Non-Conforming Sliding Interfaces for Relative Motion in 3D Finite Element Analysis of Electrical Machines by Magnetic Scalar Potential Formulation Without Cuts

Stefan Boehmer[1], Enno Lange[2], and Kay Hameyer[1]

[1]Institute of Electrical Machines, RWTH Aachen University, Aachen, D-52062 Germany
[2]CST-Computer Simulation Technology AG, Darmstadt, D-64289 Germany

This paper discusses non-conforming sliding interfaces for motion in combination with a magnetic scalar potential formulation. Lagrange multiplier are used to implement the relative motion of stator and rotor. The utilization of the specific Lagrange multiplier approach implies the application of a magnetic scalar potential formulation in 3D Finite Element (FE) modeling of electrical machines because up to the present a canonical definition of biorthogonal basis functions for the magnetic vector potential is not available.

Classical magnetic scalar potential formulations require the definition of cuts to make the potential single-valued. The presented approach uses a decomposition of the magnetic field into a scalar potential and loop fields defined on the whole domain to avoid the explicit definition of cuts.

***Index Terms*—Dual formulation, electrical machines, finite element methods, sliding interfaces.**

I. Introduction

NOWADAYS several approaches for handling relative motion of stator and rotor in FE analysis of electrical machines are available [1], [2]. Static, transient and particularly field coupling simulations of electrical machines require the flexible displacement of the rotor by an arbitrary angle in rotating machines or a distance in translational electric machines. In 2D FE modeling the Moving-Band method can be employed where an annulus-shaped band in the airgap between rotor and stator is remeshed in every time step [3]. In 3D FE modeling this approach is not feasible because it would require a full mesh generator whereas remeshing is done by a simple mapping in 2D. Thus, the Lockstep method [1] is usually applied in 3D which is based on a regular discretization of the rotor and the stator surfaces. The major disadvantage of this method is the lack of arbitrary displacement because step size is fixed by the discretization. As a consequence, a smooth movement leads to a significant increase in the number of elements resulting in an increase of computing time which is highly undesirable. Lagrange multiplier approaches seek to overcome the disadvantages being applicable to 2D as well as 3D problems [4], [5].

II. Motion by Lagrange Multiplier

The Lagrange multiplier method ensures the continuity of the fields across the non-conforming interface between the stationary and moving FE discretizations of the stator and rotor of the electrical machine. As a consequence an arbitrary displacement without restrictions in time or space discretization is possible. In general, the application of Lagrange multiplier methods yields a saddle point problem, which cannot be solved by standard Krylov-subspace algorithms. In order to preserve the numerical properties of a conforming approach, the FE-space of the discrete Lagrange multiplier is spanned by basis functions fulfilling the biorthogonality condition as described in [4] and applied to electromagnetic field computation in [2] and [6]. Thereby, the resulting symmetric positive definite system can be solved by Krylov-subspace algorithms. Previous studies have indicated that it is not feasible to construct such biorthogonal basis function in 3D for the magnetic vector potential formulation in a canonical way but for magnetic scalar potential $t-\omega$ formulation [2].

In the presented paper the authors hence utilize the magnetic scalar potential formulation shown in [7] and simplify the algorithm to compute the source fields on basis of spanning trees. Furthermore, this algorithm is extended to overcome possible termination issues. The resulting method is combined with the Lagrange multiplier approach presented by the authors in [2] and is applied to an exemplary electrical machine.

III. Topological Structure

In the classical $t-\omega$ formulation the field t is computed in the conducting region only whereas ω is computed in the whole domain ([8], [9]). This approach requires the definition of cuts in the non-conducting domain for multiple connected regions because otherwise ω may become multi-valued. In order to avoid these cuts the approach in this paper consists in decomposing the magnetic field h appropriately. The theoretical background has been presented in [7] and is recapitulated as far as necessary.

Let Ω be a connected mesh, $\Omega_{\mathrm{nc}} \subset \Omega$ the non-conducting domain, $\Omega_c \subset \Omega$ the domain of all conductors and $\partial\Omega_c$ its boundary as shown in Fig. 1. Furthermore, let $W^p(\Omega)$ be the set of differential forms of degree p which are defined on the domain Ω. In the non-conducting domain one has to solve $\operatorname{div} b = 0, \operatorname{curl} h = 0$ and $b = \mu_0 h$ with $b \in B(\Omega_{nc})$ and $h \in H(\Omega_{nc})$:

$$\begin{aligned} B(\Omega_{nc}) &= \{b \in W^2(\Omega_{nc}),\ (b - b^0) \cdot n = 0 \text{ on } \Gamma_b\} \\ H(\Omega_{nc}) &= \{h \in W^1(\Omega_{nc}), \operatorname{curl} h \cdot n = 0 \text{ on } \partial\Omega_{nc}, \\ &\qquad (h - h^0) \times n = 0 \text{ on } \Gamma_h\} \end{aligned} \tag{1}$$

While $\operatorname{div} b = 0$ can be satisfied by introduction of the magnetic vector potential a with $b = \operatorname{curl} a$, the situation is more complicated for $\operatorname{curl} h = 0$. The Poincaré lemma states that h can be represented by a continuous gradient field ($h = \operatorname{grad}\omega$) in

Manuscript received October 31, 2012; revised January 12, 2013; accepted January 13, 2013. Date of current version May 07, 2013. Corresponding author: S. Boehmer (e-mail: Stefan.Boehmer@IEM.RWTH-Aachen.de).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TMAG.2013.2242051

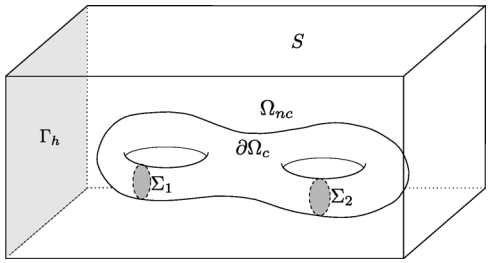


Fig. 1. Volumes and surfaces in the 3D domain Ω [7].

a contractible domain like a ball or a tetrahedron. If the domain contains holes and thus is not contractible, the topological structure of the functional space $W^1(\Omega_{nc})$ has to be considered. Let $B^1(\Omega_{nc})$ be the set of all gradients and $Z^1(\Omega_{nc})$ the set of all curl-free fields on Ω_{nc}. Following De Rham's theorem the quotient vector space $H^1(\Omega_{nc}) = Z^1(\Omega_{nc})/B^1(\Omega_{nc})$ is of finite dimension and equals the number of holes in Ω_{nc}, respectively the number of independent loops N_Σ formed by the conductors, also called Betti number β_1. The first cohomology group $H^1(\Omega_{nc})$ represents the set of curl-free 1-forms which cannot be expressed as a gradient of a 0-form.

The magnetic field in Ω_{nc} can then be expressed by

$$h = \sum_{k=0}^{N_\Sigma} I_k t^k + \text{grad}\ \omega \tag{2}$$

where I_k is the current in conductor loop k which is imposed in an independent section Σ_k of the conductor as seen in Fig. 1 and $\omega \in W(\Omega_{nc})$ is a continuous scalar potential without cuts defined on

$$W(\Omega_{nc}) = \{\omega \in W^0(\Omega_{nc}), \text{grad}\ \omega \times n = 0 \text{ on } \Gamma_h\}. \tag{3}$$

The loop fields $t^k \in H^1(\Omega_{nc})$ form a basis of the first cohomology group $H^1(\Omega_{nc})$ and are computed for every single Σ_k by imposing a unity current in Σ_k and zero current in all other $\Sigma_i \forall i \neq k$. Instead of a direct computation of $H^1(\Omega_{nc})$, which is in general a task of high complexity, we utilize the duality between cohomology and homology groups. The first homology group $H_1(\Omega_{nc}) = Z_1(\Omega_{nc})/B_1(\Omega_{nc})$ is defined as the set of all closed curves in Ω_{nc} which are not the boundary of any surface in Ω_{nc}. By De Rham's theorem the first homology group is isomorph to the first cohomology group $H_1(\Omega_{nc}) \cong H^1(\Omega_{nc})$.

The construction of the loop fields can hence be achieved by three single steps:

1) Generate a basis for the first homology group $H_1(\Omega_{nc})$ which is done by the definition of the surfaces Σ_k.
2) Build a spanning tree on Ω_{nc} corresponding to $H_1(\Omega_{nc})$.
3) For each Σ_k construct a basis of the first cohomology group $H^1(\Omega_{nc})$ which forms the according loop field.

IV. Constrained Spanning Tree

The presented approach of using spanning trees is a modified and extended version of [7] which is based on [10]. An (edge) spanning tree of a mesh is a subset of edges which does not contain any cycle and visits every node of the mesh.

To construct a constrained spanning tree, it is necessary to build a spanning tree on Ω which is also a spanning tree on all constrained volumes and surfaces. Therefore, the following constraints have to be considered:

- curl $t = 0$ on Ω_{nc}
- curl $t \cdot n = 0$ on $\partial\Omega_c$
- $t \times n = 0$ on S
- $(t - h^0) \times n = 0$ on Γ_h
- $\int_{e \in \partial\Sigma_k} t_e = I_k \forall k \in \{1, \ldots, N_\Sigma\}$

The algorithm to construct the spanning tree, which is described later in this section, works by iteratively appending edges to the tree which do not close a cycle and hence preserve the spanning tree property. To ensure that the spanning tree is also valid on the constrained volumes and surfaces, edges corresponding to these sets have to be put into the tree before other ones. This is ensured by assigning a priority to every edge in the mesh. Let us denote the set of volumes by $\{V_i, i = 1, \ldots, N_V\}$ and the set of the constrained surfaces by $\{S_i, i = 1, \ldots, N_S\} = \{\Gamma_h, S, \partial\Omega_c, \Sigma_k, k = 1, \ldots, N_\Sigma\}$. The set of edges corresponding to the volume V_i, or the surface S_i are denoted by L_{V_i} and L_{S_i} respectively and the priority of edge e is denoted by $prio[e]$. In contrast to the algorithm presented in [7] a simpler algorithm to compute the priority for all edges is proposed:

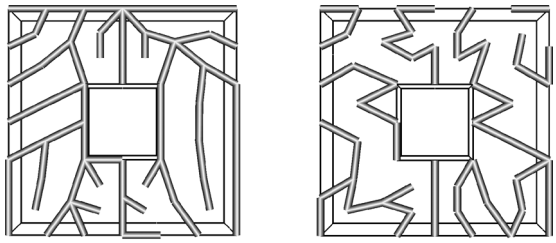
1) For each edge e in L_Ω (all edges):
   - $prio[e] := 0$
2) For each volume V_i:
   - For each edge e in L_{V_i}: $prio[e] + +$
3) For each surface S_i:
   - For each edge e in L_{S_i}: $prio[e] + +$

This algorithm ensures that a higher priority is assigned to an edge e_i which is contained in more volumes and surfaces than an edge e_j which is contained in fewer entities.

Afterwards, the constrained spanning tree can be constructed by the following straightforward algorithm which appends iteratively edges to the set L_{tree} representing the spanning tree. The set L_{nodes} contains all the nodes visited by the spanning tree:

1) $L_{\text{temp}} := \emptyset, L_{\text{tree}} := \emptyset, L_{\text{nodes}} := \emptyset$
2) Pick one arbitrary initial node n_i and append it to L_{nodes}
3) Append all edges associated with n_i to L_{temp}
4) While $L_{temp} \neq \emptyset$:
   - Pick and remove one edge $e_j = (n_i, n_k)$ with highest priority from L_{temp}
   - If end node n_k not in L_{nodes}:
     — Append n_k to L_{nodes}
     — Append e_i to L_{tree}
     — Insert all edges associated with n_k into L_{temp}

The termination of the algorithm is guaranteed by removing one edge from L_{temp} in every iteration of the while loop. As soon as L_{nodes} contains all nodes of the mesh we have a valid spanning tree in L_{tree} and no further edges are appended to L_{temp} which results in successful termination of the algorithm. The constructed spanning tree is not unique and is mainly influenced by the choice of the edge e_j in step four. Exploiting a first in first out (FIFO) stack for L_{temp} corresponds to breadth first search (BFS) for the spanning tree and results in a balanced spanning tree. Balanced spanning trees are favorable with respect to the later loop field construction in comparison to random trees or trees constructed by depth first search (DFS) ([7], [11]). Fig. 2(a) shows a spanning tree on top of the surface of a threedimensional conducting torus $\partial\Omega_c$ which has been constructed by BFS. In comparison to the spanning tree in Fig. 2(b) which has



(a) (b)

Fig. 2. Spanning tree on the top surface of a torus. (a) breadth first search, (b) depth first search.

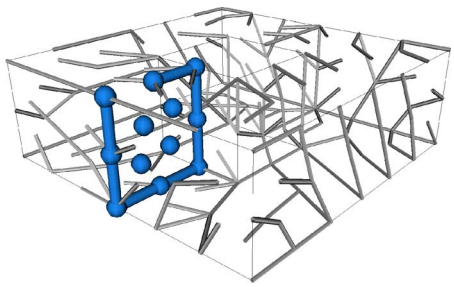


Fig. 3. Spanning tree on the surface of a torus with Σ_1.

been constructed by DFS, one can observe that BFS results in much more nodes with branching edges.

V. Loop Fields

After the construction of the spanning tree on Ω_{nc} the loop fields t_k can be build which have to fulfill the condition curl $t = 0$ on the non-conducting domain Ω_{nc} using edge-based Whitney elements. This is done iteratively for every Σ_k by imposing a unity current in Σ_k and zero current in all other $\Sigma_i \forall i \neq k$. By construction of the spanning tree there is exactly one edge for every set $\Sigma_i \cap \partial\Omega_c$ which is not contained in the tree. This is illustrated in Fig. 3 where Σ_1 is highlighted by exposing its nodes by spheres and all the tree edges in $\Sigma_1 \cap \partial\Omega_c$ being drawn by bold edges. It can be seen, that the top left edge of $\Sigma_1 \cap \partial\Omega_c$ is the only edge which is not contained in the spanning tree. Therefore, it is possible to impose unity current by assigning a unity value to the edge e_k with $e_k \in \Sigma_i \cap \partial\Omega_c \wedge e_k \notin L_{tree}$. To compute the desired loop fields t_k, it is necessary to satisfy the curl condition for all faces in the non-conducting region Ω_{nc}:

$$\sum_{e_i \in \partial f_j} t_{e_i} = 0, \forall f_j \in L_F(\Omega_{nc}) \tag{4}$$

where $L_F(\Omega_{nc})$ denotes all faces of elements in Ω_{nc}. Equation (4) can be transformed to a linear system of equations with unknown t_{e_i} for every edge e_i and one equation for every face $L_F(\Omega_{nc})$. This system of equation does not have full rank, there is an infinite number of possible loop fields t_k, but the kernel of the system of equations can be eliminated by fixing all edge values t_{e_i} corresponding to spanning tree edges $e_i \in L_{\text{tree}}$ [12]. This results in a unique solution of the system of equations and thus to a unique loop field t_k.

Instead of utilizing an equation solver, the algorithm to compute the loop field t_k for Σ_k works by backsubstitution as follows (cf. [7]):

1) Set all values of tree edges to zero: $t_{e_i} := 0, \forall e_i \in L_{\text{tree}}$
2) For each i in $\{1, \ldots, N_\Sigma\} \backslash \{k\}$:
   - Set $t_{e_j} := 0$ for $e_j \in \Sigma_i \cap \partial\Omega_c \wedge e_j \notin L_{\text{tree}}$

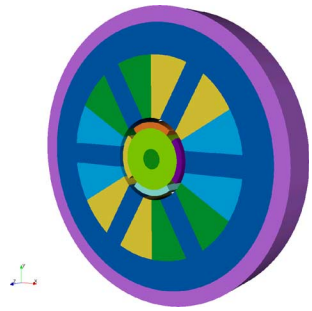


Fig. 4. Application of the presented approach to an example PMSM. (a) non-conforming discretization in the airgap, (b) computed magnetic flux density.

3) Set $t_{e_k} := 1$
4) Put all faces of Ω_{nc} into the list $L_F(\Omega_{nc})$
5) While $L_F \neq \emptyset$:
   - Pick and remove one face f_i from $L_F(\Omega_{nc})$ which has two edges fixed
   - Fix the value of the third edge according to (4)

If the algorithm terminates, it returns a valid loop field t_k. However, the algorithm exposes termination issues in 3D at step five in the while loop: If the list of faces L_F is not empty and there is no face which has only one unfixed edge left, the algorithm is stuck in an infinite loop. The frequency of occurence in which the algorithm fails can be reduced by applying BFS to construct the spanning tree, e.g., if the spanning tree in Fig. 3 is build by DFS instead of BFS. However, also if BFS is applied, there are simple counter examples for which the algorithm does not terminate [13]. To overcome this issue, the algorithm has to be extended to handle this case.

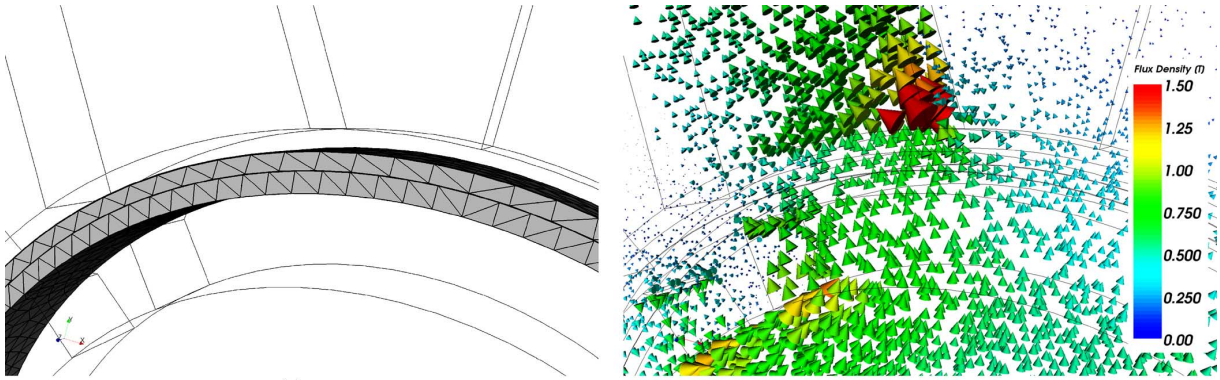
Let us assume there is no face with only one unfixed edge left in the set of faces. Then take one arbitrary face f_i containing the three edges e_{i_1}, e_{i_2} and e_{i_3} with only one fixed edge e_{i_1}. By setting all values corresponding to edges of the spanning tree to zero it is ensured that there exist a unique solution for the loop field t_k. Thus, we can replace one of the two unfixed edges by a linearcombination of the other one to satisfy (4):

$$t_{e_{i2}} = -(t_{e_{i3}} + t_{e_{i1}}) \text{ with } t_{e_{i1}} \in \mathbb{Z} \tag{5}$$

The edge value $t_{e_{i3}}$ is declared free and all edges which are fixed consecutively are described as a linearcombination of free edges. This results in a linear system of equations which contains all free edges as variables and has to be solved. A similar approach is presented in [11] where the authors exploit object-oriented features to implement a symbolic representation of the reals. This approach leads to higher computational cost for the arithmetic operations corresponding with higher execution times. In [11] it is proposed to apply the standard algorithm and only switch to the symbolic representation of the reals and restart the algorithm if the standard one fails. This switch is not necessary using the approach presented above.

VI. Application

The combination of the scalar potential formulation with the proposed construction of the loop fields and the non-conforming sliding interfaces allows for an efficient 3D FE analysis of electrical machines. To handle the relative motion between stator and rotor, the mesh of the complete domain Ω contains two connected components, the stator mesh Ω_s and the rotor mesh Ω_r enclosing a non-conforming interface in the airgap. The support



(a) (b)

Fig. 5. 3D geometry of the example PMSM.

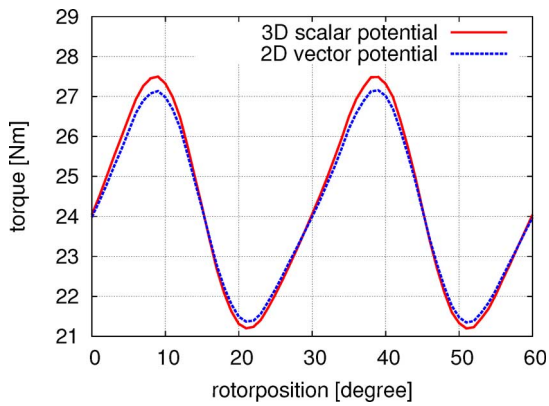


Fig. 6. Comparison of the resulting torques in 2D and 3D.

of the loop field for each conductor or eddy current region has to be limited to one of these components. This is possible because the support can be limited to a topological trivial region containing conductors with its holes only. The non-conforming interface is contained completely in the airgap of the machine, which is obviously a non-conducting region. For all loop fields t_k the edge values t_{e_i} adjacent to the non-conforming interface between Ω_s and Ω_r are set to zero.

The presented algorithms have been implemented in the institute's in-house FE-package iMOOSE [www.iem.rwth-aachen.de] and have been applied to an exemplary permanent magnet excited synchronous machine (PMSM) shown in Fig. 5. The 3D geometry has been generated by extrusion of a 2D geometry in order to use the 2D model as reference simulation. Fig. 4(a) shows the non-conforming discretization in the airgap between the meshes Ω_s and Ω_r. The magnetic flux density for an arbitrary position is depicted in Fig. 4(b). The resulting torque of the machine is evaluated by the eggshell method [14] with respect to the rotorposition. The comparison of torque vs. angle of the 2D and 3D model is shown in Fig. 6 and verifies the chosen approach.

VII. Conclusion

The presented approach allows the 3D simulation of electrical machines including motion by a magnetic scalar potential formulation without the tedious process of defining cuts. The motion is based on a non-conforming Lagrange multiplier method with biorthogonal scalar shape functions allowing for an arbitrary angle of the rotorposition. The computation of the source fields for the $t-\omega$ formulation is based on homology theory to avoid the explicit definition of cuts. All required algorithms to compute the loop fields are presented and applied to the example of a 3D electrical machine field problem. The algorithms have been implemented in the FE-package iMOOSE.

The future research will focus on extending the presented algorithms to take into account the eddy current effects within electrical machines.

References

[1] T. Preston, A. Reece, and P. Sangha, "Induction motor analysis by time-stepping techniques," *IEEE Trans. Magn.*, vol. MAG-24, no. 1, pp. 471–474, 1988.

[2] E. Lange, F. Henrotte, and K. Hameyer, "Biorthogonal shape functions for nonconforming sliding interfaces in 3-D electrical machine Fe models with motion," *IEEE Trans. Magn.*, vol. 48, no. 2, pp. 855–858, 2012.

[3] B. Davat, Z. Ren, and M. Lajoie-Mazenc, "The movement in field modeling," *IEEE Trans. Magn.*, vol. MAG-21, no. 6, pp. 2296–2298, 1985.

[4] B. Wohlmuth, "A comparison of dual Lagrange multiplier spaces for mortar finite element discretizations," *ESAIM: Mathematical Modelling and Numerical Analysis*, vol. 36, no. 6, pp. 995–1012, 2002.

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

[6] E. Lange, F. Henrotte, and K. Hameyer, "A variational formulation for nonconforming sliding interfaces in finite element analysis of electric machines," *IEEE Trans. Magn.*, vol. 46, no. 8, pp. 2755–2758, 2010.

[7] F. Henrotte and K. Hameyer, "An algorithm to construct the discrete cohomology basis functions required for magnetic scalar potential formulations without cuts," *IEEE Trans. Magn.*, vol. 39, no. 3, pp. 1167–1170, May 2003.

[8] J. Simkin and C. W. Trowbridge, "On the use of the total scalar potential on the numerical solution of fields problems in electromagnetics," *Int. J. Numer. Meth. Eng.*, vol. 14, no. 3, pp. 423–440, 1979.

[9] D. L. P. Zhou, Z. Badics, and Z. Cendes, "Nonlinear t-ω formulation including motion for multiply connected 3-D problems," *IEEE Trans. Magn.*, vol. 44, no. 6, pp. 718–721, Jun. 2008.

[10] J. Webb and B. Forghani, "A single scalar potential method for 3D magnetostatics using edge elements," *IEEE Trans. Magn.*, vol. 25, no. 5, pp. 4126–4128, Sep. 1989.

[11] P. Dłotko and R. Specogna, "Efficient generalized source field computation for h-oriented magnetostatic formulations," *Eur. Phys. J.—Appl. Phys.*, vol. 53, no. 02, 2011.

[12] G. Strang, *Linear Algebra and Its Applications*, 3rd ed. Pacific Grove, CA, USA: Brooks Cole, 2003.

[13] P. Dłotko and R. Specogna, "Critical analysis of the spanning tree techniques," *SIAM J. Numer. Anal.*, vol. 48, no. 4, pp. 1601–1624, Sep. 2010.

[14] F. Henrotte, G. Deliége, and K. Hameyer, "The eggshell approach for the computation of electromagnetic forces in 2D and 3D," *COMPEL*, vol. 23, no. 4, pp. 996–1005, 2004.