Finite element approximation of the curl−div system

Alberto Valli

Department of Mathematics, University of Trento, Italy

In collaboration with:

Ana Alonso Rodríguez, University of Trento, Italy

Enrico Bertolazzi, University of Trento, Italy

Riccardo Ghiloni, University of Trento, Italy.

Outline

1. Introduction
   - Harmonic fields and their approximation
   - curl–div system
2. Finite element approximation
3. The fundamental discrete problem
4. Numerical results

Motivation

Aim of this talk is two-fold:

- construct a finite element approximation of the space of harmonic fields

$$\mathbb{H}_\mu(\Omega)=\{\mathbf{v} \in (L^2(\Omega))^3 \,|\, \mathbf{curl}\,\mathbf{v} = \mathbf{0}, \mathrm{div}(\mu\mathbf{v}) = 0, \mu\mathbf{v}\cdot\mathbf{n} = 0 \text{ on } \partial\Omega\}$$

- furnish a finite element numerical solution to the **curl**-div system

$$\begin{aligned} \mathbf{curl}\,\mathbf{H} = \mathbf{J} & \quad \text{in } \Omega \\ \mathrm{div}(\mu\mathbf{H}) = f & \quad \text{in } \Omega \\ \mu\mathbf{H}\cdot\mathbf{n} = q & \quad \text{on } \partial\Omega\,. \end{aligned}$$

[Here: $\Omega \subset \mathbb{R}^3$ a bounded domain with a Lipschitz boundary $\partial\Omega$ and unit outward normal vector $\mathbf{n}$; μ a symmetric matrix, uniformly positive definite in Ω and with entries in $L^\infty(\Omega)$.]

Motivation (cont'd)

In particular:

- we give an efficient computational way for constructing the so-called *loop fields*, i.e., the irrotational vector fields $\mathbf{T}_0$ that cannot be expressed in Ω as the gradient of any single-valued scalar potential (there exists a loop in Ω such that the line integral of $\mathbf{T}_0$ on it is different from 0)
- we give an efficient computational way for constructing a so-called *source field*, i.e., a vector field $\mathbf{H}_e$ satisfying $\mathbf{curl}\, \mathbf{H}_e = \mathbf{J}$ in Ω.

Motivation (cont'd)

- A suitable set of loop fields furnishes a basis of the first de Rham cohomology group of Ω (the quotient space between curl-free vector fields and gradients defined in Ω).

[Here we need a definition: if the only linear combination of a set of loop fields that equals a gradient is the trivial one, we say that those loop fields are cohomologically independent. Then, "suitable set of loop fields" means "a maximal set of cohomologically independent loop fields".]

- Source fields are often needed for formulating electromagnetic problems (for instance, eddy current problems in terms of a magnetic scalar potential in the insulating region). In fluid dynamics, reconstructing the velocity field from the vorticity is a source field problem.

More on the space of harmonic fields $\mathbb{H}_\mu(\Omega)$

Let us start from the approximation of $\mathbb{H}_\mu(\Omega)$. The dimension of this vector space is g, the first Betti number of Ω.

[The first Betti number is the rank of the first homology group of $\overline{\Omega}$, i.e., the number of the elements of a maximal set of homologically independent non-bounding cycles in $\overline{\Omega}$; it is also the dimension of the first de Rham cohomology group of Ω.]

[Another definition: if the only linear combination of a set of cycles that coincides with the boundary of a surface is the trivial one, we say that those cycles are homologically independent.]

Loop fields and harmonic fields

A theoretical way for determining a basis of $\mathbb{H}_\mu(\Omega)$ is grounded on the fact that there exist g surfaces Σ_n, each one "cutting" a non-bounding cycle in $\overline{\Omega}$.

The method reads as follows: denoting by $[\cdot]_{\Sigma_n}$ the jump across Σ_n, take a function φ_n^* that satisfies $[\varphi_n^*]_{\Sigma_n} = 1$ and define $\mathbf{T}_{0,n}^*$ the extension to Ω of $\mathbf{grad}\, \varphi_n^*$, computed in $\Omega \setminus \Sigma_n$.

It is clear that $\mathbf{T}_{0,n}^*$ is curl-free and has line integral equal to 1 on the non-bounding cycle cut by the surface Σ_n; therefore, it is a loop field [but it is not divergence free, nor tangential to the boundary].

A basis of $\mathbb{H}_\mu(\Omega)$ is given by a correction of these fields, i.e, by $\boldsymbol{\rho}_n = \mathbf{T}_{0,n}^* + \mathbf{grad}\, \psi_n$, where ψ_n solves the Neumann problem

$$\begin{array}{ll} \mathrm{div}(\mu \mathbf{grad}\, \psi_n) = -\mathrm{div}(\mu \mathbf{T}_{0,n}^*) & \text{in } \Omega \\ \mu \mathbf{grad}\, \psi_n \cdot \mathbf{n} = -\mu \mathbf{T}_{0,n}^* \cdot \mathbf{n} & \text{on } \partial\Omega\,. \end{array}$$

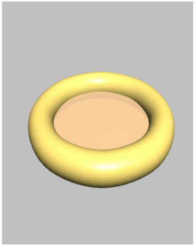
"Cutting" surfaces

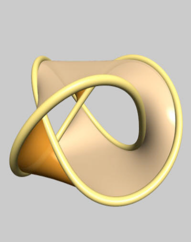
There is an extensive literature concerning their construction (see Kotiuga [1987,1988,1989], Harrold and Simkin [1985], Leonard et al. [1993], Ren [2002], Simkin et al. [2004], Dular [2005]). However, in general topological situations (for instance, in the case of domains that are the complement of "knotted" domains) and for real-sized finite element meshes this construction is not feasible, as it can be quite expensive from the computational point of view (see Bossavit [1998], Dłotko et al. [2009]).

To give an idea of the shape of a "cutting" surface, we recall that, when Ω is the complement in a box of a knot, it is the Seifert surface of the knot. [Instead, in the case of a link of two or more knots, the Seifert surface is not enough to finish the construction, as the "cutting" surfaces must be as many as the knots.]

Seifert surfaces

[Images produced with SeifertView, Jarke J. van Wijk, Technische Universiteit Eindhoven.]





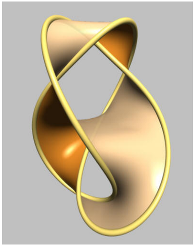


Figure : Torus, trefoil knot, knot 4_1.

Seifert surfaces (cont'd)

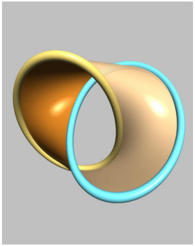
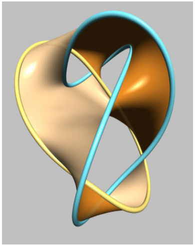
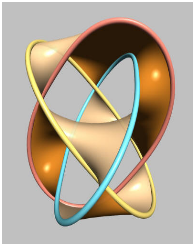
  

Figure : Hopf link, Whitehead link, Borromean rings.

An alternative procedure for constructing the loop fields

We have seen that "cutting" surfaces are not always available. Therefore, it is interesting to propose an alternative procedure for the determination of a basis of discrete loop fields.

Tools:

- homology theory
  - generators of the first homology group of $\partial\Omega$, $\overline{\Omega}$ and $\mathbb{R}^3 \setminus \Omega$
- graph theory applied to the mesh
  - a spanning tree of the graph given by the edges of the mesh
- direct elimination procedure
  - a direct algorithm of Webb and Forghani [1989]
  - an explicit formula for the discrete loop fields in terms of *linking numbers*.

curl-div system: variational formulation

Focusing on the **curl**-div system, it must be observed that, in order to achieve well-posedness, an orthogonality condition has to be added:

$$\begin{cases} \mathbf{curl}\,\mathbf{H} = \mathbf{J} & \text{in } \Omega \\ \mathrm{div}(\mu\mathbf{H}) = f & \text{in } \Omega \\ \mu\mathbf{H}\cdot\mathbf{n} = q & \text{on } \partial\Omega \\ \int_\Omega \mu\mathbf{H}\cdot\boldsymbol{\eta} = 0 & \forall\,\boldsymbol{\eta}\in\mathbb{H}_\mu(\Omega)\,. \end{cases}$$

However, we can always assume that $f = 0$ and $q = 0$. In fact, having solved the problem with these data, the solution of the full problem is obtained by adding a gradient to the solution of the simplified problem, precisely, the gradient of a solution ω of the Neumann problem

$$\begin{cases} \mathrm{div}(\mu\,\mathbf{grad}\,\omega) = f & \text{in } \Omega \\ \mu\,\mathbf{grad}\,\omega\cdot\mathbf{n} = q & \text{on } \partial\Omega\,. \end{cases}$$

curl-div system: variational formulation (cont'd)

A quite simple variational formulation of the **curl**-div system (with $f = 0$ and $q = 0$) is: given $\mathbf{J} \in (L^2(\Omega))^3$ satisfying the necessary conditions, find $\mathbf{H} \in (L^2(\Omega))^3$ such that

$$\begin{array}{ll} \mathbf{curl}\, \mathbf{H} = \mathbf{J} & \text{in } \Omega \\ \int_\Omega \mu \mathbf{H} \cdot \mathbf{z} = 0 & \forall\, \mathbf{z} \in H^0(\mathbf{curl}\,; \Omega)\,, \end{array} \qquad (1)$$

where $H^0(\mathbf{curl}\,;\Omega) = \{\mathbf{z} \in (L^2(\Omega))^3 \,|\, \mathbf{curl}\, \mathbf{z} = \mathbf{0}\}$, and we have taken into account the μ-orthogonal decomposition

$$H^0(\mathbf{curl}\,;\Omega) = \mathbf{grad}\, H^1(\Omega) \oplus \mathbb{H}_\mu(\Omega)\,.$$

Problem unknowns and pre-calculation

Unkowns are the projection of **H** on **grad** $H^1(\Omega)$ (therefore a scalar "potential") and the projections of **H** on $\mathbb{H}_\mu(\Omega)$: the "cheapest" formulation, as:

- the field **H** is not the unknown
- a vector potential **A** of μ**H** is not introduced
- we have one unknown per node for determining the scalar "potential", plus g parameters (coming from topology).

On the other hand, to reformulate the problem in the vector space $H^0(\mathbf{curl}\,;\Omega)$ we need to know a source field $\mathbf{H}_e$ such that $\mathbf{curl}\,\mathbf{H}_e = \mathbf{J}$.

Discrete source fields

Therefore, the determination of the discrete source fields will be a necessary tool for numerical approximation. This problem has been widely considered, mainly for simple topological domains (see, e.g., Webb and Forghani [1989], Preis et al. [1992], Dular et al. [1997], Le Ménach et al. [1998], Rapetti et al. [2003], Dular [2005], Badics and Cendes [2007], Dłotko and Specogna [2010]).
Our recipe:

- proceed as for the loop fields [generators of the homology group on $\partial\Omega$, spanning tree of the mesh in $\overline{\Omega}$, Webb–Forghani algorithm]
- when the algorithm stops, introduce a dual graph for the remaining edges
- use a direct solver for the final (small and sparse) system.

Finite element spaces

Being given with a triangulation $\mathcal{T}_h$ of $\overline{\Omega}$ composed by tetrahedra, we consider the following spaces of finite elements:

- The space $L_h \subset H^1(\Omega)$ of continuous piecewise linear finite elements. Its dimension is n_v, the number of vertices in $\mathcal{T}_h$.
- The space $N_h \subset H(\mathbf{curl}\,;\Omega)$ of Nédélec edge finite elements of degree 1 [locally: $\mathbf{a} + \mathbf{b} \times \mathbf{x}$]. Its dimension is n_e, the number of edges in $\mathcal{T}_h$.
- The space $RT_h \subset H(\mathrm{div};\Omega)$ of Raviart–Thomas finite elements of degree 1 [locally: $\mathbf{a} + b\,\mathbf{x}$]. Its dimension is n_f, the number of faces in $\mathcal{T}_h$.

We have $\mathbf{grad}\, L_h \subset N_h$ and $\mathbf{curl}\, N_h \subset RT_h$.

Finite element **curl**-div problem

The finite element approximation of (1) reads as follows. Given $\mathbf{J}_h \in RT_h$, a suitable finite element approximation of $\mathbf{J}$ satisfying the necessary conditions, find $\mathbf{H}_h \in N_h$ such that

$$\begin{array}{ll} \mathbf{curl}\, \mathbf{H}_h = \mathbf{J}_h & \text{in } \Omega \\ \int_\Omega \mu \mathbf{H}_h \cdot \mathbf{z}_h = 0 & \forall\, \mathbf{z}_h \in N_h \cap H^0(\mathbf{curl}\,;\Omega)\,. \end{array} \tag{2}$$

If a source field $\mathbf{H}_{e,h} \in N_h$ with $\mathbf{curl}\, \mathbf{H}_{e,h} = \mathbf{J}_h$ is known, we can write:

$$\begin{array}{l} \text{find } \mathbf{K}_h \in N_h \cap H^0(\mathbf{curl}\,;\Omega) : \\ \qquad \int_\Omega \mu \mathbf{K}_h \cdot \mathbf{z}_h = -\int_\Omega \mu \mathbf{H}_{e,h} \cdot \mathbf{z}_h \\ \forall\, \mathbf{z}_h \in N_h \cap H^0(\mathbf{curl}\,;\Omega)\,, \end{array} \tag{3}$$

and define $\mathbf{H}_h = \mathbf{K}_h + \mathbf{H}_{e,h}$.

Finite element **curl**-div problem (cont'd)

We have thus seen that a finite element approximation of (1) is standard provided that:

- we know a discrete source field $\mathbf{H}_{e,h}$ satisfying $\mathbf{curl}\,\mathbf{H}_{e,h} = \mathbf{J}_h$
- we are able to characterize in a simple way the space $N_h \cap H^0(\mathbf{curl}\,;\Omega)$.

With respect to the latter point, we mimic the μ-orthogonal decomposition $H^0(\mathbf{curl}\,;\Omega) = \mathbf{grad}\,H^1(\Omega) \oplus \mathbb{H}_\mu(\Omega)$ and write the elements $\mathbf{z}_h \in N_h \cap H^0(\mathbf{curl}\,;\Omega)$ as [we will be back to this...]

$$\mathbf{z}_h = \mathbf{grad}\,\phi_h + \sum_{n=1}^{g} \xi_n \mathbf{T}_{0,n} \,,$$

where $\mathbf{T}_{0,n}$ are suitable finite element loop fields. [Not harmonic fields! Thus note that this decomposition is not μ-orthogonal.]

Finite element **curl**-div problem (cont'd)

Therefore, problem (3) can be rewritten as:

find $\beta_i \in \mathbb{R}$, $i = 1, \ldots, n_v - 1$, and $\eta_j \in \mathbb{R}$, $j = 1, \ldots, g$:

$$\sum_{i=1}^{n_v-1} \beta_i \int_\Omega \mu \mathbf{grad}\, \Phi_{h,i} \cdot \mathbf{grad}\, \Phi_{h,l} + \sum_{j=1}^{g} \eta_j \int_\Omega \mu \mathbf{T}_{0,j} \cdot \mathbf{grad}\, \Phi_{h,l}$$
$$= -\int_\Omega \mu \mathbf{H}_{e,h} \cdot \mathbf{grad}\, \Phi_{h,l} \quad \forall\, l = 1, \ldots, n_v - 1$$
$$\sum_{i=1}^{n_v-1} \beta_i \int_\Omega \mu \mathbf{grad}\, \Phi_{h,i} \cdot \mathbf{T}_{0,n} + \sum_{j=1}^{g} \eta_j \int_\Omega \mu \mathbf{T}_{0,j} \cdot \mathbf{T}_{0,n}$$
$$= -\int_\Omega \mu \mathbf{H}_{e,h} \cdot \mathbf{T}_{0,n} \quad \forall\, n = 1, \ldots, g\,.$$

(Here $\{\Phi_{h,1}, \ldots, \Phi_{h,n_v}\}$ is a basis of L_h, and n_v is the number of the vertices of the mesh $\mathcal{T}_h$.)

Finite element **curl**-div problem (cont'd)

The solution of problem (2) is then determined by setting

$$\mathbf{H}_h = \sum_{i=1}^{n_v-1} \beta_i \, \mathbf{grad}\, \Phi_{h,i} + \sum_{j=1}^{g} \eta_j \mathbf{T}_{0,j} + \mathbf{H}_{e,h} \, . \qquad (4)$$

Summing up:

- a scalar unknown $\psi_h = \sum_{i=1}^{n_v-1} \beta_i \Phi_{h,i}$
- g "topological" unknowns η_j.

The fundamental discrete problem

Let us consider:

- a basis σ_n of the first homology group of $\overline{\Omega}$
- a basis $\widehat{\sigma}_n$ of the first homology group of $\mathbb{R}^3 \setminus \Omega$
- a spanning tree $\mathcal{S}_h$ of the graph given by the edges of $\mathcal{T}_h$.

We focus now on our *main problem*: given $\mathbf{J}_h \in RT_h$ satisfying the necessary conditions, find $\mathbf{Z}_h \in N_h$ such that

$$\begin{array}{ll} \mathbf{curl}\, \mathbf{Z}_h = \mathbf{J}_h & \text{in } \Omega \\ \oint_{\sigma_n} \mathbf{Z}_h \cdot d\mathbf{s} = \kappa_n & \forall\, n = 1, \ldots, g \\ \int_{e'} \mathbf{Z}_h \cdot \boldsymbol{\tau} = 0 & \forall\, e' \in \mathcal{S}_h \,, \end{array} \tag{5}$$

where $\kappa_1, \ldots, \kappa_g$ are real numbers.
[Note that the number of edges e' in $\mathcal{S}_h$ is $n_v - 1$; therefore $(5)_3$ can be seen as a "filtre" for gradients.]

Back to source fields, loop fields and finite element basis

Clearly,

- a discrete source field $\mathbf{H}_{e,h}$ can be computed by solving (5), for any choice of κ_n.

But also:

- a set of cohomologically independent finite element loop fields $\mathbf{T}_{0,j}$ can be determined by solving (5) with $\mathbf{J}_h = \mathbf{0}$ and $\kappa_n = m_{n,j}$, for any choice of a non-singular matrix $M = (m_{n,j})$
- a basis of $N_h \cap H^0(\mathbf{curl}\,;\Omega)$ can be computed starting from $\{\Phi_{h,1}, \ldots, \Phi_{h,n_v}\}$, a basis of L_h, and using these loop fields in this simple way:

$$\{\mathbf{grad}\,\Phi_{h,1}, \ldots, \mathbf{grad}\,\Phi_{h,n_v-1}\} \cup \{\mathbf{T}_{0,1}, \ldots, \mathbf{T}_{0,g}\}\,.$$

An algorithm for solving (5)

Since we are looking for a Nédélec edge element, the number of unknowns in (5) is given by the number n_e of the edges of the mesh $\mathcal{T}_h$.

Since we are imposing the matching between two Raviart–Thomas elements, the number of equations of $\mathbf{curl}\,\mathbf{Z}_h = \mathbf{J}_h$ is given by the number n_f of the faces of the mesh $\mathcal{T}_h$ (and its null-space has dimension $g + n_v - 1$).

Therefore, (5) is rectangular system with more equations $(n_f + g + n_v - 1)$ than unknowns (n_e). However, it has a full rank and has a unique solution.

- Can we find an efficient solver?

A variation on the theme

What about solving

$$\mathbf{grad}\,\varphi = \mathbf{q} \quad \text{in } \Omega\,?$$

We can take an edge element approximation $\mathbf{q}_h$ of $\mathbf{q}$, and look for a nodal element $\varphi_h \in L_h$ such that $\mathbf{grad}\,\varphi_h = \mathbf{q}_h$ in Ω. This means that we have to match two Nédélec edge elements, hence the line integrals of $\mathbf{grad}\,\varphi_h$ and $\mathbf{q}_h$ on each edge of the mesh have to be the same.

Starting from the root v_* of a spanning tree $\mathcal{S}_h$, where we impose $\varphi_h(v_*) = 0$, we have only to compute

$$\varphi_h(v^*) = \varphi_h(v_*) + \int_{e'} \mathbf{q}_h \cdot \boldsymbol{\tau}$$

for an edge $e' = [v_*, v^*] \in \mathcal{S}_h$, and, since $\mathcal{S}_h$ is a spanning tree, going on in this way we can visit all the vertices of $\mathcal{T}_h$.

A variation on the theme (cont'd)

In other words, the matrix associated to the linear system to solve has exactly two non-zero values for each row. The spanning tree is a tool for selecting the rows for which, using the additional equation $\varphi_h(v_*) = 0$, one can eliminate the other unknowns one after the other.

- Can we do something similar for problem (5)?

An algorithm for solving (5) (cont'd)

For problem $\mathbf{curl}\, \mathbf{Z}_h = \mathbf{J}_h$ we have to match two Raviart–Thomas elements, hence their fluxes across each face of $\mathcal{T}_h$ have to be the same.

Since the Stokes theorem assures that

$$\int_{e_1} \mathbf{Z}_h \cdot \boldsymbol{\tau} + \int_{e_2} \mathbf{Z}_h \cdot \boldsymbol{\tau} + \int_{e_2} \mathbf{Z}_h \cdot \boldsymbol{\tau} = \int_f \mathbf{J}_h \cdot \boldsymbol{\nu} \,, \qquad (6)$$

where $\partial f = e_1 \cup e_2 \cup e_3$ and $\boldsymbol{\nu}$ is the unit normal vector on f (with consistent orientation), we deduce that the corresponding linear system has exactly three non-zero values for each row.

With respect to the preceding case:

- need to work on the edges instead of on the vertices
- three unknowns per row instead of two.

Webb–Forghani algorithm

Webb and Forghani [1989] proposed the following solution algorithm:

1. set value 0 to the unknowns corresponding to an edge belonging to the spanning tree
2. take a face f for which at least one edge unknown has not yet been assigned
    1. if exactly one edge unknown is not determined, compute its value from the Stokes relation (6)
    2. if two or three edge unknowns are not determined, pass to another face
3. if the iterations stop, use $\oint_{\sigma_n} \mathbf{Z}_h \cdot d\mathbf{s} = \kappa_n$ to restart.

Webb–Forghani algorithm (cont'd)

The Webb–Forghani algorithm is a simple elimination procedure for solving the linear system at hand, and it is quite efficient, as the computational costs is linearly dependent on the number of unknowns.

The weak point is that:

- it strongly depends on the choice of the spanning tree and it can stop without having determined all the edge unknowns (even in simple topological situations!)

(see Dłotko and Specogna [2010]).

Webb–Forghani algorithm in action

	n_e	$n_e^{(2)}$ breadth-first	$n_e^{(2)}$ depth-first
Test A	42200	0	27912
Test B	35380	0	23595
Test C	25768	0	15707
Test D	15349	2092	9554
Test E	34372	6002	22776
Test F	80504	12916	53488

Table : Dependence of the reduction of the unknowns on the choice of the spanning tree.

[A: 2-torus; B: Borromean rings; C: two-5-tori link; D: trefoil knot; E: knot 4_1; F: two-4_1-knots link.]

A modified algorithm

We propose a modification: when Webb–Forghani algorithm stops

- construct a dual graph whose nodes are the non-assigned edges and the arcs are the 1-faces (faces where only one degree of freedom has been determined).

Since each 1-face naturally "connects" two non-assigned edges, this is a graph (in general, not connected).
Choosing a spanning tree on each connected component, we have reduced the problem to a small and sparse linear system with as many unknowns as the number of connected components of the dual graph (and as many equations as the number of the faces with no edge assigned, plus g_*, $0 \leq g_* \leq g$).
This problem has a unique solution, hence it can be solved by using an algebraic direct method.

An explicit formula for the loop fields

If $\mathbf{J}_h = \mathbf{0}$ we devise an explicit formula for the solution to (5).

The idea is the following: the Biot–Savart law gives the magnetic field generated by a unitary density current concentrated along the edge cycle $\widehat{\sigma}_j$ (a generator of the first homology group of $\mathbb{R}^3 \setminus \Omega$) by means of the formula:

$$\widehat{\mathbf{H}}(\mathbf{x}) = \frac{1}{4\pi} \oint_{\widehat{\sigma}_j} \frac{\mathbf{y} - \mathbf{x}}{|\mathbf{y} - \mathbf{x}|^3} \times d\mathbf{s}_y \quad , \quad \mathbf{x} \notin \widehat{\sigma}_j \, .$$

Since the cycle $\widehat{\sigma}_j$ can be chosen external to $\overline{\Omega}$, one has $\mathbf{curl}\, \widehat{\mathbf{H}} = \mathbf{0}$ in Ω. Moreover, on each cycle $\gamma \subset \overline{\Omega}$ that is linking the current passing in $\widehat{\sigma}_j$ one finds $\oint_\gamma \widehat{\mathbf{H}} \cdot d\mathbf{s} \neq 0$, hence $\widehat{\mathbf{H}}$ is a loop field.

[There are cycles γ with the required property: for instance, one of the generators of the first homology group of $\overline{\Omega}$.]

An explicit formula for the loop fields (cont'd)

Clearly, the Nédélec interpolant $\Pi^{N_h}\widehat{\mathbf{H}}$ is a finite element loop field. For each $e \in \mathcal{T}_h$, its degrees of freedom are given by

$$\widehat{q}_e = \frac{1}{4\pi} \int_e \left(\oint_{\widehat{\sigma}_j} \frac{\mathbf{y} - \mathbf{x}}{|\mathbf{y} - \mathbf{x}|^3} \times d\mathbf{s}_y \right) \cdot \boldsymbol{\tau}_x \,.$$

This resembles the formula for computing the linking number between $\widehat{\sigma}_j$ and another disjoint cycle σ:

$$LK(\sigma, \widehat{\sigma}_j) = \frac{1}{4\pi} \oint_\sigma \left(\oint_{\widehat{\sigma}_j} \frac{\mathbf{y} - \mathbf{x}}{|\mathbf{y} - \mathbf{x}|^3} \times d\mathbf{s}_y \right) \cdot d\mathbf{s}_x \,.$$

- The linking number is an integer that represents the number of times that each cycle winds around the other.

An explicit formula for the loop fields (cont'd)

- Is it possible to reduce the definition of the finite element loop field to the computation of suitable linking numbers?

Consider the spanning tree $\mathcal{S}_h$, its root v_*, and define in the vertices of $\mathcal{T}_h$ the scalar function $\phi_h \in L_h$ as $\phi_h(v_*) = 0$ and

$$\phi_h(v_b) = \phi_h(v_a) + \widehat{q}_{[v_a,v_b]} \quad \forall e' = [v_a, v_b] \in \mathbf{S}_h \, .$$

The Nédélec finite element $\mathbf{Z}_h = \Pi^{N_h}\widehat{\mathbf{H}} - \mathbf{grad}\, \phi_h$ is a loop field, and its degrees of freedom are equal to 0 for all the edges e' of the spanning tree $\mathcal{S}_h$.

For each $e \in \mathcal{T}_h$, define now by D_e the edge cycle constituted by: the edges from the root of the spanning tree $\mathcal{S}_h$ to the first vertex v_e^- of e, the edge e, the edges from the second vertex v_e^+ of e to the root of the spanning tree $\mathcal{S}_h$. In particular, $D_{e'}$ is a trivial cycle if $e' \in \mathcal{S}_h$.

An explicit formula for the loop fields (cont'd)

When $e \notin \mathcal{S}_h$ the cycle D_e is constituted by edges all belonging to the spanning tree (except e): hence we have

$$\begin{aligned}
&\tfrac{1}{4\pi} \oint_{D_e} \left(\oint_{\widehat{\sigma}_j} \tfrac{\mathbf{y}-\mathbf{x}}{|\mathbf{y}-\mathbf{x}|^3} \times d\mathbf{s}_y \right) \cdot d\mathbf{s}_x \\
&\quad = \widehat{q}_e + \textstyle\sum_{e' \in D_e \cap \mathcal{S}_h} \widehat{q}_{e'} \\
&\quad = \widehat{q}_e + \textstyle\sum_{e' \in D_e \cap \mathcal{S}_h} \left(\phi_h(v_{e'}^+) - \phi_h(v_{e'}^-)\right) \\
&\quad = \widehat{q}_e + (\phi_h(v_e^-) - \phi_h(v_e^+)) = \int_e \mathbf{Z}_h \cdot \boldsymbol{\tau} \,,
\end{aligned}$$

and thus the degrees of freedom of $\mathbf{Z}_h$ are given by

$$\int_e \mathbf{Z}_h \cdot \boldsymbol{\tau} = LK(D_e, \widehat{\sigma}_j) \,.$$

In particular, the loop field $\mathbf{Z}_h$ thus defined satisfies problem (5) with $\kappa_n = m_{n,j} = LK(\sigma_n, \widehat{\sigma}_j)$, a non-singular matrix.

- Selecting $j = 1, \ldots, g$ we have an explicit formula for a basis of the first de Rham cohomology group.

Webb–Forghani algorithm and linking numbers

Since a linking number is a sum of simple double integrals, its computation can be done efficiently (see Bertolazzi and Ghiloni [2012]).
However, for a fine mesh it is too expensive if used for all the edges (not belonging to the spanning tree...).

- Recipe: when the Webb–Forghani algorithm stops, use the formula for computing the value of one single unknown, and restart the algorithm.

Numerical experiments show that the use of the explicit formula is necessary very few times [one for Test D and Test E, four for Test F].

Geometries

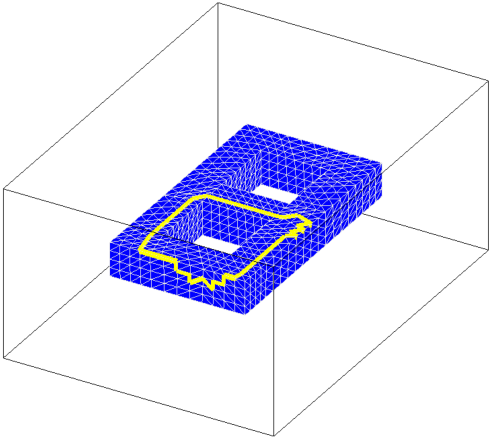


Figure : Case A: 2-torus (one homological cycle σ_n is drawn).

Geometries (cont'd)

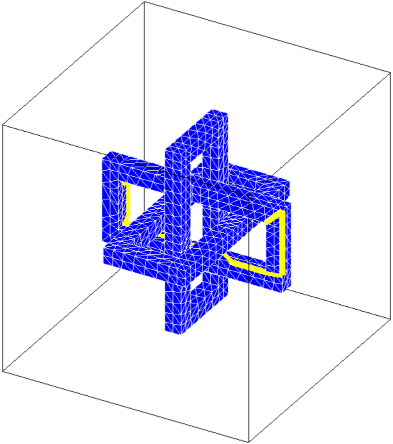


Figure : Case B: Borromean rings (one homological cycle σ_n is drawn).

Geometries (cont'd)

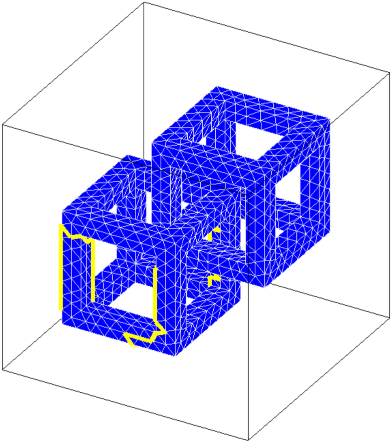


Figure : Case C: two-5-tori link (one homological cycle σ_n is drawn).

Geometries (cont'd)

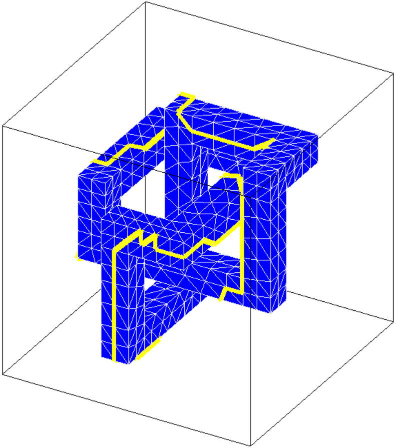


Figure : Case D: trefoil knot (one homological cycle σ_n is drawn).

Geometries (cont'd)

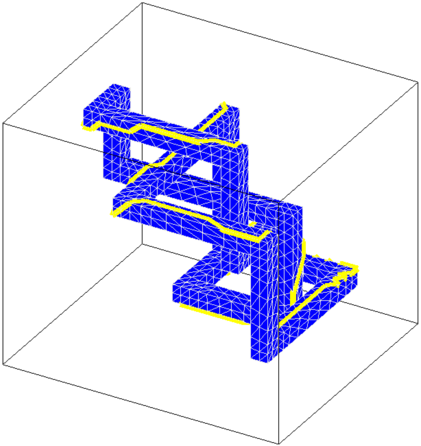


Figure : Case E: knot 4_1 (one homological cycle σ_n is drawn).

Geometries (cont'd)

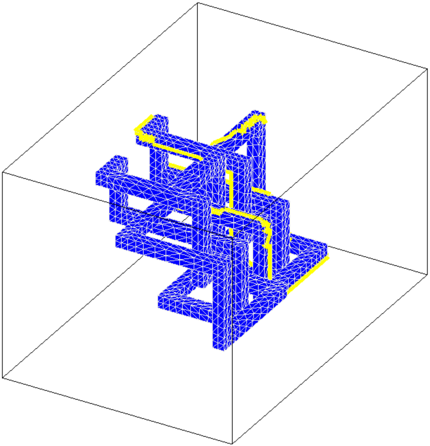


Figure : Case F: two-4_1-knots link (one homological cycle σ_n is drawn).

Numerical results

	Mesh 1		Mesh 2		Mesh 3	
	n_e	ms	n_e	ms	n_e	ms
Test A	42200	138	325904	868	2560416	6770
Test B	35380	93	273348	586	2147096	4397
Test C	25768	293	195256	1318	1517328	7434
Test D	15349	79	116170	294	902388	2016
Test E	34372	144	264548	749	2073688	4760
Test F	80504	310	624352	2671	4913792	12723

Table : CPU time for computing all the homological cycles σ_n and $\widehat{\sigma}_n$.

Numerical results (cont'd)

	n_e	$n_e - \#L$	$n_e^{(1)}$	$n_e^{(2)}$	$\#CC$
Test A	2560416	2185729	58987	0	-
Test B	2147096	1832896	110245	0	-
Test C	1517328	1292168	124239	0	-
Test D	902388	768384	54273	34506	30
Test E	2073688	1769408	150694	98603	107
Test F	4913792	4196608	275832	212088	145

Table : Reduction of the number of unknowns.

[n_e: number of edges; $\#L$: number of spanning tree edges; $n_e^{(1)}$: number of unknowns left after the algorithm has stopped; $n_e^{(2)}$: number of unknowns left after having used the homological equations; $\#CC$: number of connected components of dual graph.]

Numerical results (cont'd)

	n_e	loop fields		source field
Test A	2560416	(2)	9659	9937
Test B	2147096	(3)	9447	8822
Test C	1517328	(10)	28187	6322
Test D	902388	(1)	3759	3814
Test E	2073688	(1)	8705	8907
Test F	4913792	(2)	37338	22210

Table : CPU time (ms) for computing all the loop fields (their number is indicated in parenthesis) and one source field.

Computed loop fields

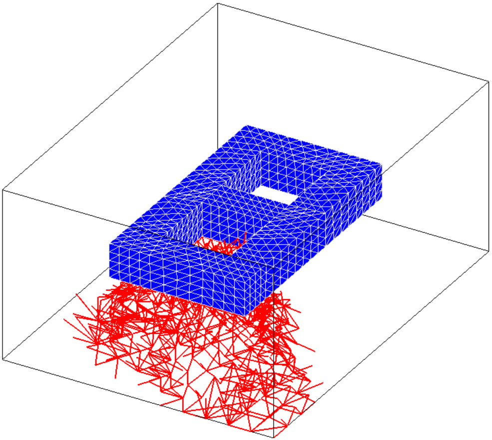


Figure : Support of a loop field. Case A: 2-torus.

Computed loop fields (cont'd)

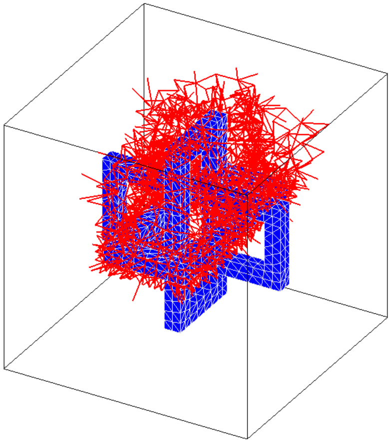


Figure : Support of a loop field. Case B: Borromean rings.

Computed loop fields (cont'd)

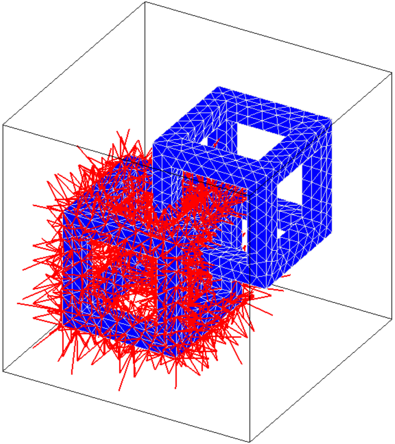


Figure : Support of a loop field. Case C: two-5-tori link.

Computed loop fields (cont'd)

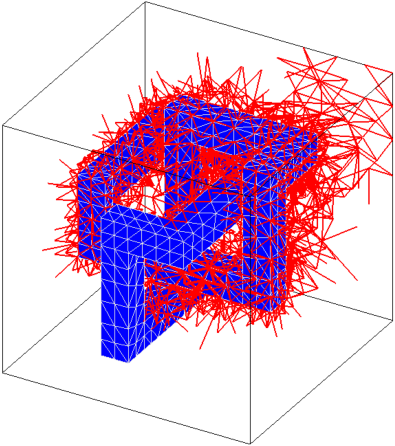


Figure : Support of a loop field. Case D: trefoil knot.

Computed loop fields (cont'd)

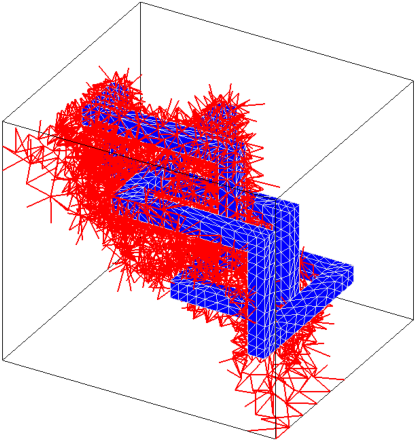


Figure : Support of a loop field. Case E: knot 4_1.

Computed loop fields (cont'd)

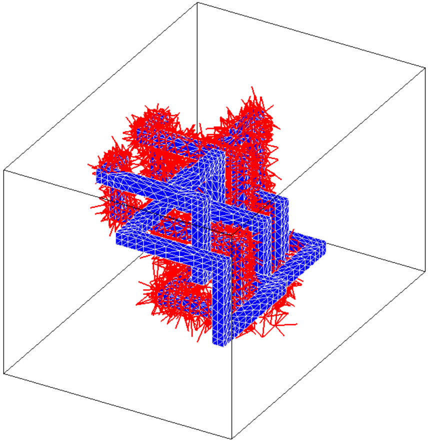


Figure : Support of a loop field. Case F: two-4_1-knots link.

References

- Z. Badics and Z.J. Cendes, *Source modeling by mesh incidence matrices*, IEEE Trans. Magn., 43 (2007), 1241–1244.
- E. Bertolazzi and R. Ghiloni, *Fast computation of the linking number via an exact explicit formula*, in preparation, 2012.
- A. Bossavit, *Computational Electromagnetism*, Academic Press, San Diego, 1998.
- P. Dłotko, R. Specogna and F. Trevisan, *Automatic generation of cuts on large-sized meshes for the T-Ω geometric eddy-current formulation*, Comput. Methods Appl. Mech. Engrg., 198 (2009), 3765–3781.

References (cont'd)

- P. Dłotko and R. Specogna, *Critical analysis of the spanning tree techniques*, SIAM J. Numer. Anal., 48 (2010), 1601–1624.
- P. Dular, F. Henrotte, F. Robert, A. Genon and W. Legros, *A generalized source magnetic field calculation method for inductors of any shape*, IEEE Trans. Magn., 33 (1997), 1398–1401.
- P. Dular, *Curl-conform source fields in finite element formulations. Automatic construction of a reduced form*, COMPEL, 24 (2005), 364–373.
- C.S. Harrold and J. Simkin, *Cutting multiply connected domains*, IEEE Trans. Magn., 21 (1985), 2495–2498.

References (cont'd)

- P.R. Kotiuga, *On making cuts for magnetic scalar potentials in multiply connected regions*, J. Appl. Phys., 61 (1987), 3916–3918.
- P.R. Kotiuga, *Toward an algorithm to make cuts for magnetic scalar potentials in finite element meshes*, J. Appl. Phys., 63 (1988), 3357–3359. Erratum: J. Appl. Phys., 64 (1988), 4257.
- P.R. Kotiuga, *An algorithm to make cuts for scalar potentials in tetrahedral meshes based on the finite element method*, IEEE Trans. Magn., 25 (1989), 4129–4131.
- Y. Le Ménach, S. Clénet and F. Piriou, *Determination and utilization of the source field in 3D magnetostatic problems*, IEEE Trans. Magn., 34 (1998), 2509–2512.

References (cont'd)

- P.J. Leonard, H.C. Lai, R.J. Hill-Cottingham and D. Rodger, *Automatic implementation of cuts in multiply connected magnetic scalar regions for 3D eddy current models*, IEEE Trans. Magn., 29 (1993), 1368–1371.
- K. Preis, I. Bardi, O. Bíró, C. Magele, G. Vrisk and K.R. Richter, *Different finite element formulations of 3D magnetostatic fields*, IEEE Trans. Magn., 28 (1992), 1056–1059.
- F. Rapetti, F. Dubois and A. Bossavit, *Discrete vector potentials for nonsimply connected three-dimensional domains*, SIAM J. Numer. Anal., 41 (2003), 1505-1527.
- Z. Ren, *T-Ω formulation for eddy-current problems in multiply connected regions*, IEEE Trans. Magn., 38 (2002), 557–560.

References (cont'd)

- J. Simkin, S.C. Taylor and E.X. Xu, *An efficient algorithm for cutting multiply connected regions*, IEEE Trans. Magn., 40 (2004), 707–709.
- J.P. Webb and B. Forghani, *A single scalar potential method for 3D magnetostatics using edge elements*, IEEE Trans. Magn., 25 (1989), 4126–4128.