energy conservation in finite-element method," *Magnetics, IEEE Transactions on*, vol. 49, no. 5, pp. 2385–2388, May 2013.

[9] P. Kunkel and V. Mehrmann, *Differential-algebraic Equations: Analysis and Numerical Solution*, ser. EMS textbooks in mathematics. European Mathematical Society, 2006. [Online]. Available: https://books.google.fi/books?id=iRZPqCwkI_IC

[10] E. Hairer, S. Nørsett, and G. Wanner, *Solving Ordinary Differential Equations II: Stiff and Differential-Algebraic Problems*, ser. Lecture Notes in Economic and Mathematical Systems. Springer, 1993. [Online]. Available: http://books.google.fi/books?id=m7c8nNLPwaIC

[11] S. J. Small, "Runge-kutta type methods for differential-algebraic equations in mechanics," Ph.D. dissertation, University of Iowa, 2011. [Online]. Available: http://ir.uiowa.edu/etd/1082

[12] P. Hansbo, "Nitsche's method for interface problems in computa-tional mechanics," *GAMM-Mitteilungen*, vol. 28, no. 2, pp. 183–206, 2005. [Online]. Available: http://dx.doi.org/10.1002/gamm.201490018

[13] A. Buffa, Y. Maday, and F. Rapetti, "Calculation of eddy currents in moving structures by a sliding mesh-finite element method," *Magnetics, IEEE Transactions on*, vol. 36, no. 4, pp. 1356–1359, Jul 2000.

[14] F. Rapetti, F. Bouillault, L. Santandrea, A. Buffa, Y. Maday, and A. Razek, "Calculation of eddy currents with edge elements on non-matching grids in moving structures," *Magnetics, IEEE Transactions on*, vol. 36, no. 4, pp. 1351–1355, Jul 2000.

[15] A. Arkkio, "Analysis of induction motors based on the numerical solution of the magnetic field and circuit equations," Ph.D. dissertation, Helsinki University of Technology, 1987. [Online]. Available: http://lib.tkk.fi/Diss/198X/isbn951226076X/isbn951226076X.pdf

[16] R. Becker, P. Hansbo, and R. Stenberg, "A finite element method for domain decomposition with non-matching grids," *ESAIM: Mathematical Modelling and Numerical Analysis - Modélisation Mathématique et Analyse Numérique*, vol. 37, no. 2, pp. 209–225, 2003. [Online]. Available: http://eudml.org/doc/245245