An Algorithm to Construct the Discrete Cohomology Basis Functions Required for Magnetic Scalar Potential Formulations Without Cuts

François Henrotte and Kay Hameyer

***Abstract*—Magnetic scalar potential formulations without cuts require the definition of a set of basis functions for the cohomology structure of the magnetic field function space. This paper presents an algorithm to construct such a basis in the general case thanks to a properly chosen spanning tree. The algorithm is based on the topological properties of the discrete Whitney complex. It applies to static and dynamic problems.**

***Index Terms*—Algorithms, circuits, differential geometry, duality, electromagnetic coupling, software design/development, spanning tree, topology.**

I. Introduction

In order to reduce computational costs, many finite-element formulations seek to take advantage of the curl-free nature of the field in some region. In particular, magnetic potential formulations are very appealing for three-dimensional (3-D) eddy current problems. In the classical $t-\omega$ formulation, cuts have to be defined in order to make the potential single-valued in the nonconducting regions. However, the definition of cuts can be avoided if special fields are introduced, which form a basis for the cohomology structure of the magnetic field function space, for the geometry under consideration. This paper demonstrates how to do so.

II. Topological Structure

Let Ω be a connected (two-dimensional (2-D) or 3-D) mesh and Γ_b and Γ_h the complementary parts of the boundary $\partial\Omega$ of Ω, where the fields $b_0\cdot n$ and $h_0\wedge n$, respectively, are known. Let $C\subset\Omega$ be the domain occupied by all the conductors of the problem, and ∂C be the boundary of C. Let $W^p(\Omega)$, $p=0$, 1, 2, 3 be the set of differential forms of degree p defined on the domain Ω.

In the classical $t-\omega$ formulation, the field t is defined in the conducting region C only and the potential ω is defined everywhere; t is arbitrarily gauged and cuts have to be adequately defined in $\Omega - C$ in order to make the potential single-valued. The aim of this paper is to introduce a $t-\omega$ formulation that does not require the cumbersome definition of cuts.

Manuscript received June 18, 2002; revised December 20, 2002.

The authors are with ESAT, ELECTA, Katholieke Universiteit Leuven, B-3001 Leuven, Belgium (e-mail: Francois.Henrotte@esat.kuleuven.ac.be).

Digital Object Identifier 10.1109/TMAG.2003.810404

In the nonconducting region, one has to solve div $b=0$, curl $h=0$, and $b=\mu_0 h$ with b and h belonging, respectively, to the sets $B(\Omega-C)$ and $H(\Omega-C)$ defined by

$$\begin{aligned} B(\Omega-C) &= \left\{b\in W^2(\Omega-C),\ (b-b^0)\cdot n = 0 \text{ on } \Gamma_b\right\}\\ H(\Omega-C) &= \left\{h\in W^1(\Omega-C),\ \text{curl}\, h\cdot n = 0 \text{ on } \partial C,\right.\\ &\qquad \left.(h-h^0)\wedge n = 0 \text{ on } \Gamma_h\right\}. \end{aligned} \tag{1}$$

Whereas div $b=0$ can be satisfied without any restriction by defining the vector potential a and $b=$ curl a, the situation is more complicated for curl $h=0$. Inside a small region like a ball or a cube, indeed, a curl-free field is always the gradient of a scalar potential, i.e., $h=\text{grad}\,\omega$. This is the Poincaré lemma. At a larger scale however, this might cease to be the general representation of h. In order to characterize what happens when passing from the local to the global level, one has to focus on the *homology structure* of the functional space $W^1(\Omega-C)$ (Fig. 1).

Let B^1 be the set of the gradients defined on $\Omega-C$ and Z^1 be the set of the curl-free fields defined on $\Omega-C$. Both are vector spaces. Since curl grad $f=0\,\forall f$, one has $B^1\subset Z^1$ but $B^1\neq Z^1$ as there might exist fields that are in Z^1 but not in B^1, i.e., curl-free on $\Omega-C$ that are not gradient fields. Those fields are associated with loops in $\Omega-C$, which link a nonzero current flowing in C. Algebraic topology tells that the vector space containing such fields is defined by the quotient $H^1=Z^1/B^1$. This quotient space is also a vector space, but it is of finite dimension whereas B^1 and Z^1 can be of infinite dimension. If the conductors of the problem under consideration form an electrical circuit with N_Σ linearly independent loops, and if the sections of the conductors in which the currents $\{I_k, k=0,\ldots,N_\Sigma\}$ can be imposed independently are called $\{\Sigma_k, k=0,\ldots,N_\Sigma\}$, it can be shown that the finite dimension of H^1 is precisely the number N_Σ of independent loops formed by the conductors. This is the De Rham theorem.

The magnetic field in the nonconducting region can then be represented in the most general manner by

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

where the so-called *loop fields* $t^k\in H^1$ are computed by imposing a unity current in section Σ_k and a zero current in all other sections.

In order to fulfill the boundary conditions, the loop fields t^k and the scalar potential field ω must belong, respectively, to the sets $H(\Omega-C)$ and $W(\Omega-C)$ defined by (1) and

$$W(\Omega-C)=\left\{\omega\in W^0(\Omega-C),\ \text{grad}\,\omega\wedge n = 0 \text{ on } \Gamma_h\right\}. \tag{3}$$

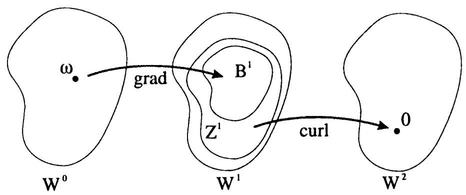

Fig. 1. Topological structure of W^1.

There is no cut involved here since the t^k fields are able to represent the nongradient part of the magnetic field in air. The potential field ω is then a simple continuous scalar field. The problem of an overall $t-\omega$ formulation without cuts has now reduced the computation of the t^k's, simply noted t in the following. The difficulty arises from the fact that the condition curl $t \cdot n = 0$ on ∂C in (1) does not imply that $t \wedge n = 0$ on ∂C if $\Omega - C$ is *not* loop-free, which is generally the case. So, a true boundary condition on t itself is lacking. Moreover, the loop field is not unique; it is determined up to the gradient of any scalar field.

III. Spanning Tree: Unconstrained Case

There exists a large mathematical literature about algebraic topology, which is however quite far away from the preoccupations of the engineers, and somewhat unaccessible to them, even though several attempts has been made to make the essential theoretical background available to a larger audience, e.g., [1]–[4].

Like [5] and [6], we wish however, in this paper, to choose an easier and more pragmatic approach by considering the problem directly at the discrete level, in a mesh of Whitney edge elements. As already mentioned, the difficulties appear when passing from the local to the global level. If one notices that all finite elements are indeed loop-free,[1] the point is then to make the algorithm we are about to describe able, along the way, to retrieve the topological characteristics of the computed regions. We will show that the needed information is simply a few lists of edges.

As in [7], the method is based on the construction of a spanning tree. Let Ω be a connected p-mesh (that means a p-dimensional mesh, $p = 1$, 2 or 3). A p-tree on a p-mesh is a set of edges of the p-mesh that provides a unique path from one node to any other node of the p-mesh. There is no closed loop in a tree.

A. Algorithm to Build a Spanning Tree

Let D be a q-mesh on which a spanning tree is to be built; D can be a part of a bigger p-mesh, $p \geq q$. Firstly, the list $L_N(D)$ of the N_D nodes of D and, for each node $n \in L_N(D)$, the list of the connected edges $L_E(n, D)$ have to be made out. The algorithm, where L_{temp} and L_{tree} are auxiliary sets of edges, is then as follows:

1) $L_{\text{temp}} = \emptyset$, $L_{\text{tree}} = \emptyset$;
2) Pick any initial node n_i in $L_N(D)$ and remove it from that list;
3) Put all the edges of $L_E(n_i, D)$ in L_{temp};
4) If the list L_{temp} is not empty, pick one edge e_j in L_{temp} and remove it from that list:

— If the end node n_j of e_j is not in $L_N(D)$:
  - add the edge e_j in L_{tree};
  - remove n_j from $L_N(D)$;
  - put all the edges of $L_E(n_j, D)$ in L_{temp};
  - repeat step 4.

The algorithm stops when L_{temp} is empty. The list L_{tree} contains then $N_D - 1$ edges of D that form a spanning tree.

There are two degrees of freedom in this algorithm: the choice of the initial node at step 2 and the choice of the edge in L_{temp} at step 4. The choice of the initial node does not influence the characteristics of the spanning tree. If the domain is multiply connected, the algorithm is simply run in each connected part with each time a new initial node. On the other hand, the choice of the edge at step 4 is of importance. If the edge picked out in L_{temp} is the last one that has been inserted, the algorithm works like a first in last out (FILO) stack (Fig. 2) and is comparable to a recursive algorithm. If it is the most formerly inserted edge that is picked out, the algorithm works rather like a first in first out stack (FIFO). With recursive algorithms, one branch of the tree grows as long as possible which results in spanning trees with long branches of unequal length. With FIFO stack algorithms, the branches grow together with the same speed. This results in better balanced trees.

B. Algorithm to Compute the Loop Fields

In the unconstrained case, the loop field in the nonconducting region verifies curl $t = 0$. When edge-based Whitney elements are used, the connectors that describe the interpolated field t are the circulations of t along the edges e_i of the mesh, say t_{e_i}. The differential property curl $t = 0$ can then be expressed by the algebraic relation

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

where $L_F(D)$ is the list of the facets of D and ∂ is the boundary operator. The algorithm to compute the loop field is as follows.

1) Put all the facets of D in the list $L_F(D)$.
2) Fix to zero all the circulations along the edges $e_i \in L_{\text{tree}}$ (gauge).
3) Choose any facet f_i in $L_F(D)$.
4) If all the edges of the facet f_i are fixed but one, fix the last one in order to verify (4) and remove f_i from $L_F(D)$.
5) If the list $L_F(D)$ is not empty, go to step 3).

The list $L_F(D)$ must usually be run through several times before making it empty. The number of iterations is generally lower if the FIFO stack algorithm is used.

IV. Minimization of the Support of the Loop Field

Generally, the loop field will not extend as far as the external boundaries of the problem. It can be more or less confined in the holes between the loops that form the conductors. Moreover, the spanning tree is not unique, even in the presence of constraints.

[1]A *simply connected* region is a region where any closed path can be contracted to a point without leaving the region. This is a notion from *homotopy* analysis. A *loop-free region* [2] is a region where any closed path links zero-current. This is a notion from *homology* analysis, which is, by the way, less restrictive than the former [3]. A loop-free region is then by definition a region where scalar potentials are single-valued. It is therefore the notion we need.

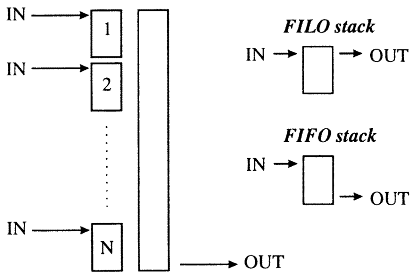

Fig. 2. Multiple stack.

This freedom can be turned to good account to reduce the support of the loop field to a minimum. If a loop-free region that contains C entirely is defined, the loop field can be set to zero on the surface S of that region and need not be computed outside it. This minimization may lead to a substantial saving in the number of degrees of freedom needed to represent the loop fields.

V. Spanning Tree: Constrained Case

In summary, the loop field must obey the following constraints:

$$\begin{cases} \text{curl}\, t = 0 \text{ in } \Omega - C \\ (t - h^0) \wedge n = 0 \text{ on } \Gamma_h \\ t \wedge n = 0 \text{ on } S \\ \text{curl}\, t \cdot n = 0 \text{ on } \partial C \\ \int_{\partial \Sigma_k} t = I_k,\ k = 1, \ldots, N_\Sigma \end{cases} . \tag{5}$$

The first one is a volume constraint; the second, third, and fourth are surface constraints and the fifth one is a set of global curve constraints, which bring into the nonconducting region the information needed about the currents flowing in C. In order to ensure that all those constraints are properly considered, the spanning tree must obey some related constraints which can be deduced from the rule:

> *"If one wishes to build a loop field on a p-mesh D and if the loop field must obey some constraint on a q-submesh $D' \subset D$, $q \le p$, the spanning tree built on D must include a spanning tree on D'."*

The algorithm that builds the spanning tree must therefore be able to ensure that the restrictions of the spanning tree to the surfaces Γ_h, S, and ∂C and to the curves $\partial \Sigma_k$ are also spanning trees themselves for those surfaces and curves. However, the union of two spanning trees defined independently on two meshes is not, in general, a spanning tree on the union of the two meshes. A global approach is needed which can be worked out thanks to the rule:

> *"The spanning tree built on the union of two meshes $D \cup D'$ includes a spanning tree on D and a spanning tree on D' if it includes a spanning tree on $D \cap D'$."*

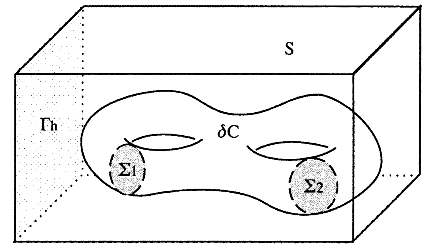

Fig. 3. Constrained surfaces.

A. Algorithm to Build the Constrained Spanning Tree

One now has the elements to establish the algorithm for building the constrained spanning tree in the general case. The user has to define, for the problem under consideration (Fig. 3), the volumes $\{V_i, i = 1, \ldots, N_V\}$ in which the spanning tree must be built, and the constrained surfaces, which are

$$\{S_j, j = 1, \ldots, N_S\} = \{\Gamma_h, S, \partial C, \Sigma_k, k = 1, \ldots, N_\Sigma.\} \tag{6}$$

Among all the edges connected to any node, the edges belonging to the intersection of some of those sets must be put into the tree before any edge of the intersecting sets. For that purpose, the algorithm works by organizing the edges of the mesh into a hierarchy. A level of priority called *level* is attributed beforehand to each edge by the following algorithm where L_{V_j} stands for the set of the edges that belong to V_j, L_{S_j} for the set of the edges that belong to S_j and where $L_\cup$, and $L_\cap$ and L_{temp} are auxiliary lists of edges.

1) $level := 0$, $L_\cup := \emptyset$, $L_\cap := \emptyset$.
2) For each volume V_j:
   - Attribute the level of priority *level* to the edges of L_{V_j},
   - $L_{\text{temp}} := L_\cup \cap L_{V_j}$,
   - $L_\cup := L_\cup \cup L_{V_j}$,
   - $L_\cap := L_\cap \cap L_{\text{temp}}$,
   - $level++$.
3) Attribute the level of priority *level* to the edges of $L_\cap$.
4) $level++$, $L_\cup := \emptyset$, $L_\cap := \emptyset$.
5) For each surface S_j:
   - Attribute the level of priority *level* to the edges of L_{S_j};
   - $L_{\text{temp}} := L_\cup \cap L_{S_j}$;
   - $L_\cup := L_\cup \cup L_{S_j}$;
   - $L_\cap := L_\cap \cap L_{\text{temp}}$;
   - $level++$.
6) Attribute the level of priority *level* to the edges of $L_\cap$.

When this prior operation is achieved, the algorithm to build the spanning tree can be run. It is mainly unchanged in comparison with the unconstrained case. The only difference is that the edge that is picked out in the list L_{temp} must be one with the highest level of priority. For that purpose, the stack structure of L_{temp} must be adapted as shown in Fig. 2. The simple FIFO stack is replaced by a multiple stack. Each individual stack is associated with one particular level of priority and they are connected end to end with increasing priority. The multiple stack works differently for pushing edges into the stack and for pulling

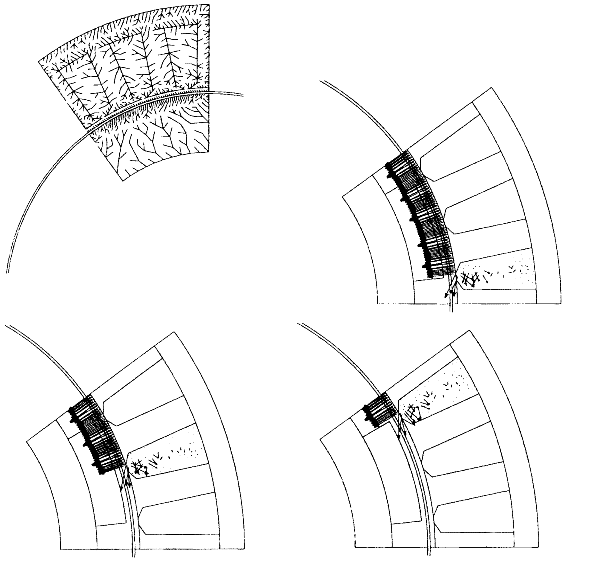

Fig. 4. Spanning tree and loop fields in the stator of a permanent magnet motor.

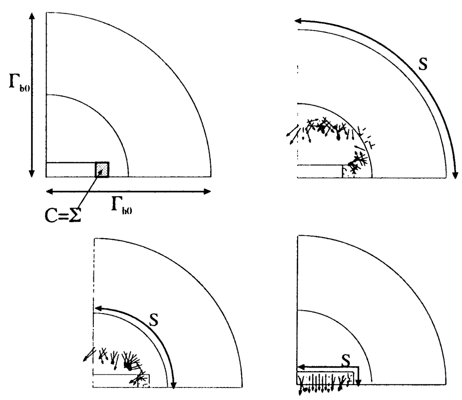

Fig. 5. Constrained surfaces.

them out. The edges are pushed into the individual stacks in accordance with their own level of priority but they are pulled out as if the whole stack were one single stack. This particular stack structure ensures that the edge pulled out of the stack at any time is one with the highest level of priority and therefore that all the intersections are considered before the intersecting sets.

VI. Examples

The first example shows the loop fields associated with the three stator phase currents in a permanent magnet motor, assuming a uniform current density in the slots. Because of the symmetries, only one tenth of the machine has to be represented in the model and the moving band technique is used.

The second example is a simple axisymmetrical coil. In this case, the surfaces C and Σ are identical. Fig. 5 shows the different surfaces $\{S_j\}$ that have been indicated to the algorithm. Several choices of the surface S are considered in order to show the minimization of the support of the loop field. Note that the surface S appears as an open surface but it is actually a closed surface if the symmetries are unfolded.

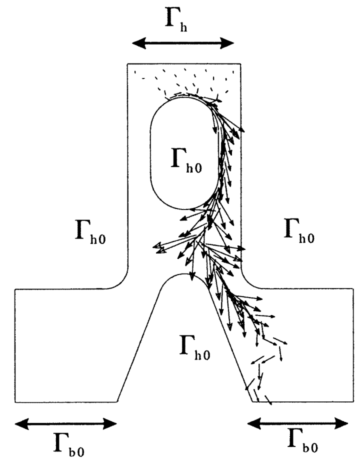

Fig. 6. Loop field in an elasticity problem.

The third example (Fig. 6) comes from a formulation of elasticity with Whitney forms. In 2-D, the stress can be represented by a scalar potential. There is no volume load in this case but a surface load, i.e., a nonhomogeneous Dirichlet boundary condition, on the upper part of the structure Γ_h. The loop field propagates the surface load throughout the structure, which is not loop-free due to the presence of the hole.

VII. Conclusion

The magnetic scalar potential formulations require the definition of loop fields in the static case as well as in the dynamic case. This paper has presented a general algorithm to compute automatically such loop fields thanks to a constrained spanning tree. Two fundamental rules governing the construction of the loop fields have been given as well as an algorithm that follows those rules. The algorithm works without any restriction, provided the geometry is known. The user just has to indicate the surfaces on which the loop field is subjected to some constraints. The algorithm does not require the solution of any system of equations. Although it may seem complicated, it considers the problem in the most general case and it can be implemented once and for all.

References

[1] L. Kettunen, K. Forsman, and A. Bossavit, "Formulation of the eddy current problem in multiply connected regions in term of h," *Int. J. Numer. Methods Eng.*, vol. 41, pp. 935–954, 1998.

[2] ——, "Discrete spaces for div and curl-free fields," *IEEE Trans. Magn.*, vol. 34, pp. 2551–2554, Sept. 1998.

[3] P. W. Gross and P. R. Kotiuga, "A challenge for magnetic scalar potential formulations of 3-D eddy current problems: multiply connected cuts in multiply connected regions which necessarily leave the cut complement multiply connected," in *Electric and Magnetic Fields*, A. Nicolet and R. Belmans, Eds. New York: Plenum Press, 1995.

[4] L. Kettunen, "Fields and circuit in computational electromagnetism," *IEEE Trans. Magn.*, vol. 37, pp. 3393–3396, Sept. 2001.

[5] Z. Ren, "$T - \Omega$ formulation for eddy-current problems in multiply connected regions," *IEEE Trans. Magn.*, vol. 38, pp. 557–560, Mar. 2002.

[6] S. Suurinemi, T. Tarhasaari, and L. Kettunen, "Generalization of the spanning-tree technique," *IEEE Trans. Magn.*, vol. 38, pp. 525–528, Mar. 2002.

[7] J. P. Webb and B. Forghani, "A single scalar potential formulation for 3D magnetostatics using edge elements," *IEEE Trans. Magn.*, vol. 25, pp. 4126–4128, Sept. 1989.