

CONSTRUCTION OF A FINITE ELEMENT BASIS OF THE FIRST DE RHAM COHOMOLOGY GROUP AND NUMERICAL SOLUTION OF 3D MAGNETOSTATIC PROBLEMS[‡]

ANA ALONSO RODRÍGUEZ[†], ENRICO BERTOLAZZI*, RICCARDO GHILONI[†], AND
ALBERTO VALLI[†]

Abstract. We devise an efficient algorithm for the finite element construction of discrete harmonic fields and the numerical solution of 3D magnetostatic problems. In particular, we construct a finite element basis of the first de Rham cohomology group of the computational domain. The proposed method works for general topological configurations and does not need the determination of “cutting” surfaces.

Key words. Edge finite elements, discrete harmonic fields, first de Rham cohomology group, magnetostatics, loop fields, source fields.

AMS subject classifications. 65N30, 65N12, 35Q60, 35J46, 14F40, 55N99

1. Introduction. The aim of this paper is two-fold. First, employing the edge finite elements introduced by Nédélec [49], we construct a discrete approximation of the space of harmonic fields

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

where Ω is a bounded three-dimensional domain with a Lipschitz boundary, \mathbf{n} is the outward unit normal vector to $\partial\Omega$, and μ is a symmetric matrix, uniformly positive definite in Ω and with entries in $L^\infty(\Omega)$ (in physical applications, the magnetic permeability).

In particular, we give a simple and 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 (therefore, there exists a loop in Ω such that the line integral of \mathbf{T}_0 on it is different from 0). These fields are of central importance for numerical electromagnetism in general topological domains (see, e.g., Kotiuga [40], Kettunen et al. [37]; see Bossavit [13], Gross and Kotiuga [31]). To make precise one of their most important properties, let us first give a definition: if the only linear combination of a maximal set of loop fields that equals a gradient is the trivial one, we say that those loop fields are linearly cohomologically independent. It is known that a maximal set of linearly cohomologically independent loop fields gives a basis of the first de Rham cohomology group of Ω , namely, the quotient space between curl-free vector fields and gradients defined in Ω .

Second, we furnish a finite element numerical solution to the magnetostatic problem, that reads as follows: given a divergence-free current density \mathbf{J} , with vanishing normal flux on all the connected components of $\partial\Omega$, find a magnetic field \mathbf{H} that satisfies

$$\begin{aligned} \operatorname{curl} \mathbf{H} &= \mathbf{J}, & \text{in } \Omega \\ \operatorname{div}(\mu \mathbf{H}) &= 0, & \text{in } \Omega \\ \mu \mathbf{H} \cdot \mathbf{n} &= 0, & \text{on } \partial\Omega. \end{aligned} \tag{1.1}$$

[†]Dipartimento di Matematica, Università di Trento, I-38123 Povo (Trento), Italy

*Dipartimento di Ingegneria Meccanica e Strutturale, Università di Trento, I-38123 Mesiano (Trento), Italy

In particular, the vector fields satisfying $\text{curl } \mathbf{H}_e = \mathbf{J}$ in Ω are often called *source fields* in the electromagnetic literature, and are needed for formulating eddy current problems in terms of a magnetic scalar potential in the insulating region (see, e.g., Bossavit [13], Gross and Kotiuga [31], Alonso Rodríguez and Valli [3]), or for solving the magnetostatic problem.

Let us start by describing in more detail the first problem, namely, the approximation of $\mathbb{H}_\mu(\Omega)$. It is well-known that the dimension of this vector space is equal to the first Betti number of Ω , that we will denote by g (see, e.g., Bossavit [13], Hiptmair [35], Gross and Kotiuga [31]). The first Betti number is the rank of the first homology group of Ω , that is, the number of a maximal set of independent non-bounding cycles in Ω ; it is also the dimension of the first de Rham cohomology group of Ω .

A theoretical way for determining a basis of $\mathbb{H}_\mu(\Omega)$ is well-known (see, e.g., Foias and Temam [29], Bossavit [12], Amrouche et al. [5]), and is grounded on the fact that there exist g connected orientable Lipschitz surfaces Σ_n , with $\partial\Sigma_n \subset \partial\Omega$, each one “cutting” a non-bounding cycle in $\bar{\Omega}$. The construction procedure reads as follows. First, we can associate a loop field to any cutting surface Σ_n : having denoted by $[\cdot]_{\Sigma_n}$ the jump across the surface Σ_n , and taking a function φ_n^* that is piecewise-smooth in $\Omega \setminus \Sigma_n$ and satisfies $[\varphi_n^*]_{\Sigma_n} = 1$, we set $\mathbf{T}_{0,n}^*$ the $(L^2(\Omega))^3$ -extension of $\text{grad } \varphi_n^*$, where the (distributional) gradient has been computed in $\Omega \setminus \Sigma_n$. It is clear that $\mathbf{T}_{0,n}^*$ has line integral equal to 1 on the non-bounding cycle that has been cut by the surface Σ_n ; therefore, it is a loop field. It is worth remarking that the function φ_n^* can be a discrete function (say, a finite element function); as a consequence, $\mathbf{T}_{0,n}^*$ can be a finite element vector field. Second, we set $\boldsymbol{\rho}_n = \mathbf{T}_{0,n}^* + \text{grad } \psi_n$, where $\psi_n \in H^1(\Omega)$ is the solution (uniquely determined up to an additive constant) to the classical Neumann problem

$$\int_{\Omega} \mu \text{grad } \psi_n \cdot \text{grad } \phi = - \int_{\Omega} \mu \mathbf{T}_{0,n}^* \cdot \text{grad } \phi, \quad \forall \phi \in H^1(\Omega), \quad (1.2)$$

having introduced the Sobolev space $H^1(\Omega) = \{\phi \in L^2(\Omega) \mid \text{grad } \phi \in (L^2(\Omega))^3\}$. The vector functions $\boldsymbol{\rho}_n$, $n = 1, 2, \dots, g$, are a set of basis functions of $\mathbb{H}_\mu(\Omega)$. Moreover, if the loop field $\mathbf{T}_{0,n}^*$ is a finite element, taking a finite element discretization of (1.2) we can construct a set of discrete fields $\boldsymbol{\rho}_{n,h} = \mathbf{T}_{0,n}^* + \text{grad } \psi_{n,h}$, finite element approximations of the basis functions $\boldsymbol{\rho}_n$.

It is now clear that the crucial point in the construction or in the approximation of the space of harmonic fields $\mathbb{H}_\mu(\Omega)$ is the knowledge of a maximal set of linearly cohomologically independent loop fields, and to this aim it is enough to determine the “cutting” surfaces Σ_n . There is an extensive literature concerning their construction (see Kotiuga [40], [41], [42], Harrold and Simkin [32], Leonard et al. [44], Ren [54], Simkin et al. [59], Dular [27]). However, in general topological situations (for instance, in the case of domains that are the complement of “knotted” domains) and for realized finite element meshes this construction is not feasible, as it can be quite expensive from the computational point of view (see Bossavit [13], Dłotko et al. [26]).

Therefore, it is interesting to propose, as we do in this paper taking inspiration from Ghiloni [30], an alternative procedure for the determination of a maximal set of linearly cohomologically independent discrete loop fields. Our method avoids the use of “cutting” surfaces and instead is based on the explicit knowledge of a maximal set of independent non-bounding cycles on $\partial\Omega$; in other words, we only require the construction of a basis of the first homology group of $\partial\Omega$. We use a *spanning tree*

(similar but different techniques, based on the so-called *belted tree*, have been proposed by several authors, but they do not work for all topological situations: see Ren and Razek [55], Kettunen et al. [38], Bossavit [13], Rapetti et al. [53], Henrotte and Hameyer [34], Dłotko et al. [26]). Another fundamental tool is the direct algorithm of Webb and Forghani [62]: however, since it is known to fail in certain topological situations (see Dłotko and Specogna [21]), we modify it in a suitable way, in order to be able to construct the finite element loop fields for every domain Ω . A basic point here is the fact that we provide an explicit formula for expressing the discrete loop fields in terms of *linking numbers*.

Before concluding this survey, we want to recall that two other recent approaches have been proposed for computing cohomology generators, one based on algebraic techniques (Dłotko and Specogna [22]) and one more similar to our (Dłotko and Specogna [25]); the first one is proved to work for any topological situation, while the second one, though not completely general, is shown to be computationally more efficient than the former.

Going back to the second topic, namely, focusing on problem (1.1), it is readily seen that it is not well-posed, as uniqueness fails (just add a harmonic field to a solution \mathbf{H}). The complete problem reads: given $\mathbf{J} \in (L^2(\Omega))^3$ with $\operatorname{div} \mathbf{J} = 0$ in Ω and $\int_{(\partial\Omega)_r} \mathbf{J} \cdot \mathbf{n} = 0$ for $r = 0, 1, \dots, p$, where $(\partial\Omega)_r$ are the connected components of $\partial\Omega$, find the magnetic field \mathbf{H} such that

$$\begin{aligned} \operatorname{curl} \mathbf{H} &= \mathbf{J}, & \text{in } \Omega \\ \operatorname{div}(\mu \mathbf{H}) &= 0, & \text{in } \Omega \\ \mu \mathbf{H} \cdot \mathbf{n} &= 0, & \text{on } \partial\Omega \\ \int_{\Omega} \mu \mathbf{H} \cdot \boldsymbol{\eta} &= 0, & \forall \boldsymbol{\eta} \in \mathbb{H}_{\mu}(\Omega). \end{aligned} \tag{1.3}$$

This problem has a solution (see, e.g., Saranen [57]), and uniqueness now is straightforward.

Let us also note that the complete curl-div problem, with a non-vanishing datum in the second and in the third equation, can be solved by adding to the solution \mathbf{H} of problem (1.3) the gradient of the solution of a standard Neumann problem. Therefore, when considering the complete curl-div system, the solution of problem (1.3) is anyhow the most important step.

Let us devise a suitable variational formulation of problem (1.3). Using the notation $H^0(\operatorname{curl}; \Omega) = \{\mathbf{z} \in (L^2(\Omega))^3 \mid \operatorname{curl} \mathbf{z} = \mathbf{0}\}$, it is well-known that any vector function $\mathbf{z} \in H^0(\operatorname{curl}; \Omega)$ can be written as

$$\mathbf{z} = \operatorname{grad} \phi + \boldsymbol{\eta}, \tag{1.4}$$

where $\phi \in H^1(\Omega)$, $\boldsymbol{\eta} \in \mathbb{H}_{\mu}(\Omega)$ and therefore $\int_{\Omega} \mu \operatorname{grad} \phi \cdot \boldsymbol{\eta} = 0$ (see, e.g., Alonso Rodríguez and Valli [3]). It is readily verified that an equivalent formulation of problem (1.3) is: find $\mathbf{H} \in (L^2(\Omega))^3$ such that

$$\operatorname{curl} \mathbf{H} = \mathbf{J}, \quad \text{in } \Omega \tag{1.5a}$$

$$\int_{\Omega} \mu \mathbf{H} \cdot \mathbf{z} = 0, \quad \forall \mathbf{z} \in H^0(\operatorname{curl}; \Omega). \tag{1.5b}$$

In fact, taking in (1.5b) the test function $\mathbf{z} = \operatorname{grad} \phi$, with $\phi \in H^1(\Omega)$, and integrating by parts we have $0 = \int_{\Omega} \mu \mathbf{H} \cdot \operatorname{grad} \phi = - \int_{\Omega} \operatorname{div}(\mu \mathbf{H}) \phi + \int_{\partial\Omega} \mu \mathbf{H} \cdot \mathbf{n} \phi$. When $\phi|_{\partial\Omega} = 0$

it follows $\operatorname{div}(\mu \mathbf{H}) = 0$ in Ω ; hence we also have $\int_{\partial\Omega} \mu \mathbf{H} \cdot \mathbf{n} \phi = 0$ for each $\phi \in H^1(\Omega)$, yielding $\mu \mathbf{H} \cdot \mathbf{n} = 0$ on $\partial\Omega$.

Though formulation (1.5) looks quite simple, to our knowledge it has not been used as the starting point for devising an efficient numerical approximation algorithm for magnetostatics. This is what we propose in this paper.

With respect to this issue, let us start with a general overview. The finite element numerical approximation of the magnetostatic problem (1.3) has been considered since a long time, though very often in simple topological situation, as it is probably the “most frequently encountered field problem in electrical engineering design” (see Chari et al. [17]). We cannot present here an exhaustive description of the various methods employed; however, we want to mention some of the most important, in order to show the advantage of the finite element method based on (1.5).

A formulation in terms of a vector potential \mathbf{A} such that $\operatorname{curl} \mathbf{A} = \mu \mathbf{H}$ is quite classical, and has been analyzed by Coulomb [18], Barton and Cendes [8], Preis et al. [52] (see also the new point of view involving mimetic finite differences presented in Brezzi and Buffa [14], Lipnikov et al. [45]): since the unknown is a vector field, the computational cost is higher than that needed to solve problem (1.5), that, as we will see in (2.1)–(2.2), is essentially a scalar problem. The same remark holds for the least squares approach of Chang and Gunzburger [16] and the even more expensive mixed methods of Kikuchi [39], Perugia [51], and Alotto and Perugia [4].

The co-volume method proposed by Nicolaides and Wu [50] is based on a system of two orthogonal grids like the classical Voronoi–Delaunay mesh pair, and for this reason this approach is not completely general, as some restrictions on the primal mesh and on the topological properties of the computational domain are needed.

Finally, the methods based on a magnetic scalar “potential” (the so-called reduced scalar potential) require the preliminary determination of a source field \mathbf{H}_e . Doing this by means of the Biot–Savart formula is not cheap from the computational point of view, and sometimes it induces cancellation errors (see Simkin and Trowbridge [60], Mayergoyz et al. [46], where it was proposed how to avoid this drawback by introducing an additional scalar potential; a complete analysis of this more complex formulation is in Bermudez et al. [10]). Let us also recall that a detailed presentation of the reasons behind these cancellation errors is given in Balac and Caloz [7].

We follow a different point of view. We start noting that a finite element approximation of (1.5) is standard provided that: (i) we know a discrete source field $\mathbf{H}_{e,h}$ satisfying $\operatorname{curl} \mathbf{H}_{e,h} = \mathbf{J}_h$, where \mathbf{J}_h is a finite element approximation of the current density \mathbf{J} ; (ii) a suitable finite dimensional subspace of $H^0(\operatorname{curl}; \Omega)$ is available.

With respect to the latter point (ii), we mimic the decomposition (1.4), and we consider the discrete functions $\mathbf{z}_h = \operatorname{grad} \phi_h + \sum_{n=1}^g \xi_n \mathbf{T}_{0,n}$, where $\mathbf{T}_{0,n}$ are suitable finite element loop fields. Note that in this way we have lost the orthogonality relation $\int_{\Omega} \mu \operatorname{grad} \phi \cdot \boldsymbol{\eta} = 0$, that was true for the decomposition in (1.4), but this is not essential for our arguments. We prove that the error between the exact and the discrete solutions is bounded by the approximation error, uniformly with respect to the mesh size.

We are finally left with point (i), namely, the determination of the discrete source fields. This problem has been widely considered, mainly for simple topological domains (see, e.g., Webb and Forghani [62], Preis et al. [52], Dular et al. [28], Le Ménach et al. [43], Rapetti et al. [53], Dular [27], Badics and Cendes [6], Dłotko and Specogna [21]). We show that a discrete source field can be determined by adopting a similar procedure to that employed for the construction of the finite element

loop fields: again, the main point is the use of the Webb–Forghani algorithm, followed by the introduction of a graph for the edges whose degree of freedom has not been yet determined when the algorithm stops and by a simple algebraic direct solver (for a similar approach, see also Dłotko and Specogna [24]).

REMARK 1. *Let us note that we could also consider harmonic fields satisfying different boundary conditions, for instance $\mu\mathbf{H} \cdot \mathbf{n} = 0$ on Γ_1 and $\mathbf{H} \times \mathbf{n} = \mathbf{0}$ on Γ_2 , where $\Gamma_1 \cup \Gamma_2 = \partial\Omega$ and $\Gamma_1 \cap \Gamma_2 = \emptyset$. Similarly, in the magnetostatic problem this set of boundary conditions could replace $\mu\mathbf{H} \cdot \mathbf{n} = 0$ on $\partial\Omega$. The results we present here can be easily adapted to these situations.*

This paper is organized as follows. In Section 2 we introduce and analyze the finite element approximation of the magnetostatic problem (1.5). Section 3 is devoted to the description of some algebraic topology concepts and to the analysis of the fundamental discrete problem, that is the main tool for the construction of source fields and loop fields performed in Section 4. An explicit formula for describing the loop fields in terms of linking numbers is presented in Section 5. Section 6 contains the construction of the bases of the first homology groups of $\bar{\Omega}$ and $\mathbb{R}^3 \setminus \Omega$, whose knowledge is needed for defining the fundamental discrete problem and for obtaining the explicit expression of the loop fields. Finally, in Section 7 we present some numerical results that illustrate the performances of the devised algorithms.

2. Finite element approximation. Let Ω be a bounded Lipschitz polyhedral domain equipped with a tetrahedral triangulation $\mathcal{T}_h = (V, E, F, T)$ of $\bar{\Omega}$. V is the set of vertices, E the set of edges, F the set of faces and T the set of tetrahedra in \mathcal{T}_h .

We consider the following spaces of finite elements:

- The space L_h of continuous piecewise linear finite elements. Its dimension is n_v , the number of vertices in \mathcal{T}_h .
- The space N_h of Nédélec edge elements of degree 1. Its dimension is n_e , the number of edges in \mathcal{T}_h .
- The space RT_h of Raviart–Thomas finite elements of degree 1. Its dimension is n_f , the number of faces in \mathcal{T}_h .

It is well-known that $L_h \subset H^1(\Omega)$, $N_h \subset H(\text{curl}; \Omega)$ and $RT_h \subset H(\text{div}; \Omega)$, where

$$\begin{aligned} H(\text{curl}; \Omega) &= \{\mathbf{v} \in (L^2(\Omega))^3 \mid \text{curl } \mathbf{v} \in (L^2(\Omega))^3\}, \\ H(\text{div}; \Omega) &= \{\mathbf{v} \in (L^2(\Omega))^3 \mid \text{div } \mathbf{v} \in L^2(\Omega)\}. \end{aligned}$$

Moreover $\text{grad } L_h \subset N_h$ and $\text{curl } N_h \subset RT_h$ (see, e.g., Monk [47]).

For determining suitable basis functions of these spaces, let us fix a total ordering v_1, \dots, v_{n_v} of the elements of V . This induces an orientation on the elements of E and F . If the end points of the edge e are v_i and v_j , with $1 \leq i < j \leq n_v$, the oriented edge denoted by $[v_i^-, v_j^+]$ is such that $v_i^- = v_i$ and $v_j^+ = v_j$. If the vertices of the face f are v_i, v_j and v_k with $1 \leq i < j < k \leq n_v$, the oriented face denoted by $[v_i^-, v_j^+, v_k^+]$ is such that $v_i^- = v_i$, $v_j^+ = v_j$ and $v_k^+ = v_k$. The unit tangent vector of oriented edge e is given by $\boldsymbol{\tau} = (v_j^+ - v_i^-) / |v_j^+ - v_i^-|$, and the unit normal vector $\boldsymbol{\nu}$ of the oriented face f is obtained by the right hand rule.

We choose a basis $\{\Phi_{h,1}, \dots, \Phi_{h,n_v}\}$ of L_h , a basis $\{\mathbf{w}_{h,1}, \dots, \mathbf{w}_{h,n_e}\}$ of N_h , and a basis $\{\mathbf{r}_{h,1}, \dots, \mathbf{r}_{h,n_f}\}$ of RT_h , such that $\Phi_{h,i}(v_j) = \delta_{i,j}$ for $1 \leq i, j \leq n_v$, $\int_{e_j} \mathbf{w}_{h,i} \cdot \boldsymbol{\tau} = \delta_{i,j}$ for $1 \leq i, j \leq n_e$, $\int_{f_j} \mathbf{r}_{h,i} \cdot \boldsymbol{\nu} = \delta_{i,j}$ for $1 \leq i, j \leq n_f$.

The finite element approximation of (1.5) reads as follows. Denoting as before the connected components of $\partial\Omega$ by $(\partial\Omega)_r$, $r = 0, 1, \dots, p$, and given $\mathbf{J}_h \in RT_h$, a

suitable approximation of \mathbf{J} , with $\operatorname{div} \mathbf{J}_h = 0$ and $\int_{(\partial\Omega)_r} \mathbf{J}_h \cdot \mathbf{n} = 0$, find $\mathbf{H}_h \in N_h$ such that

$$\begin{aligned} \operatorname{curl} \mathbf{H}_h &= \mathbf{J}_h, & \text{in } \Omega \\ \int_{\Omega} \mu \mathbf{H}_h \cdot \mathbf{z}_h &= 0, & \forall \mathbf{z}_h \in N_h \cap H^0(\operatorname{curl}; \Omega). \end{aligned} \quad (2.1)$$

Assuming that a function $\mathbf{H}_{e,h} \in N_h$ such that $\operatorname{curl} \mathbf{H}_{e,h} = \mathbf{J}_h$ is known, this problem can be easily rewritten as follows: find $\mathbf{K}_h \in N_h \cap H^0(\operatorname{curl}; \Omega)$ such that

$$\int_{\Omega} \mu \mathbf{K}_h \cdot \mathbf{z}_h = - \int_{\Omega} \mu \mathbf{H}_{e,h} \cdot \mathbf{z}_h, \quad \forall \mathbf{z}_h \in N_h \cap H^0(\operatorname{curl}; \Omega),$$

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

In the next section we present a general strategy for the computation of a suitable source field $\mathbf{H}_{e,h}$ and a basis of $N_h \cap H^0(\operatorname{curl}; \Omega)$. The latter is based on the knowledge of a maximal set of linearly cohomologically independent discrete loop fields: if they are denoted by $\mathbf{T}_{0,j}$, $j = 1, 2, \dots, g$, a basis of $N_h \cap H^0(\operatorname{curl}; \Omega)$ is given by $\{\operatorname{grad} \Phi_{h,1}, \dots, \operatorname{grad} \Phi_{h,n_v-1}\} \cup \{\mathbf{T}_{0,1}, \dots, \mathbf{T}_{0,g}\}$ (see Theorem 3). Thus, the solution of problem (2.1) is then determined by setting

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

where from (2.1) the scalars $\beta_i \in \mathbb{R}$, $i = 1, 2, \dots, n_v - 1$, and $\eta_j \in \mathbb{R}$, $j = 1, 2, \dots, g$, must satisfy:

$$\begin{aligned} \sum_{i=1}^{n_v-1} \beta_i \int_{\Omega} \mu \operatorname{grad} \Phi_{h,i} \cdot \operatorname{grad} \Phi_{h,l} + \sum_{j=1}^g \eta_j \int_{\Omega} \mu \mathbf{T}_{0,j} \cdot \operatorname{grad} \Phi_{h,l} \\ = - \int_{\Omega} \mu \mathbf{H}_{e,h} \cdot \operatorname{grad} \Phi_{h,l}, \quad \forall l = 1, 2, \dots, n_v - 1 \\ \sum_{i=1}^{n_v-1} \beta_i \int_{\Omega} \mu \operatorname{grad} \Phi_{h,i} \cdot \mathbf{T}_{0,n} + \sum_{j=1}^g \eta_j \int_{\Omega} \mu \mathbf{T}_{0,j} \cdot \mathbf{T}_{0,n}, \\ = - \int_{\Omega} \mu \mathbf{H}_{e,h} \cdot \mathbf{T}_{0,n} \quad \forall n = 1, 2, \dots, g. \end{aligned} \quad (2.2)$$

The solution of problem (2.2) is quite standard and computationally cheap once the source field $\mathbf{H}_{e,h}$ and the loop fields $\mathbf{T}_{0,j}$ are available. Therefore in the following sections we will focus only on these two issues.

From the theoretical point of view, it is straightforward to furnish an error estimate. Let us denote by Π^{RT_h} and Π^{N_h} the interpolation operators defined for smooth functions and valued in RT_h and N_h , respectively.

THEOREM 1. *Assume that \mathbf{J} and the solution \mathbf{H} of problem (1.5) are smooth. Then the solution \mathbf{H}_h of problem (2.1) with $\mathbf{J}_h = \Pi^{RT_h} \mathbf{J}$ satisfies the following error estimate*

$$\|\mathbf{H} - \mathbf{H}_h\|_0 + \|\operatorname{curl} \mathbf{H} - \operatorname{curl} \mathbf{H}_h\|_0 \leq C \|\mathbf{H} - \Pi^{N_h} \mathbf{H}\|_0 + \|\mathbf{J} - \Pi^{RT_h} \mathbf{J}\|_0. \quad (2.3)$$

where $\|\cdot\|_0$ denotes the norm in $(L^2(\Omega))^3$.

Proof. We notice that $\int_{\Omega} \mu (\mathbf{H} - \mathbf{H}_h) \cdot \mathbf{z}_h = 0$ for all $\mathbf{z}_h \in N_h \cap H^0(\operatorname{curl}; \Omega)$, hence

$$\|\mathbf{H} - \mathbf{H}_h\|_0^2 \leq C_1 \int_{\Omega} \mu (\mathbf{H} - \mathbf{H}_h) \cdot (\mathbf{H} - \mathbf{H}_h) = C_1 \int_{\Omega} \mu (\mathbf{H} - \mathbf{H}_h) \cdot (\mathbf{H} - \mathbf{v}_h),$$

for each $\mathbf{v}_h \in V_h(\mathbf{J}_h) := \{\mathbf{v}_h \in N_h \mid \text{curl } \mathbf{v}_h = \mathbf{J}_h\}$. Thus we have the optimal error estimate

$$\|\mathbf{H} - \mathbf{H}_h\|_0 \leq C \inf_{\mathbf{v}_h \in V_h(\mathbf{J}_h)} \|\mathbf{H} - \mathbf{v}_h\|_0.$$

Since \mathbf{J} and \mathbf{H} are smooth, the interpolants $\Pi^{RT_h} \mathbf{J}$ and $\Pi^{N_h} \mathbf{H}$ are well defined; therefore from $\mathbf{J}_h = \Pi^{RT_h} \mathbf{J}$ it clearly follows $\Pi^{N_h} \mathbf{H} \in V_h(\mathbf{J}_h)$, as $\text{curl}(\Pi^{N_h} \mathbf{H}) = \Pi^{RT_h}(\text{curl } \mathbf{H})$. In this case we conclude with (2.3), as $\text{curl } \mathbf{H} = \mathbf{J}$ and $\text{curl } \mathbf{H}_h = \mathbf{J}_h = \Pi^{RT_h} \mathbf{J}$. \square

We recall that the interpolants of \mathbf{J} and \mathbf{H} are well-defined if, for instance, \mathbf{J} and \mathbf{H} belong to $(H^{1/2+\delta}(\Omega))^3$ with $\delta > 0$.

3. The fundamental discrete problem. In the recent years many works investigate the use of algebraic topology techniques in computational electromagnetism, exploiting the geometrical nature of Maxwell equations (see, e.g., Bossavit [13], Tarhasaari and Kettunen [61], Hiptmair [35], Gross and Kotiuga [31], Dłotko and Specogna [23]).

In the following we introduce some notions of homology theory and graph theory. We consider a mesh $\mathcal{T}_h = (V, E, F, T)$ of $\bar{\Omega}$, having assigned the orientation to the edges and faces as explained before. The basic concept is that of chain: a 2-chain is a formal linear combination of oriented faces, a 1-chain a formal linear combination of oriented edges and a 0-chain a formal linear combination of vertices, in all cases taking the coefficients in \mathbb{Z} . We denote by $C_k(\mathcal{T}_h, \mathbb{Z})$ the abelian group of all the k -chains in \mathcal{T}_h , $k = 0, 1, 2$.

Now we can define the boundary operator $\partial_k : C_k(\mathcal{T}_h, \mathbb{Z}) \rightarrow C_{k-1}(\mathcal{T}_h, \mathbb{Z})$ for $k = 1, 2$. For the oriented face $f = [v_i, v_j, v_k]$ we have $\partial_2 f := [v_i, v_j] + [v_j, v_k] - [v_i, v_k]$, where $[v_r, v_s]$ denotes the oriented edge from the vertex v_r to the vertex v_s . Analogously for the oriented edge $e = [v_i, v_j]$ we have $\partial_1 e := v_j - v_i$. We extend the definition of the boundary operator to chains by linearity.

We also introduce some other notations that will be useful in the description of our algorithms. The orientation map $o_f(e) = \pm 1$ takes value 1 if the edge e appears with positive sign in $\partial_2 f$, and takes value -1 if the edge e appears with negative sign in $\partial_2 f$. The symbol $E\{f\}$ denotes the set of the oriented edges of the face f ; the symbol $F\{e\}$ denotes the set of the oriented faces f such that $e \subset f$; $V\{e\}$ denotes the set of the vertices of the edge e . Having this in mind, we can write $\partial_2 f = \sum_{e \in E\{f\}} o_f(e)e$.

A 1-chain c of \mathcal{T}_h is a 1-cycle if $\partial_1 c = 0$, and is a 1-boundary if there exists a 2-chain C such that $\partial_2 C = c$. Notice that all 1-boundaries are 1-cycles but, in general, not all 1-cycles are 1-boundaries.

Let us denote by $Z_1(\mathcal{T}_h, \mathbb{Z})$ the set of 1-cycles, $Z_1(\mathcal{T}_h, \mathbb{Z}) := \text{Ker}(\partial_1)$, and $B_1(\mathcal{T}_h, \mathbb{Z})$ the set of 1-boundaries, $B_1(\mathcal{T}_h, \mathbb{Z}) := \text{Im}(\partial_2)$. Two 1-cycles c and c' are called homologous in \mathcal{T}_h if $c - c'$ is a 1-boundary in \mathcal{T}_h . If c is homologous to the zero 1-cycle (namely, it is a 1-boundary), then we say that c bounds in \mathcal{T}_h .

The first homology group of \mathcal{T}_h consists of all homology classes of 1-cycles of \mathcal{T}_h , that is, it is the quotient group

$$\mathcal{H}_1(\mathcal{T}_h, \mathbb{Z}) = Z_1(\mathcal{T}_h, \mathbb{Z}) / B_1(\mathcal{T}_h, \mathbb{Z}).$$

It is well-known that, up to isomorphisms, $\mathcal{H}_1(\mathcal{T}_h, \mathbb{Z})$ depends only on $\bar{\Omega}$ and not on \mathcal{T}_h . For this reason, we can refer to $\mathcal{H}_1(\mathcal{T}_h, \mathbb{Z})$ as the first homology group of $\bar{\Omega}$, and

we can write $\mathcal{H}_1(\bar{\Omega}, \mathbb{Z})$ instead of $\mathcal{H}_1(\mathcal{T}_h, \mathbb{Z})$. This group is an abelian group of rank g , the first Betti number of $\bar{\Omega}$ (see Munkres [48, p. 24]). More precisely, it is a free abelian group isomorphic to \mathbb{Z}^g (see Gross and Kotiuga [31, Sect. 3D], and Benedetti et al. [9, Lemma 5.4]). It is worth remarking that $\mathcal{H}_1(\bar{\Omega}, \mathbb{Z})$ is also isomorphic to the (singular) homology group $\mathcal{H}_1(\Omega, \mathbb{Z})$ of Ω (see Munkres [48, Chap. 4] for the definition of the latter group).

The elements of $\mathcal{H}_1(\mathcal{T}_h, \mathbb{Z})$ are integer combinations of the homology classes of g cycles, denoted with $\{\sigma_n\}_{n=1}^g$, that are representatives of a basis of the homology group $\mathcal{H}_1(\mathcal{T}_h, \mathbb{Z})$. Thus, any cycle $\eta \in Z_1(\mathcal{T}_h, \mathbb{Z})$ can be written as $\eta = \beta + \sum_{n=1}^g \alpha_n \sigma_n$ with $\alpha_n \in \mathbb{Z}$ and $\beta \in B_1(\mathcal{T}_h, \mathbb{Z})$.

Given the tetrahedral triangulation $\mathcal{T}_h = (V, E, F, T)$ of $\bar{\Omega}$, let $\{\sigma_n\}_{n=1}^g$ be a set of 1-cycles that are representatives of a basis of the homology group $\mathcal{H}_1(\bar{\Omega}, \mathbb{Z})$. Let us also consider a spanning tree $\mathcal{S}_h = (V, L)$ of the (connected) graph (V, E) : it is a maximal subgraph of (V, E) (maximal because it visits all the vertices) without loops (this means that it is a tree).

We focus now on the *main problem* of our approach, namely: find $\mathbf{Z}_h \in N_h$ such that

$$\operatorname{curl} \mathbf{Z}_h = \mathbf{J}_h, \quad \text{in } \Omega, \quad (3.1a)$$

$$\oint_{\sigma_n} \mathbf{Z}_h \cdot \mathbf{ds} = \kappa_n, \quad \forall n = 1, 2, \dots, g \quad (3.1b)$$

$$\int_e \mathbf{Z}_h \cdot \boldsymbol{\tau} = 0, \quad \forall e \in L, \quad (3.1c)$$

where $\kappa_1, \dots, \kappa_g$ are real numbers.

The formulation of this problem uses a tree–cotree decomposition, similar to what has been previously done by many authors (see, for instance, Albanese and Rubinacci [1], Kettunen et al. [38], Bossavit [13], Henrotte and Hameyer [34], Henneron et al. [33]).

THEOREM 2. *Assume that $\mathbf{J}_h \in RT_h$, $\operatorname{div} \mathbf{J}_h = 0$ and $\int_{(\partial\Omega)_r} \mathbf{J}_h \cdot \mathbf{n} = 0$ for any connected component $(\partial\Omega)_r$ of $\partial\Omega$, $r = 0, 1, \dots, p$. Then problem (3.1) has a solution and this solution is unique.*

Proof. If \mathbf{Z}_h and $\tilde{\mathbf{Z}}_h$ are two different solutions of (3.1), then $\mathbf{Z}_h - \tilde{\mathbf{Z}}_h \in N_h$, $\operatorname{curl}(\mathbf{Z}_h - \tilde{\mathbf{Z}}_h) = \mathbf{0}$ and $\oint_{\sigma_n} (\mathbf{Z}_h - \tilde{\mathbf{Z}}_h) \cdot \mathbf{ds} = 0$ for all $n = 1, 2, \dots, g$. From the de Rham theorem for Whitney forms (see Hiptmair [35]) there exists $\psi_h \in L_h$ such that $\mathbf{Z}_h - \tilde{\mathbf{Z}}_h = \operatorname{grad} \psi_h$. Since the degrees of freedom of $\operatorname{grad} \psi_h$ are equal to 0 for each edge of the spanning tree \mathcal{S}_h , we can conclude that ψ_h is constant: in fact if the edge e belongs to \mathcal{S}_h we have $0 = \int_e \operatorname{grad} \psi_h \cdot \mathbf{ds} = \psi_h(v_e^+) - \psi_h(v_e^-)$, then $\psi_h(v_e^+) = \psi_h(v_e^-)$, and ψ_h is constant because \mathcal{S}_h is a spanning tree.

Concerning the existence of a solution, the assumptions on \mathbf{J}_h assure that there exists $\mathbf{H}^* \in (L^2(\Omega))^3$ such that $\operatorname{curl} \mathbf{H}^* = \mathbf{J}_h$, $\operatorname{div} \mathbf{H}^* = 0$ and $\mathbf{H}^* \cdot \mathbf{n} = 0$ in $\partial\Omega$ (see, e.g., Saranen [57]). Since Ω is a Lipschitz bounded polyhedral domain there exists $s_\Omega \in (1/2, 1)$ such that $H(\operatorname{curl}; \Omega) \cap H_0(\operatorname{div}; \Omega) \subset (H^{s_\Omega}(\Omega))^3$ (see, e.g., Alonso and Valli [2]). Hence $\mathbf{H}^* \in (H^{s_\Omega}(\Omega))^3$ and $\operatorname{curl} \mathbf{H}^* = \mathbf{J}_h \in (L^p(\Omega))^3$ for some $p > 2$, thus the Nédélec interpolant of \mathbf{H}^* , $\Pi^{N_h} \mathbf{H}^*$, is well defined (see Amrouche et al. [5], Monk [47, Lemma 5.38]).

Now we want to prove the existence of $\mathbf{W}_h \in N_h \cap H^0(\operatorname{curl}; \Omega)$ such that

$$\begin{aligned} \oint_{\sigma_n} \mathbf{W}_h \cdot \mathbf{ds} &= \kappa_n - \oint_{\sigma_n} \Pi^{N_h} \mathbf{H}^* \cdot \mathbf{ds}, & \forall n = 1, 2, \dots, g \\ \int_e \mathbf{W}_h \cdot \boldsymbol{\tau} &= - \int_e \Pi^{N_h} \mathbf{H}^* \cdot \boldsymbol{\tau}, & \forall e \in L. \end{aligned}$$

This is a linear system of $g + n_v - 1$ equations. It is well-known that the dimension of $N_h \cap H^0(\text{curl}; \Omega)$ is $g + n_v - 1$ (see, e.g., Hiptmair [35]), hence existence follows from uniqueness, that has been already proved. Finally, setting $\mathbf{Z}_h = \mathbf{W}_h + \Pi^{N_h} \mathbf{H}^*$ we have found a solution to (3.1). \square

Clearly, a field $\mathbf{H}_{e,h}$ such that $\text{curl} \mathbf{H}_{e,h} = \mathbf{J}_h$, namely, a discrete source field, can be computed solving (3.1), having chosen the constants κ_n in any arbitrary way.

On the other hand, as shown in the following theorem, a maximal set of linearly cohomologically independent finite element loop fields $\mathbf{T}_{0,j}$, $j = 1, 2, \dots, g$, can be determined by solving (3.1) with $\mathbf{J}_h = \mathbf{0}$ and $\kappa_n = M_{n,j}$, for any choice of a non-singular matrix \mathbf{M} with entries $M_{n,j}$. In particular, a basis of $N_h \cap H^0(\text{curl}; \Omega)$ can be computed starting from $\{\Phi_{h,1}, \dots, \Phi_{h,n_v}\}$, a basis of L_h .

THEOREM 3. *Let $\mathbf{T}_{0,j}$, $j = 1, 2, \dots, g$, be the solutions to problem (3.1) with $\mathbf{J}_h = \mathbf{0}$ and $\kappa_n = M_{n,j}$, where the matrix $\mathbf{M} = (M_{n,j})$ is non-singular. Then they are linearly cohomologically independent and the set*

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

is a basis of $N_h \cap H^0(\text{curl}; \Omega)$.

Proof. Since the dimension of $N_h \cap H^0(\text{curl}; \Omega)$ is equal to $g + n_v - 1$, it is enough to prove linear independence. If we have $\sum_{i=1}^{n_v-1} p_i \text{grad } \Phi_{h,i} + \sum_{j=1}^g q_j \mathbf{T}_{0,j} = \mathbf{0}$, it follows for all $n = 1, 2, \dots, g$

$$0 = \sum_{i=1}^{n_v-1} p_i \oint_{\sigma_n} \text{grad } \Phi_{h,i} \cdot \mathbf{ds} + \sum_{j=1}^g q_j \oint_{\sigma_n} \mathbf{T}_{0,j} \cdot \mathbf{ds} = \sum_{j=1}^g q_j M_{n,j},$$

hence, since \mathbf{M} is non-singular, $q_j = 0$ for each $j = 1, 2, \dots, g$. We thus have $\sum_{i=1}^{n_v-1} p_i \text{grad } \Phi_{h,i} = \mathbf{0}$, so that $\sum_{i=1}^{n_v-1} p_i \Phi_{h,i} = \text{const}$; the conclusion follows from the fact that $\Phi_{h,i}(v_{n_v}) = 0$ for each $i = 1, 2, \dots, n_v - 1$.

The proof that the loop fields $\mathbf{T}_{0,j}$ are linearly cohomologically independent follows the same argument. \square

4. The construction of source fields and loop fields. In this section we introduce an algorithm for solving problem (3.1). Since we are looking for a solution $\mathbf{Z}_h \in N_h$ of (3.1), we need to compute its degrees of freedom

$$q(e) = \int_e \mathbf{Z}_h \cdot \boldsymbol{\tau}, \quad \forall e \in E.$$

The first step is very simple: we just assign the value 0 to the degrees of freedom associated to any edge $e \in L$ (namely, belonging to the spanning tree), solving in this way (3.1c).

After that, the standard technique for taking into account (3.1b), namely, the homological equations, is based on the use of a *belted tree* instead of a spanning tree. Before continuing, we warn the reader that we do not follow this approach in this paper, since we prefer to follow a cheaper procedure that avoids the construction of a belted tree; however, for the sake of completeness, we present here the main features of this technique.

The notion of belted tree has been proposed by Ren and Razek [55] (see also Kettunen et al. [38], Bossavit [13], Rapetti et al. [53], Dłotko et al. [26]). A belted

tree $\mathcal{B}_h = (V, L \cup E^*)$ is a graph consisting in a spanning tree $\mathcal{S}_h = (V, L)$ with g additional edges $E^* = \{e_1^*, e_2^*, \dots, e_g^*\}$. (Note that, since any graph obtained from a spanning tree adding an edge contains a 1-cycle, the belted tree is no longer a tree.) These additional edges should have the following property: each one closes a non-bounding 1-cycle σ_n , and the set $\{\sigma_n\}_{n=1}^g$ represents a basis of $\mathcal{H}_1(\bar{\Omega}, \mathbb{Z})$. It is clear that, using these 1-cycles σ_n in (3.1b), this problem takes a simple form, as the degree of freedom corresponding to the edge e_n^* is equal to κ_n , being $\int_{e_n^*} \mathbf{Z}_h \cdot \boldsymbol{\tau} = \oint_{\sigma_n} \mathbf{Z}_h \cdot d\mathbf{s}$.

Let us note, however, that in three-dimensions the construction of a belted tree is not straightforward (see Rapetti et al. [53], Dłotko et al. [26], Dłotko and Specogna [21]).

Even if a belted tree inside Ω is not used, the homological equations (3.1b) can be simplified. In fact, in Section 6 we construct a spanning tree $\mathcal{S}_h^\partial = (V', L')$ and then a belted tree $\mathcal{B}_h^\partial = (V', L' \cup \{\epsilon_1^*, \dots, \epsilon_{2g}^*\})$ of the mesh induced on $\partial\Omega$ (since this is a two-dimensional problem, this construction is much easier than the analogous construction inside a three-dimensional domain). Let us denote by $\{\gamma_l\}_{l=1}^{2g}$ the 1-cycles in \mathcal{B}_h^∂ ; they represent a basis of $\mathcal{H}_1(\partial\Omega, \mathbb{Z})$. Then we extend \mathcal{S}_h^∂ to a spanning tree of the whole mesh. The 1-cycles σ_n can be expressed in terms of γ_l as $\sigma_n = \sum_{j=1}^{2g} A_{n,l} \gamma_l$, where $A_{n,l} \in \mathbb{Z}$. Hence, taking into account (3.1c),

$$\oint_{\sigma_n} \mathbf{Z}_h \cdot d\mathbf{s} = \sum_{l=1}^{2g} A_{n,l} \oint_{\gamma_l} \mathbf{Z}_h \cdot d\mathbf{s} = \sum_{l=1}^{2g} A_{n,l} \int_{\epsilon_l^*} \mathbf{Z}_h \cdot \boldsymbol{\tau} = \sum_{l=1}^{2g} A_{n,l} q(\epsilon_l^*),$$

and thus (3.1b) is reduced to this system of g equations with $2g$ unknowns:

$$\sum_{l=1}^{2g} A_{n,l} q(\epsilon_l^*) = \kappa_n, \quad \forall n = 1, 2, \dots, g. \quad (4.1)$$

Let us come now to equations (3.1a). Since the degrees of freedom of a function $\mathbf{v}_h \in RT_h$ are the face fluxes $\int_f \mathbf{v}_h \cdot \boldsymbol{\nu}$ for $f \in F$, relations (3.1a) are in fact a linear system with n_f (number of faces in \mathcal{T}_h) equations and n_e (number of edges in \mathcal{T}_h) unknowns. For each face $f \in F$ we have

$$\int_f \mathbf{J}_h \cdot \boldsymbol{\nu} = \int_f \operatorname{curl} \mathbf{Z}_h \cdot \boldsymbol{\nu} = \oint_{\partial_2 f} \mathbf{Z}_h \cdot d\mathbf{s} = \sum_{e \in E\{f\}} o_f(e) q(e).$$

Thus, each row in the linear system (3.1a) has exactly three non-zero entries.

System (3.1) can be finally rewritten as

$$\sum_{e \in E\{f\}} o_f(e) q(e) = \int_f \mathbf{J}_h \cdot \boldsymbol{\nu}, \quad \forall f \in F, \quad (4.2a)$$

$$\sum_{l=1}^{2g} A_{n,l} q(\epsilon_l^*) = \kappa_n, \quad \forall n = 1, 2, \dots, g \quad (4.2b)$$

$$q(e) = 0, \quad \forall e \in L. \quad (4.2c)$$

Webb and Forghani [62] have proposed a simple algorithm to solve this system. Here we rewrite their algorithm splitting it in two parts, *initialization* and *advancing*. In the initialization step we construct the sets \mathcal{F}_k , $k = 0, 1, 2, 3$, given by the faces having exactly k edges in K (the set of edges whose degree of freedom has been

computed using (3.1c) and, possibly, (3.1b)). On the other hand, advancing is quite natural. At any step, let us denote by D the set of edges whose value has been already computed; if there are faces with two edges in D , one computes from (4.2a) the degree of freedom corresponding to the third edge of such a face.

We can describe the initialization step in the following way ($|E\{f\} \cap A|$ denotes the number of edges belonging to the face f and to a certain set A).

Procedure WebbForghaniInit
<pre> 1 $\mathcal{F}_0 \leftarrow \emptyset$; $\mathcal{F}_1 \leftarrow \emptyset$; $\mathcal{F}_2 \leftarrow \emptyset$; $\mathcal{F}_3 \leftarrow \emptyset$; $D \leftarrow K$; 2 forall the $f \in F$ do $k \leftarrow E\{f\} \cap K$; $\mathcal{F}_k \leftarrow \mathcal{F}_k \cup \{f\}$; </pre>

Concerning the advancing step, we first need the description of the procedure that assigns to the edge e the computed degree of freedom v , and that updates the sets \mathcal{F}_k and D .

Procedure WebbForghaniSetEdge(e, v)
<pre> 1 forall the $f \in F\{e\}$ do $k \leftarrow E\{f\} \cap D$; $\mathcal{F}_k \leftarrow \mathcal{F}_k \setminus \{f\}$; $\mathcal{F}_{k+1} \leftarrow \mathcal{F}_{k+1} \cup \{f\}$; 2 $q(e) \leftarrow v$; $D \leftarrow D \cup \{e\}$; // assign value v to the edge e </pre>

Using this procedure, the advancing step reads

Procedure WebbForghaniAdvance
<pre> 1 while $\mathcal{F}_2 \neq \emptyset$ do 2 Let $f \in \mathcal{F}_2$ and $t \leftarrow \int_f \mathbf{J}_h \cdot \boldsymbol{\nu}$; 3 forall the $e \in E\{f\} \cap D$ do $t \leftarrow t - o_f(e)q(e)$; 4 $e \leftarrow E\{f\} \setminus D$; // the unassigned edge 5 WebbForghaniSetEdge($e, o_f(e)t$); 6 end while </pre>

In conclusion, the algorithm proposed by Webb and Forghani [62] is a simple call of WebbForghaniInit followed by WebbForghaniAdvance.

Algorithm 1: Advancing procedure of Webb–Forghani
<pre> 1 WebbForghaniInit; 2 WebbForghaniAdvance; </pre>

In the original algorithm of Webb and Forghani [62] the initialization step uses $K = L$, the set of edges of \mathcal{S}_h , namely, it takes information only from (3.1c). If a belted tree is available, it is possible to start with $K = L \cup E^*$, thus using both (3.1c) and (3.1b).

The algorithm stops when $\mathcal{F}_2 = \emptyset$ and it is successful if $D = E$ (or, equivalently, $\mathcal{F}_0 = \mathcal{F}_1 = \emptyset$, besides $\mathcal{F}_2 = \emptyset$). Two questions are in order. Does the algorithm start? Does the algorithm terminate with $D = E$?

If the spanning tree \mathcal{S}_h is constructed in a suitable way, for instance a breadth-first spanning tree, there exist faces in F with exactly two sides in L : therefore, in these cases, since $L \subset K$, the algorithm does start.

A careful analysis of the termination properties of this algorithm can be found in Dłotko and Specogna [21], where it is referred to as GSTT (generalized spanning tree technique) if the initialization set is $K = L \cup E^*$, or STT (spanning tree technique), if the initialization set is $K = L$. Clearly, if the domain Ω has a simple topological shape (namely, $g = 0$) the two algorithms coincide.

In [21] it is shown that the termination properties of GSTT and STT are strongly dependent on the choice of the spanning tree; for instance, with a depth-first spanning tree these algorithms can fail even if the domain Ω has a simple topological shape, and no choice of the spanning tree is known that allows the termination of the algorithms if the domain Ω is the complement of a trefoil knot.

Therefore, in a general topological situation it is not possible to say that these algorithms are able to determine all the degrees of freedom, and it is necessary to provide a strategy for the computation of the remaining unknowns.

To this aim, in the case of a domain of simple topological shape Dłotko and Specogna [24] proposed ESTT (extended spanning tree technique), based on symbolic computations.

The algorithm that we propose is valid for any topological situation and reads as follows: first, we assign the value 0 on all the edges of the spanning tree; then, without introducing a belted tree, we apply Algorithm 1 with the initial set $K = L$ (namely, STT). This procedure can stop without having determined all the degrees of freedom: in this case, we check if the homological equations (3.1b) permit to compute one or more unknowns, and we recall the Procedure WebbForghaniAdvance. When the homological equations are no longer able to give additional information, we adopt a *residual graph* approach, that has some similarities with ESTT and indeed shows to be very efficient.

Let us describe the procedure: we are left with some faces where only one degree of freedom has been determined (say, 1-faces in the set \mathcal{F}_1), and some faces where no degree of freedom has been determined (say, 0-faces in the set \mathcal{F}_0). We construct a graph based on the fact that a 1-face naturally connects its two non-assigned edges: in other words, in this graph the nodes are the non-assigned edges and the arcs are the 1-faces.

In general, this residual graph, that will be denoted by \mathcal{G}^R , is not connected. We construct a spanning tree on each connected component \mathcal{G}_s^R , $s = 1, 2, \dots, S$, namely, a spanning forest, and we choose a root for each spanning tree. This process is resumed in Algorithm 2.

Algorithm 2: Build a spanning forest of the residual graph

- 1 Set $s \leftarrow 0$ and build the graph $\mathcal{G}^R = (E \setminus D, \mathcal{F}_1)$ whose “vertices” are the unassigned edges and “edges” are the 1-faces which connect two unassigned edges;
- 2 **forall** the $f \in \mathcal{F}_0$ **do**
- 3 **forall** the $e \in E\{f\}$ not reachable for all \mathcal{T}_k^R with $k = 1, 2, \dots, s$ **do**
- 4 Set $s \leftarrow s + 1$ and build $\mathcal{T}_s^R = (E_s, F_s)$, the spanning tree in \mathcal{G}_s^R with root e ;
- 5 **end forall**
- 6 **end forall**

Each degree of freedom of a fixed connected component can be expressed in an affine way with respect to the degree of freedom of the corresponding root, i.e., if $q(e)$ is the degree of freedom of the edge e , then $q(e) = a_e q(e_r) + b_e$, where e_r is the root of the spanning tree of that connected component. The computation of the affine coefficients a_e and b_e is very fast and is described in function `propagateValue(e)`. There, we use the functions `predNode(e)` and `predEdge(e)`, which for each node on the tree return the corresponding parent node and the corresponding parent edge, respectively (remember that a node of the residual graph corresponds to an edge of the mesh, while an edge of the residual graph corresponds to a face of the mesh).

Function propagateValue(e)
<pre> 1 Let $\mathcal{T}_s^R = (E_s, F_s)$ the tree such that $e \in E_s$; $a \leftarrow 1$; $b \leftarrow 0$; 2 while e is not the root of \mathcal{T}_s^R do 3 $e' \leftarrow \text{predNode}(e)$; $f' \leftarrow \text{predEdge}(e)$; $e'' \leftarrow E\{f'\} \cap D$; 4 $a \leftarrow -o_{f'}(e) o_{f'}(e')$; $b \leftarrow o_{f'}(e)(\int_{f'} \mathbf{J}_h \cdot \boldsymbol{\nu} - o_{f'}(e'')q(e'') - o_{f'}(e')b)$; 5 $e \leftarrow e'$; 6 end while 7 return $[a, b, s]$; </pre>

The equations associated to the 0-faces are affine equations in terms of no more than three roots, and read as follows

$$\int_f \mathbf{J}_h \cdot \boldsymbol{\nu} = \sum_{e \in E\{f\}} o_f(e)q(e) = \sum_{e \in E\{f\}} o_f(e)(a_e q(e_r) + b_e).$$

The construction of the linear system associated to the 0-faces is quite easy using propagateValue(e) and is described in Algorithm 3.

Algorithm 3: Construction of the linear system associated to the 0-faces
<pre> 1 $\mathbf{Q} \leftarrow \mathbf{0}$; $\mathbf{c} \leftarrow \mathbf{0}$; $i \leftarrow 0$; 2 forall the $f \in \mathcal{F}_0$ do 3 $i \leftarrow i + 1$; $c_i \leftarrow \int_f \mathbf{J}_h \cdot \boldsymbol{\nu}$; 4 forall the $e \in E\{f\}$ do 5 $[a, b, s] \leftarrow \text{propagateValue}(e)$; $c_i \leftarrow c_i - b o_f(e)$; $Q_{is} \leftarrow a o_f(e)$; 6 end forall 7 end forall </pre>

Clearly, also the homological equations (4.2b) can be expressed in terms of the unknowns corresponding to the roots. Doing this, the linear system associated to the homological equations (4.2b) is built as described in Algorithm 4.

Algorithm 4: Construction of the final linear system
<pre> 1 Build the linear system with Algorithm 3; 2 $i \leftarrow \mathcal{F}_0$; // the number of equations already inserted 3 for $n = 1, 2, \dots, g$ do 4 $i \leftarrow i + 1$; $c_i \leftarrow \kappa_n$; 5 for $j = 1, 2, \dots, 2g$ do 6 $[a, b, s] \leftarrow \text{propagateValue}(\epsilon_j^*)$; $c_i \leftarrow c_i - b A_{n,j}$; $Q_{is} \leftarrow a A_{n,j}$; 7 end for 8 end for </pre>

In conclusion, we have thus reduced the problem to the solution of a small and sparse linear system with as many unknowns as the number of connected components of the residual graph (and as many equations as the number of 0-faces plus g).

The solution algorithm is described in Algorithm 5.

Algorithm 5: Extended Webb–Forghani

```

1 WebbForghaniInit;
2 WebbForghaniAdvance;
3 while  $D \neq E$  and  $q(e)$  for some  $e \notin D$  can be deduced using (4.2b) do
4   |   WebbForghaniSetEdge( $e, q(e)$ );
5   |   WebbForghaniAdvance;
6 end while
7 if  $D \neq E$  then
8   |   Build the residual graph and the spanning forest with Algorithm 2, build
      |   the reduced linear system with Algorithms 3 and 4, solve the reduced
      |   linear system using a direct method;
9 end if

```

Note that, in the numerical experiments reported in Section 7, if the domain Ω is the complement of a non-knotted domain Algorithm 5 terminates at line 6. In the remaining examples, the number of the unknowns of the reduced system is extremely small (see Table 7.2).

Lines 3–6 in Algorithm 5 are an optimization strategy that is convenient to employ because in many cases avoids the construction of the residual graph.

REMARK 2. *Let us add a comment about the way of verifying if one can compute a degree of freedom $q(e)$ from (4.2b) as requested at line 3 in the algorithm above.*

The simplest strategy is to check if (4.2b) (that eventually has been reduced by the elimination of some degrees of freedom due to the advancing of Webb–Forghani algorithm) has a row with only one unknown left.

Another possibility is to verify this situation after having applied Gauss–Jordan elimination. However, this depends on the choice of the pivoting. In order to be sure that a degree of freedom can be computed, one should check all the possible row and column permutations, but, since this procedure is too costly, our recipe is simply to use a single shoot of Gauss–Jordan elimination.

5. An explicit formula for the loop fields. When one is interested in the computation of the loop fields, namely, $\mathbf{J}_h = \mathbf{0}$ in (3.1a), a different procedure can be employed. In fact, when $\mathbf{J}_h = \mathbf{0}$ each degree of freedom of the solution to (3.1) can be expressed by an explicit formula in terms of linking numbers.

The computation of a linking number can be done efficiently by means of an exact and explicit formula written in terms of simple double integrals (see Bertolazzi and Ghiloni [11]); however, for a fine mesh the number of edges is quite large, hence this formula turns out to be too expensive if used for all the edges in $E \setminus L$. The recipe we adopt is the following: in Algorithm 5 replace line 8 by the computation of the value of one single unknown using this explicit formula (see Algorithm 6). In the numerical experiments presented in Section 7 we show that the use of the explicit formula is necessary very few times.

We recall that the linking number is an integer that, given two closed and disjoint curves in the three-dimensional space, represents the number of times that each curve winds around the other (see, e.g., Rolfsen [56, pp. 132–136]). We use this concept in a slightly different case, that is natural in the homological framework (see Seifert and Threlfall [58, Sects. 70, 73, 77]). We consider a 1-cycle γ in \mathbb{R}^3 of the form $\gamma = \sum_{i=1}^{n_e} \alpha_i [v_{e_i}^-, v_{e_i}^+]$, where $\alpha_i \in \mathbb{Z}$, $[v_{e_i}^-, v_{e_i}^+]$ is the oriented edge e_i , and the boundary $\partial_1 \gamma = \sum_{i=1}^{n_e} \alpha_i (v_{e_i}^+ - v_{e_i}^-)$ is null. The support of γ is the union of the segments joining

$v_{e_i}^-$ and $v_{e_i}^+$ for which $\alpha_i \neq 0$. For a continuous vector field \mathbf{U} we set

$$\oint_{\gamma} \mathbf{U} \cdot d\mathbf{s} = \sum_{i=1}^{n_e} \alpha_i \int_{[v_{e_i}^-, v_{e_i}^+]} \mathbf{U} \cdot \boldsymbol{\tau}, \quad \oint_{\gamma} \mathbf{U} \times d\mathbf{s} = \sum_{i=1}^{n_e} \alpha_i \int_{[v_{e_i}^-, v_{e_i}^+]} \mathbf{U} \times \boldsymbol{\tau},$$

where, as before, $\boldsymbol{\tau}$ is the unit tangent vector of $[v_{e_i}^-, v_{e_i}^+]$. The linking number is a double line integral.

DEFINITION 1. Given $\gamma = \sum_{i=1}^{n_e} \alpha_i [v_{e_i}^-, v_{e_i}^+]$ and $\gamma' = \sum_{j=1}^{n_e} \beta_j [v_{e_j}^-, v_{e_j}^+]$, two 1-cycles in \mathbb{R}^3 with disjoint supports, we define their linking number by

$$\begin{aligned} \ell_{\kappa}(\gamma, \gamma') &:= \frac{1}{4\pi} \oint_{\gamma} \oint_{\gamma'} \frac{\mathbf{x}-\mathbf{y}}{|\mathbf{x}-\mathbf{y}|^3} \cdot d\mathbf{s}(\mathbf{x}) \times d\mathbf{s}(\mathbf{y}) \\ &= \frac{1}{4\pi} \oint_{\gamma} \left(\oint_{\gamma'} \frac{\mathbf{y}-\mathbf{x}}{|\mathbf{y}-\mathbf{x}|^3} \times d\mathbf{s}(\mathbf{y}) \right) \cdot d\mathbf{s}(\mathbf{x}). \end{aligned} \quad (5.1)$$

Let us observe that $\ell_{\kappa}(\gamma, \gamma') = \frac{1}{4\pi} \sum_{i,j=1}^{n_e} \alpha_i \beta_j L_{ij}$, where

$$L_{ij} = \int_{[v_{e_i}^-, v_{e_i}^+]} \left(\int_{[v_{e_j}^-, v_{e_j}^+]} \frac{\mathbf{y}-\mathbf{x}}{|\mathbf{y}-\mathbf{x}|^3} \times \boldsymbol{\tau}(\mathbf{y}) \right) \cdot \boldsymbol{\tau}(\mathbf{x}).$$

Therefore, in order to compute the integer $\ell_{\kappa}(\gamma, \gamma')$, it suffices to have an efficient way for evaluating the double integrals L_{ij} .

We also need the following definitions. For each 1-cycle σ of $\partial\Omega$, we denote by $[\sigma]^+$ its homology class in Ω and by $[\sigma]^-$ its homology class in $\mathbb{R}^3 \setminus \Omega$. It is clear that $[\sigma]^+$ and $[\sigma]^-$ depend only on $[\sigma]$, and not on its representative σ in $\partial\Omega$.

Now we describe how a basis of loop fields can be explicitly computed by choosing in (3.1) the constants κ_n equal to suitable linking numbers. Recall that $\mathcal{S}_h = (V, L)$ is a spanning tree of the graph (V, E) , with $\mathcal{T}_h = (V, E, F, T)$; we can also think that v_1 is its root. Given $v_j \in V$, let C_{v_j} be the unique 1-chain in \mathcal{S}_h from v_1 to v_j (namely, the coefficients in C_{v_j} are 0, 1, -1 and $\partial_1 C_{v_j} = v_j - v_1$). If $e \in E$ is the edge with end points v_e^- and v_e^+ , we denote by D_e the 1-cycle of \mathcal{T}_h given by $D_e := C_{v_e^-} + e - C_{v_e^+}$ (note that, if $e \in L$, $C_{v_e^+} = C_{v_e^-} + e$, hence $D_e = 0$).

THEOREM 4. Let $\hat{\gamma}$ be a 1-cycle of $\mathbb{R}^3 \setminus \bar{\Omega}$ and let $\boldsymbol{\eta}_{h, \hat{\gamma}} \in N_h$ be given by

$$\boldsymbol{\eta}_{h, \hat{\gamma}} = \sum_{k=1}^{n_e} q(e_k) \mathbf{w}_{h, k}, \quad (5.2)$$

where $q(e_k) = \ell_{\kappa}(D_{e_k}, \hat{\gamma})$ and $\mathbf{w}_{h, k}$ are the basis function of N_h . The following assertions hold:

- (i) $\oint_{\gamma} \boldsymbol{\eta}_{h, \hat{\gamma}} \cdot d\mathbf{s} = \ell_{\kappa}(\gamma, \hat{\gamma})$ for each 1-cycle γ of \mathcal{T}_h ,
- (ii) $\text{curl } \boldsymbol{\eta}_{h, \hat{\gamma}} = \mathbf{0}$.

Proof. Let $\gamma = \sum_{j=1}^{n_e} \alpha_j e_j$ be a 1-cycle of \mathcal{T}_h , with $\alpha_j \in \mathbb{Z}$; being a 1-cycle, its boundary is null, hence

$$0 = \partial_1 \gamma = \sum_{j=1}^{n_e} \alpha_j (v_{e_j}^+ - v_{e_j}^-).$$

Since $C_{v_{e_j}^-}$ and $C_{v_{e_j}^+}$ depend only on $v_{e_j}^-$ and $v_{e_j}^+$, respectively, we also obtain

$$\sum_{j=1}^{n_e} \alpha_j D_{e_j} = \sum_{j=1}^{n_e} \alpha_j (C_{v_{e_j}^-} + e_j - C_{v_{e_j}^+}) = \gamma - \sum_{j=1}^{n_e} \alpha_j (C_{v_{e_j}^+} - C_{v_{e_j}^-}) = \gamma. \quad (5.3)$$

Now we can compute

$$\oint_{\gamma} \boldsymbol{\eta}_{h,\hat{\gamma}} \cdot d\mathbf{s} = \sum_{j=1}^{n_e} \alpha_j \int_{e_j} \boldsymbol{\eta}_{h,\hat{\gamma}} \cdot \boldsymbol{\tau} = \sum_{j=1}^{n_e} \sum_{k=1}^{n_e} \alpha_j q(e_k) \int_{e_j} \mathbf{w}_{h,k} \cdot \boldsymbol{\tau}.$$

Using that $\int_{e_j} \mathbf{w}_{h,k} \cdot \boldsymbol{\tau} = \delta_{k,j}$ and equation (5.3) one has

$$\oint_{\gamma} \boldsymbol{\eta}_{h,\hat{\gamma}} \cdot d\mathbf{s} = \sum_{j=1}^{n_e} \alpha_j q(e_j) = \sum_{j=1}^{n_e} \alpha_j \ell_{\kappa}(D_{e_j}, \hat{\gamma}) = \ell_{\kappa}\left(\sum_{j=1}^{n_e} \alpha_j D_{e_j}, \hat{\gamma}\right) = \ell_{\kappa}(\gamma, \hat{\gamma}),$$

and we have proved (i). Concerning (ii), for each $f \in F$ we have

$$\int_f \operatorname{curl} \boldsymbol{\eta}_{h,\hat{\gamma}} \cdot \mathbf{n} = \oint_{\partial_2 f} \boldsymbol{\eta}_{h,\hat{\gamma}} \cdot d\mathbf{s} = \ell_{\kappa}(\partial_2 f, \hat{\gamma}) = 0.$$

Since $\operatorname{curl} \boldsymbol{\eta}_{h,\hat{\gamma}}$ belongs to the space of Raviart–Thomas finite elements of degree 1, this means that all its degrees of freedom are vanishing, so it is equal to $\mathbf{0}$. \square

Let us now assume that we know a set of 1-cycles $\{\sigma_n\}_{n=1}^g \cup \{\hat{\sigma}_n\}_{n=1}^g$ of $\partial\Omega$ such that: they are representatives of a basis of the homology group of $\partial\Omega$; $\{\sigma_n\}_{n=1}^g$ (respectively, $\{\hat{\sigma}_n\}_{n=1}^g$) represent a basis of the homology group $\mathcal{H}_1(\bar{\Omega}, \mathbb{Z})$ (respectively, of the homology group $\mathcal{H}_1(\mathbb{R}^3 \setminus \Omega, \mathbb{Z})$). The construction of these two sets of 1-cycles is faced in the next section.

We recall that we have denoted by $[\sigma_j]^+$ the homology class of σ_j in $\bar{\Omega}$ and by $[\hat{\sigma}_j]^-$ the homology class of $\hat{\sigma}_j$ in $\mathbb{R}^3 \setminus \Omega$. Here below we also introduce the 1-cycle $R^+ \sigma_j$, a representative of $[\sigma_j]^+$ whose support is completely contained in Ω , and the 1-cycle $R^- \hat{\sigma}_j$, a representative of $[\hat{\sigma}_j]^-$ whose support is completely contained in $\mathbb{R}^3 \setminus \bar{\Omega}$. $R^+ \sigma_j$ can be obtained by slightly “retracting” σ_j inside Ω , and similarly $R^- \hat{\sigma}_j$ by slightly “retracting” $\hat{\sigma}_j$ inside $\mathbb{R}^3 \setminus \bar{\Omega}$.

THEOREM 5. *The vector fields*

$$\mathbf{T}_{0,j} := \boldsymbol{\eta}_{h,R^- \hat{\sigma}_j}, \quad j = 1, 2, \dots, g,$$

are a set of linearly cohomologically independent finite element loop fields, hence they represent a finite element basis of the first de Rham cohomology group of Ω .

Proof. The vector fields $\mathbf{T}_{0,j}$ are the solutions of problem (3.1) with $\mathbf{J}_h = \mathbf{0}$ and $\kappa_n = \ell_{\kappa}(\sigma_n, R^- \hat{\sigma}_j)$. Since the matrix \mathbf{M} with entries $M_{i,j} = \ell_{\kappa}(\sigma_n, R^- \hat{\sigma}_j)$ is non-singular by the Alexander duality theorem (see Munkres [48, Sect. 71] and Seifert and Threlfall [58, point 47, p. 337]), they are linearly cohomologically independent (see Theorem 3). \square

REMARK 3. *The vector fields $\mathbf{T}_{0,j} = \boldsymbol{\eta}_{h,R^- \hat{\sigma}_j}$ just defined depend only on the homology class $[\hat{\sigma}_j]^-$ of $\hat{\sigma}_j$ in $\mathbb{R}^3 \setminus \Omega$ and not on the chosen representative $R^- \hat{\sigma}_j$ with support in $\mathbb{R}^3 \setminus \bar{\Omega}$. In fact, if $\hat{\sigma}'_j$ is another representative of $[\hat{\sigma}_j]^-$ with support in*

$\mathbb{R}^3 \setminus \overline{\Omega}$, then $R^- \hat{\sigma}_j$ and $\hat{\sigma}'_j$ are homologous not only in $\mathbb{R}^3 \setminus \Omega$ but also in $\mathbb{R}^3 \setminus \overline{\Omega}$. This is a consequence of the fact that $\partial\Omega$ is a locally flat surface of \mathbb{R}^3 , hence it has a bicollar in \mathbb{R}^3 . Now, if γ is a 1-cycle in \mathbb{R}^3 with support in $\overline{\Omega}$, then $\ell_\kappa(\gamma, R^- \hat{\sigma}_j) = \ell_\kappa(\gamma, \hat{\sigma}'_j)$ (see Seifert and Threlfall [58, p. 289], Rolfsen [56, p. 133]).

We also observe that $\ell_\kappa(\gamma, R^- \hat{\sigma}_j) = \ell_\kappa(\gamma', \hat{\sigma}_j)$, if γ' is any 1-cycle of \mathbb{R}^3 with support in Ω and homologous to γ in $\overline{\Omega}$. In particular, if the support of γ is contained in $\partial\Omega$, we can write $\ell_\kappa(\gamma, R^- \hat{\sigma}_j) = \ell_\kappa(R^+ \gamma, \hat{\sigma}_j)$. This can be useful for implementation, as in this way one avoids to go outside the computational domain $\overline{\Omega}$.

REMARK 4. A physical interpretation of the explicit formula (5.2) is the following. The Biot–Savart law gives the magnetic field generated by a unitary density current concentrated along the cycle $R^- \hat{\sigma}_j$ by means of the formula:

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

Since the cycle $R^- \hat{\sigma}_j$ is external to $\overline{\Omega}$, one has $\text{curl } \widehat{\mathbf{H}} = \mathbf{0}$ in Ω . Moreover, on each cycle $\gamma \subset \overline{\Omega}$ that is linking the current passing in $R^- \hat{\sigma}_j$ 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, at least one of the generators σ_n of the first homology group of $\overline{\Omega}$.)

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

$$\widehat{q}_e = \frac{1}{4\pi} \int_e \left(\oint_{R^- \hat{\sigma}_j} \frac{\mathbf{y} - \mathbf{x}}{|\mathbf{y} - \mathbf{x}|^3} \times d\mathbf{s}(\mathbf{y}) \right) \cdot \boldsymbol{\tau}(\mathbf{x}) \quad (5.4)$$

(and this visibly resembles the formula for computing the linking number between $R^- \hat{\sigma}_j$ and another disjoint cycle).

Introduce now the spanning tree L , with its root v_1 , and define the scalar function $\phi_h \in L_h$ in all the vertices of \mathcal{T}_h as $\phi_h(v_1) = 0$ and

$$\phi_h(v_{e'}^+) - \phi_h(v_{e'}^-) = \widehat{q}_{e'}, \quad \forall e' \in L.$$

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

Finally consider the 1-cycle $D_e = C_{v_e^-} + e - C_{v_e^+}$ defined before and its support Y_e . We already know that $D_{e'} = 0$ if $e' \in L$, while when $e \in E \setminus L$ the 1-cycle D_e is constituted by edges all belonging to the spanning tree (except e). For $e \notin L$ and $e' \in Y_e \setminus \{e\}$ define $o(e') = \pm 1$, where the sign is positive if the orientation of e' is the same of the path D_e and negative otherwise. By using (5.4) it is straightforward to verify that

$$\begin{aligned} \frac{1}{4\pi} \oint_{D_e} \left(\oint_{R^- \hat{\sigma}_j} \frac{\mathbf{y} - \mathbf{x}}{|\mathbf{y} - \mathbf{x}|^3} \times d\mathbf{s}(\mathbf{y}) \right) \cdot d\mathbf{s}(\mathbf{x}) &= \widehat{q}_e + \sum_{e' \in Y_e \setminus \{e\}} o(e') \widehat{q}_{e'} \\ &= \widehat{q}_e + \sum_{e' \in Y_e \setminus \{e\}} o(e') (\phi_h(v_{e'}^+) - \phi_h(v_{e'}^-)) \\ &= \widehat{q}_e + \phi_h(v_e^-) - \phi_h(v_e^+) = \int_e \Pi^{N_h} \widehat{\mathbf{H}} \cdot \boldsymbol{\tau} - \int_e \text{grad } \phi_h \cdot \boldsymbol{\tau} \\ &= \int_e \mathbf{Z}_h \cdot \boldsymbol{\tau}, \end{aligned} \quad (5.5)$$

and thus the degrees of freedom of \mathbf{Z}_h are given by $\ell_\kappa(D_e, R^- \hat{\sigma}_j)$.

As we already noted, formula (5.2) can be too expensive if used for all the edges of the mesh \mathcal{T}_h . We use it as an alternative to the construction and solution of the reduced system in Algorithm 5, as illustrated in the following algorithm.

Algorithm 6: Alternative extended Webb–Forghani for the loop fields $\mathbf{T}_{0,j}$

```

1 WebbForghaniInit;
2 WebbForghaniAdvance;
3 while  $D \neq E$  and  $q(e)$  for some  $e \in E \setminus D$  can be deduced using (4.2b) do
4   |   WebbForghaniSetEdge( $e, q(e)$ );
5   |   WebbForghaniAdvance;
6 end while
7 while  $D \neq E$  do
8   |   Let  $e \in E \setminus D$ , compute  $q(e) = \ell_{\kappa}(D_e, R^{-}\hat{\sigma}_j)$ ;
9   |   WebbForghaniSetEdge( $e, q(e)$ );
10  |   WebbForghaniAdvance;
11 end while

```

As we noticed at the end of the previous section, in the numerical experiments reported in Section 7 this algorithm terminates at line 6 if the domain Ω is the complement of a non-knotted domain. In the remaining examples the number of linking numbers that have to be computed for concluding the procedure is very small (say, between one and four).

6. The construction of the homology bases. The aim of this section is to present an algorithm for computing simultaneously a basis of $\mathcal{H}_1(\bar{\Omega}, \mathbb{Z})$ and a basis of $\mathcal{H}_1(\mathbb{R}^3 \setminus \Omega, \mathbb{Z})$. We will mainly follow the construction proposed in Hiptmair and Ostrowski [36], with some modifications in order to assure that the obtained 1-cycles have integer coefficients and to cover the case in which the boundary $\partial\Omega$ is not connected.

First we recall two theoretical results. The Alexander duality theorem, applied to $\bar{\Omega}$, asserts that the abelian groups $\mathcal{H}_1(\bar{\Omega}, \mathbb{Z})$ and $\mathcal{H}_1(\mathbb{R}^3 \setminus \Omega, \mathbb{Z})$ are isomorphic. Indeed, they are both isomorphic to \mathbb{Z}^g .

Consider a 1-cycle σ of $\partial\Omega$ (clearly, it is also a 1-cycle in $\bar{\Omega}$ and in $\mathbb{R}^3 \setminus \Omega$). We can define the following homomorphism $\varphi : \mathcal{H}_1(\partial\Omega, \mathbb{Z}) \rightarrow \mathcal{H}_1(\bar{\Omega}, \mathbb{Z}) \oplus \mathcal{H}_1(\mathbb{R}^3 \setminus \Omega, \mathbb{Z})$ by setting

$$\varphi([\sigma]) := ([\sigma]^+, [\sigma]^-). \quad (6.1)$$

The Mayer–Vietoris exact sequence associated with the splitting $\mathbb{R}^3 = \bar{\Omega} \cup (\mathbb{R}^3 \setminus \Omega)$ ensures that φ is an isomorphism. It follows that $\mathcal{H}_1(\partial\Omega, \mathbb{Z})$ is isomorphic to \mathbb{Z}^{2g} . We refer the reader to Cantarella et al. [15, Sect. 6] for a friendly description of the isomorphism φ . This isomorphism is important because it permits to build the basis cycles $\{\sigma_n\}_{n=1}^g$ of the group $\mathcal{H}_1(\bar{\Omega}, \mathbb{Z})$ working only on $\partial\Omega$ and not in the whole $\bar{\Omega}$.

Recalling that $\bar{\Omega}$ is equipped with a tetrahedral triangulation $\mathcal{T}_h = (V, E, F, T)$, the algorithm reads as follows.

Let us assume for a while that $\partial\Omega$ is connected. Let (V', E', F') be the triangulation of $\partial\Omega$ induced by \mathcal{T}_h , let (V', L') be a spanning tree of the graph (V', E') and let v' be a vertex in V' . For every $v \in V'$ and for every $\epsilon = [v, w] \in E'$, we denote by C'_v the unique 1-chain in (V', L') from v' to v and by D'_ϵ the 1-cycle $C'_v + \epsilon - C'_w$ of $\partial\Omega$. In [36, Sect. 3], Hiptmair and Ostrowski describe how to find $2g$ edges $\epsilon_1^*, \dots, \epsilon_{2g}^*$ in

$E' \setminus L'$ in such a way that the 1-cycles $D'_{\epsilon_1^*}, \dots, D'_{\epsilon_{2g}^*}$ represent a basis of $\mathcal{H}_1(\partial\Omega, \mathbb{Z})$. For simplicity, define $\gamma_l := D'_{\epsilon_l^*}$ for every $l = 1, 2, \dots, 2g$.

Thanks to the connectedness of $\partial\Omega$, it is possible to prove that, given any 1-cycle σ of $\partial\Omega$, one has $[\sigma]^- = [0]$, that is, σ bounds in $\mathbb{R}^3 \setminus \Omega$, if and only if $\ell_\kappa(\sigma, R^+\gamma_k) = 0$ for all $k = 1, 2, \dots, 2g$ (see Hiptmair and Ostrowski [36, Corol. 4.4]). Similarly, we have that $[\sigma]^+ = [0]$, that is, σ bounds in $\bar{\Omega}$, if and only if $\ell_\kappa(\sigma, R^-\gamma_k) = 0$ for all $k = 1, 2, \dots, 2g$.

Since $\{\gamma_l\}_{l=1}^{2g}$ represents a basis of $\mathcal{H}_1(\partial\Omega, \mathbb{Z})$, there exist, and are unique, $2g$ integers c_1, \dots, c_{2g} such that $[\sigma] = \sum_{l=1}^{2g} c_l [\gamma_l]$ in $\mathcal{H}_1(\partial\Omega, \mathbb{Z})$ (namely, σ and $\sum_{l=1}^{2g} c_l \gamma_l$ are homologous in $\partial\Omega$).

Denote by $\mathbf{G} = (G_{l,k}) \in \mathbb{Z}^{2g \times 2g}$ the matrix with entries $G_{l,k} := \ell_\kappa(\gamma_l, R^+\gamma_k)$ and by \mathbf{c} the vector (c_1, \dots, c_{2g}) . Since $\ell_\kappa(\sigma, R^+\gamma_k) = \sum_{l=1}^{2g} c_l \ell_\kappa(\gamma_l, R^+\gamma_k)$ for all $k = 1, 2, \dots, 2g$, it follows that $[\sigma]^- = [0]$ if and only if $\mathbf{G}^T \mathbf{c} = \mathbf{0}$, namely, $\mathbf{c} \in \text{Ker } \mathbf{G}^T$.

Similarly, $[\sigma]^+ = [0]$ if and only if $\sum_{l=1}^{2g} c_l \ell_\kappa(\gamma_l, R^-\gamma_k) = 0$ for all $k = 1, 2, \dots, 2g$. Bearing in mind Remark 3 and that the linking number is symmetric with respect to its components (see Rolfsen [56, p. 135]), we have that $\ell_\kappa(\gamma_l, R^-\gamma_k) = \ell_\kappa(R^+\gamma_l, \gamma_k) = \ell_\kappa(\gamma_k, R^+\gamma_l)$ and hence $[\sigma]^+ = [0]$ if and only if $\mathbf{G}\mathbf{c} = \mathbf{0}$, namely, $\mathbf{c} \in \text{Ker } \mathbf{G}$.

The existence of the isomorphism φ , defined in equation (6.1), ensures that $\text{Ker } \mathbf{G}^T$ and $\text{Ker } \mathbf{G}$ are isomorphic to $\mathcal{H}_1(\bar{\Omega}, \mathbb{Z})$ and to $\mathcal{H}_1(\mathbb{R}^3 \setminus \Omega, \mathbb{Z})$, respectively. In particular, the rank of \mathbf{G} (and hence of \mathbf{G}^T) is equal to g (see also Hiptmair and Ostrowski [36, Theorem 4.5]).

Let us perform the reduction of \mathbf{G} to Smith normal form (see Munkres [48, Sect. 11]). We obtain two matrices $\mathbf{L}, \mathbf{R} \in \mathbb{Z}^{2g \times 2g}$, *non-singular* over $\mathbb{Z}^{2g \times 2g}$, namely, $|\det \mathbf{L}| = |\det \mathbf{R}| = 1$, and non-zero integers s_1, \dots, s_g such that s_j divides s_{j+1} for all $j = 1, 2, \dots, g-1$ and

$$\mathbf{LGR} = \left[\begin{array}{cc|c} s_1 & 0 & 0 \\ & \ddots & \vdots \\ 0 & s_g & 0 \\ \hline 0 & \cdots & 0 \end{array} \right].$$

Evidently, the last g columns of \mathbf{R} form a basis of $\text{Ker } \mathbf{G}$, while the last g columns of \mathbf{L}^T form a basis of $\text{Ker } \mathbf{G}^T$. Denote by $\mathbf{A} = (A_{n,l})$ and $\mathbf{B} = (B_{n,l})$ the matrices in $\mathbb{Z}^{g \times 2g}$ formed by the last g rows of \mathbf{L} and by the last g rows of \mathbf{R}^T , respectively.

Define the 1-cycles $\{\sigma_n\}_{n=1}^g \cup \{\hat{\sigma}_n\}_{n=1}^g$ of $\partial\Omega$ by setting

$$\sigma_n = \sum_{l=1}^{2g} A_{n,l} \gamma_l \quad \text{and} \quad \hat{\sigma}_n = \sum_{l=1}^{2g} B_{n,l} \gamma_l$$

for all $n = 1, 2, \dots, g$. By construction, $\{\sigma_n\}_{n=1}^g$ are representatives of a basis of $\mathcal{H}_1(\bar{\Omega}, \mathbb{Z})$ and $\{\hat{\sigma}_n\}_{n=1}^g$ of a basis of $\mathcal{H}_1(\mathbb{R}^3 \setminus \Omega, \mathbb{Z})$. Moreover, we have that $[\sigma_n]^- = [0]$ and $[\hat{\sigma}_n]^+ = [0]$ for all $n = 1, 2, \dots, g$.

It is worth noting that, by using the isomorphism φ again, one sees immediately that the 1-cycles $\{\sigma_n\}_{n=1}^g \cup \{\hat{\sigma}_n\}_{n=1}^g$ represent a basis of $\mathcal{H}_1(\partial\Omega, \mathbb{Z})$. In this way, the determinant of the $(2g \times 2g)$ -matrix $\mathbf{P} = [\mathbf{A}^T | \mathbf{B}^T]$ is ± 1 . In fact, \mathbf{P} is the matrix associated with the change of basis of $\mathcal{H}_1(\partial\Omega, \mathbb{Z})$, from the one represented by $\{\sigma_n\}_{n=1}^g \cup \{\hat{\sigma}_n\}_{n=1}^g$ to the one represented by $\{\gamma_l\}_{l=1}^{2g}$.

Let us pass now to the case in which $\partial\Omega$ is not connected, as frequently occurs in applications. As usual, we denote by $\{(\partial\Omega)_r\}_{r=0}^p$ the connected components of $\partial\Omega$, with $(\partial\Omega)_0$ the external one. Thanks to the Jordan separation theorem (see Munkres [48, Corol. 74.2]), for each $r = 0, 1, \dots, p$ the set $\mathbb{R}^3 \setminus (\partial\Omega)_r$ has exactly two connected components, both having $(\partial\Omega)_r$ as boundary. Denote by D_r the bounded connected component of $\mathbb{R}^3 \setminus (\partial\Omega)_r$ and by g_r the first Betti number of D_r . Clearly, it holds: $\overline{\Omega} = \overline{D_0} \setminus \bigcup_{r=1}^p D_r$ and $\mathbb{R}^3 \setminus \Omega = (\mathbb{R}^3 \setminus D_0) \cup (\bigcup_{r=1}^p \overline{D_r})$. Moreover, being $\mathcal{H}_1(\partial\Omega, \mathbb{Z})$ isomorphic to $\bigoplus_{r=0}^p \mathcal{H}_1((\partial\Omega)_r, \mathbb{Z})$, we have that $\sum_{r=0}^p 2g_r = 2g$, namely $\sum_{r=0}^p g_r = g$.

Let us apply to each $(\partial\Omega)_r$ the algorithm described above. For each $r = 0, 1, \dots, p$ we obtain a set of 1-cycles $\{\sigma_{r,s}\}_{s=1}^{g_r} \cup \{\widehat{\sigma}_{r,s}\}_{s=1}^{g_r}$ of $(\partial\Omega)_r$ representing a basis of $\mathcal{H}_1((\partial\Omega)_r, \mathbb{Z})$. Furthermore, we have that

$$\sigma_{r,s} \text{ bounds in } \mathbb{R}^3 \setminus D_r \text{ and } \widehat{\sigma}_{r,s} \text{ bounds in } \overline{D_r} \text{ for all } s = 1, 2, \dots, g_r. \quad (6.2)$$

The next theorem completes the algorithm, giving a basis of $\mathcal{H}_1(\overline{\Omega}, \mathbb{Z})$ and a basis of $\mathcal{H}_1(\mathbb{R}^3 \setminus \Omega, \mathbb{Z})$. To prove such a result, we essentially follow the arguments used in the proof of Theorem 3.2.2.1 of Dey and Guha [19].

THEOREM 6. *The following assertions hold:*

- (i) *The set of 1-cycles $\{\sigma_{0,s}\}_{s=1}^{g_0} \cup \{\widehat{\sigma}_{1,s}\}_{s=1}^{g_1} \cup \dots \cup \{\widehat{\sigma}_{p,s}\}_{s=1}^{g_p}$ of $\partial\Omega$ are representatives of a basis of $\mathcal{H}_1(\overline{\Omega}, \mathbb{Z})$.*
- (ii) *The set of 1-cycles $\{\widehat{\sigma}_{0,s}\}_{s=1}^{g_0} \cup \{\sigma_{1,s}\}_{s=1}^{g_1} \cup \dots \cup \{\sigma_{p,s}\}_{s=1}^{g_p}$ of $\partial\Omega$ are representatives of a basis of $\mathcal{H}_1(\mathbb{R}^3 \setminus \Omega, \mathbb{Z})$.*

Proof. Since $\mathbb{R}^3 \setminus D_0, \overline{D_1}, \dots, \overline{D_p}$ are the connected components of $\mathbb{R}^3 \setminus \Omega$, we have that $\mathcal{H}_1(\mathbb{R}^3 \setminus \Omega, \mathbb{Z})$ is isomorphic to $\mathcal{H}_1(\mathbb{R}^3 \setminus D_0, \mathbb{Z}) \oplus \mathcal{H}_1(\overline{D_1}, \mathbb{Z}) \oplus \dots \oplus \mathcal{H}_1(\overline{D_p}, \mathbb{Z})$. Point (ii) follows immediately from this fact. Let us prove (i). Define the g -uples S_1 and S_2 of 1-cycles of $\partial\Omega$ by setting

$$\begin{aligned} S_1 &:= (\sigma_{0,1}, \dots, \sigma_{0,g_0}, \widehat{\sigma}_{1,1}, \dots, \widehat{\sigma}_{1,g_1}, \dots, \widehat{\sigma}_{p,1}, \dots, \widehat{\sigma}_{p,g_p}), \\ S_2 &:= (\widehat{\sigma}_{0,1}, \dots, \widehat{\sigma}_{0,g_0}, \sigma_{1,1}, \dots, \sigma_{1,g_1}, \dots, \sigma_{p,1}, \dots, \sigma_{p,g_p}). \end{aligned}$$

The 2 g -uple (S_1, S_2) represents a (ordered) basis B of $\mathcal{H}_1(\partial\Omega, \mathbb{Z})$ and, by (ii), S_2 represents a basis B_2 of $\mathcal{H}_1(\mathbb{R}^3 \setminus \Omega, \mathbb{Z})$. Let C_1 be a fixed basis of $\mathcal{H}_1(\overline{\Omega}, \mathbb{Z})$ and let Φ be the matrix associated with the isomorphism φ defined in (6.1), with respect to the bases B of $\mathcal{H}_1(\partial\Omega, \mathbb{Z})$ and (C_1, B_2) of $\mathcal{H}_1(\overline{\Omega}, \mathbb{Z}) \oplus \mathcal{H}_1(\mathbb{R}^3 \setminus \Omega, \mathbb{Z})$. Clearly, it has the following form:

$$\Phi = \left[\begin{array}{c|c} \mathbf{X} & \mathbf{Y} \\ \hline \mathbf{Z} & \mathbf{I}_g \end{array} \right],$$

where $\mathbf{X}, \mathbf{Y}, \mathbf{Z} \in \mathbb{Z}^{g \times g}$ and \mathbf{I}_g is the $(g \times g)$ -identity matrix. Thanks to (6.2), we infer at once that $\mathbf{Z} = \mathbf{0}$. Since φ is an isomorphism, we know that $|\det \Phi| = 1$. It follows that $|\det \mathbf{X}| = 1$ as well, and hence S_1 represents a basis of $\mathcal{H}_1(\overline{\Omega}, \mathbb{Z})$. This proves (i). \square

7. Numerical results. In this section we present some numerical experiments with the aim of illustrating the efficiency of the proposed methods. The algorithms have been developed using the LEMON library [20], which provides an efficient implementation of data structures and algorithms for graphs and networks.

We will consider six different test problems. The domain Ω is the complement in a box of: a 2-torus (Test A); the Borromean rings (Test B); a link constituted by two skeletons of a cube (topologically, two 5-tori) (Test C); a trefoil knot (Test D); the knot listed as 4_1 in Rolfsen [56, pp. 391]) (Test E); a link constructed with two knotted 4_1 -knots (Test F) (see Figure 7.1).

In all computations, except when explicitly stated, we use a spanning tree constructed via breadth-first search.

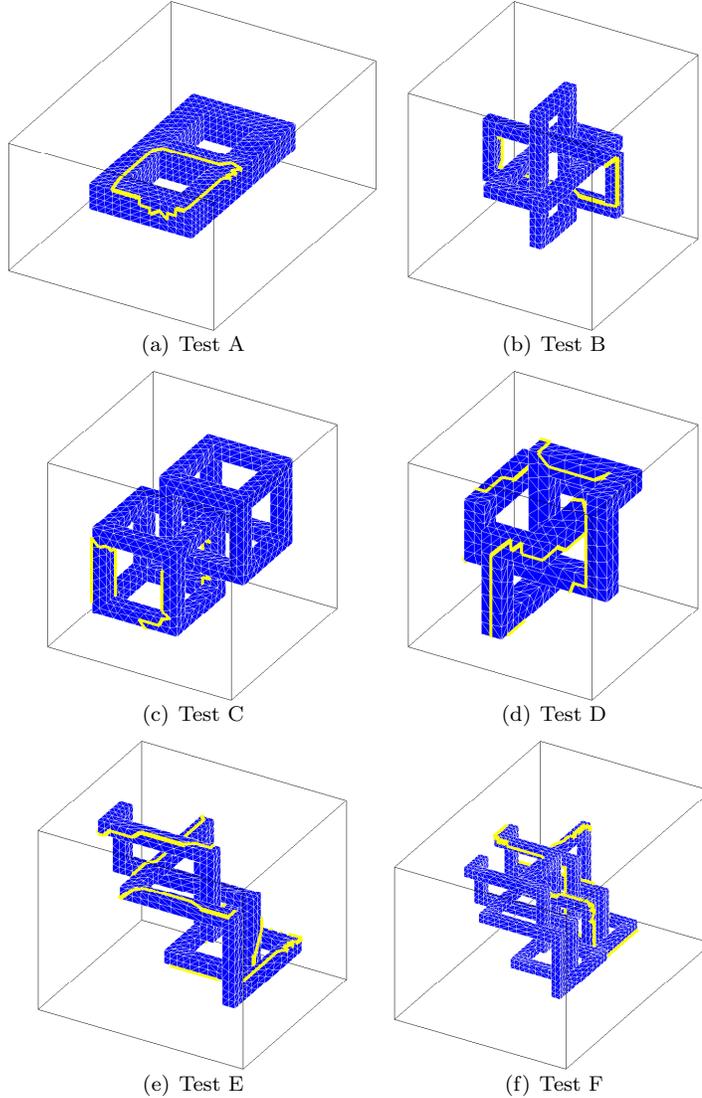


FIG. 7.1. The domain Ω (the complement in the box of the blue set). One homological 1-cycle $\hat{\sigma}_j$ (yellow color) is also reported.

A preliminary step for computing loop and source fields is the construction of the bases of the homology groups $\mathcal{H}_1(\bar{\Omega}, \mathbb{Z})$ and $\mathcal{H}_1(\mathbb{R}^3 \setminus \Omega, \mathbb{Z})$. Table 7.1 shows the CPU time required for the computation of all the homological cycles σ_n and $\hat{\sigma}_n$ for

three different meshes, following the construction described in Section 6. Here we recall that a different mesh is only furnishing a different basis of the homology group (as well as of the cohomology group), but we are interested in the construction of a basis of $N_h \cap H^0(\text{curl}; \Omega)$ and in the computation of the source fields, whose accuracy clearly depends on the mesh size.

TABLE 7.1
CPU time for computing all the homological 1-cycles.

	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 7.2 shows how the number of edge unknowns decreases along the solution procedure in Algorithm 5: n_e is the number of the edges of the mesh, $\#L$ the number of the edges of the spanning tree, $n_e^{(1)}$ the number of unknowns left after line 2, $n_e^{(2)}$ the number of unknowns left after line 6. Finally, $\#cc$ is the number of the connected components of the residual graph.

TABLE 7.2
Reduction of the number of unknowns.

	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

In Table 7.3 we report the dimension of the linear system $\text{curl } \mathbf{Z}_h = \mathbf{J}_h$ at the different steps of Algorithm 5, for cases D, E and F. The first column refers to the original system, the second one to the system remaining after line 2, the third one to the system remaining after line 6, the last one to the reduced system constructed at line 8. We denote by n_f , $n_f^{(1)}$, $n_f^{(2)}$ the number of the faces in the different situations, and by $|\mathcal{F}_0|$ the number of the faces for which no degree of freedom has been determined (the so-called 0-faces); the homological constraints are not counted.

TABLE 7.3
Dimension of the linear system at each step.

	$n_f \times n_e$	$n_f^{(1)} \times n_e^{(1)}$	$n_f^{(2)} \times n_e^{(2)}$	$ \mathcal{F}_0 \times \#cc$
Test D	1518464×902388	134087×54273	86186×34506	1175×30
Test E	3509696×2073688	372839×150694	246924×98603	3372×107
Test F	8337664×4913792	686896×275832	531280×212088	7416×145

We have seen that Algorithm 5 terminates at line 6 in cases A, B, C, namely, when the domain is the complement of a non-knotted set. However, this behavior is strongly dependent on the choice of the spanning tree. Table 7.4 illustrates the influence of this choice, showing the number of remaining unknowns after line 6, when adopting a breadth-first search or a depth-first search spanning tree.

TABLE 7.4
Dependence of the reduction on the choice of the spanning tree.

	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

For the loop fields, we can adopt either Algorithm 5 or Algorithm 6. In the latter case, for Test D and Test E the computation of one linking number is enough, while for Test F the procedure has been repeated four times. For the source fields, the reduced system is solved by a direct method. In Table 7.5 we report the CPU time for obtaining all the loop fields by means of Algorithm 6 and one source field by means of Algorithm 5.

TABLE 7.5
CPU time (ms) for computing all the loop fields (their number is indicated in parenthesis) and one source field.

	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

Finally, Figure 7.2 shows the support of a loop field for each test case. It is worth noting that these supports are relatively concentrated.

To conclude we present some numerical tests that illustrate the convergence of the finite element approximation of problem (1.5). Let \mathbf{H} and \mathbf{H}_h be the solution of problem (1.5) and problem (2.1), respectively. In order to compute the relative error $e(h) = \frac{\|\mathbf{H} - \mathbf{H}_h\|_0}{\|\mathbf{H}\|_0}$, we need to know an explicit expression for the exact solution \mathbf{H} .

Having chosen $\mathbf{x}_0 \in \Omega$ and $r_0 > 0$, let us define

$$p(\mathbf{x}) = \begin{cases} q\left(\frac{|x_1 - x_{0,1}|}{r_0}\right) q\left(\frac{|x_2 - x_{0,2}|}{r_0}\right) q\left(\frac{|x_3 - x_{0,3}|}{r_0}\right) & \text{if } \max_{i=1,2,3} |x_i - x_{0,i}| \leq r_0 \\ 0 & \text{if } \max_{i=1,2,3} |x_i - x_{0,i}| > r_0, \end{cases}$$

where q is a regular function with $q(0) = 1$, $q(1) = 0$ and $q'(0) = q'(1) = 0$ (we used the function $q(t) = \frac{1}{2} + \frac{1}{16}[9 \cos(\pi t) - \cos(3\pi t)]$). Let us set $\mathbf{W}(\mathbf{x}) = (0, 0, p(\mathbf{x}))$. It is easy to verify that, if the cube centered in \mathbf{x}_0 with side length $2r_0$ is completely contained

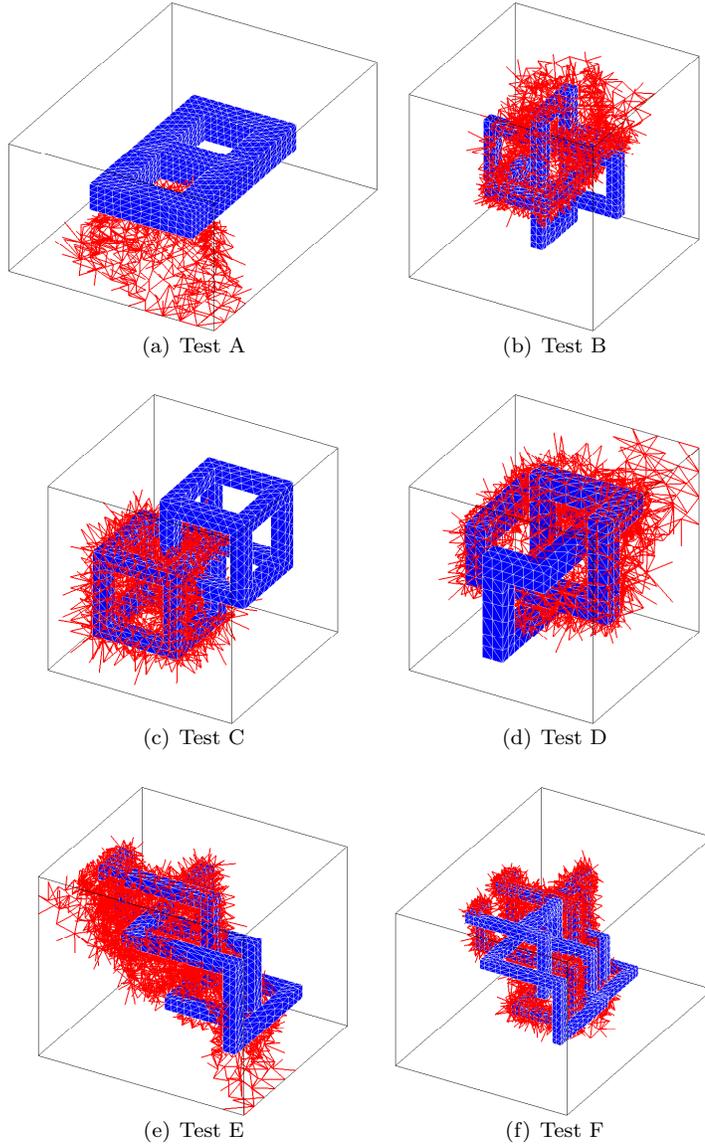


FIG. 7.2. Support of the loop field $\mathbf{T}_{0,j}$, associated to the yellow cycle $\hat{\sigma}_j$ in Figure 7.1.

in Ω and if μ is constant, then $\mathbf{H} = \text{curl} \mathbf{W}$ is the solution of problem (2.1) with $\mathbf{J} = \text{curl} \text{curl} \mathbf{W}$. In fact, since the support of the function p is completely contained in Ω , it follows that $\mathbf{W} \times \mathbf{n} = \mathbf{0}$ on $\partial\Omega$, hence $\mathbf{H} \cdot \mathbf{n} = \text{curl} \mathbf{W} \cdot \mathbf{n} = \text{div}_\tau(\mathbf{W} \times \mathbf{n}) = 0$ on $\partial\Omega$ and $\int_\Omega \mathbf{H} \cdot \boldsymbol{\eta} = \int_\Omega \text{curl} \mathbf{W} \cdot \boldsymbol{\eta} = \int_\Omega \mathbf{W} \cdot \text{curl} \boldsymbol{\eta} - \int_{\partial\Omega} (\mathbf{W} \times \mathbf{n}) \cdot \boldsymbol{\eta} = 0$ for all $\boldsymbol{\eta} \in \mathbb{H}_\mu(\Omega)$.

We consider three different geometric configurations. In the first one the domain Ω is the complement of the torus $[(-1.5, 1.5) \times (-1.5, 1.5) \times (-0.25, 0.25)] \setminus [(-1, 1) \times (-1, 1) \times (-0.25, 0.25)]$ in the box $(-2, 2) \times (-2, 2) \times (-1, 1)$. The support of the

source is centered in the origin and $r_0 = 1$. In the second one the domain Ω is the complement of a 5-torus in the cube centered in the origin with side length equal to 10. The 5-torus, centered with respect to the origin, is similar to that in Test C: the length of the cube side is equal to 6 and the width of each edge is equal to 1. Also in this case the support of the source is centered in the origin with $r_0 = 1$. Finally, in the third example Ω is the complement of a 4_1 -knot as in Test E; the source is centered in $(-3.25, -3.25, -3.25)$ and $r_0 = 2.5$.

In Table 7.6 we report the number of vertices and the relative errors for the three geometric configurations and for five meshes obtained by a repeated uniform refinement. Figure 7.3 shows the plots in a log–log scale of the relative error versus the mesh sizes h , $h/2$, $h/4$, $h/8$ and $h/16$. Linear convergence can be observed.

TABLE 7.6
Relative errors for five different meshes.

	Torus		5-torus		4 ₁ -knot	
	n_v	$e(h)$	n_v	$e(h)$	n_v	$e(h)$
Mesh 1	141	0.733783	307	0.896280	648	0.755626
Mesh 2	865	0.485331	2071	0.690457	4548	0.513166
Mesh 3	5892	0.279500	14796	0.436918	33453	0.296665
Mesh 4	43030	0.148187	110858	0.242721	255063	0.157373
Mesh 5	327786	0.075957	856022	0.126696	–	–

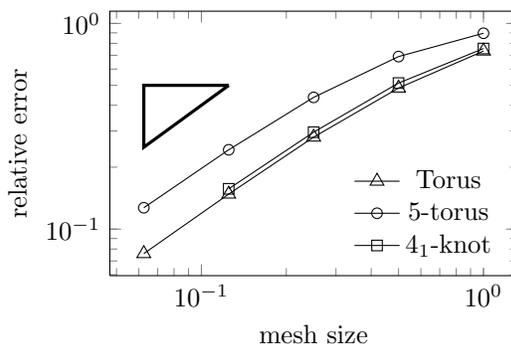


FIG. 7.3. Relative errors versus mesh size h .

REFERENCES

- [1] R. ALBANESE AND G. RUBINACCI, *Integral formulation for 3D eddy-current computation using edge elements*, IEE Proc. A, 135 (1988), pp. 457–462.
- [2] A. ALONSO AND A. VALLI, *An optimal domain decomposition preconditioner for low-frequency time-harmonic Maxwell equations*, Math. Comp., 68 (1999), pp. 607–631.
- [3] A. ALONSO RODRÍGUEZ AND A. VALLI, *Eddy Current Approximation of Maxwell Equations*, Springer-Verlag Italia, Milan, 2010.
- [4] P. ALOTTO AND I. PERUGIA, *A field-based finite element method for magnetostatics derived from an error minimisation approach*, Internat. J. Numer. Methods Engrg., 49 (2000), pp. 573–598.
- [5] C. AMROUCHE, C. BERNARDI, M. DAUGE, AND V. GIRAULT, *Vector potentials in three-dimensional nonsmooth domains*, Math. Meth. Appl. Sci., 21 (1998), pp. 823–864.

- [6] Z. BADICS AND Z. J. CENDES, *Source modeling by mesh incidence matrices*, IEEE Trans. Magn., 43 (2007), pp. 1241–1244.
- [7] S. BALAC AND G. CALOZ, *The reduced scalar potential in regions with permeable materials: Reasons for loss of accuracy and cancellation*, Int. J. Numer. Model., 20 (2007), pp. 163–180.
- [8] M. L. BARTON AND Z. J. CENDES, *New vector finite elements for three-dimensional magnetic field computation*, J. Appl. Phys., 61 (1987), pp. 3919–3921.
- [9] R. BENEDETTI, R. FRIGERIO, AND R. GHILONI, *The topology of Helmholtz domains*. arXiv:1001.4418, 2010.
- [10] A. BERMÚDEZ, R. RODRÍGUEZ, AND P. SALGADO, *A finite element method for the magnetostatic problem in terms of scalar potentials*, SIAM J. Numer. Anal., 46 (2008), pp. 1338–1363.
- [11] E. BERTOLAZZI AND R. GHILONI, *Fast computation of the linking number via an exact explicit formula*. in preparation, 2012.
- [12] A. BOSSAVIT, *Magnetostatic problems in multiply connected regions: some properties of the curl operator*, IEE Proc. A, 135 (1988), pp. 179–187.
- [13] A. BOSSAVIT, *Computational Electromagnetism*, Academic Press Inc., San Diego, 1998.
- [14] F. BREZZI AND A. BUFFA, *Innovative mimetic discretizations for electromagnetic problems*, J. Comput. Appl. Math., 234 (2010), pp. 1980–1987.
- [15] J. CANTARELLA, D. DETURCK, AND H. GLUCK, *Vector calculus and the topology of domains in 3-space*, Amer. Math. Monthly, 109 (2002), pp. 409–442.
- [16] C.-L. CHANG AND M. D. GUNZBURGER, *A finite element method for first order elliptic systems in three dimensions*, Appl. Math. Comput., 23 (1987), pp. 171–184.
- [17] M. V. K. CHARI, G. BEDROSIAN, J. D’ANGELO, AND A. KONRAD, *Finite element applications in electrical engineering*, IEEE Trans. Magn., 29 (1993), pp. 1306–1314.
- [18] J.-L. COULOMB, *Finite element three dimensional magnetic field computation*, IEEE Trans. Magn., MAG-17 (1981), pp. 3241–3246.
- [19] T. DEY AND S. GUHA, *Computing homology groups of simplicial complexes in \mathbb{R}^3* , J. ACM, 45 (1998), pp. 266–287.
- [20] B. DEZS, A. JÜTTNER, AND P. KOVÁCS, *LEMON - an open source C++ graph template library*, Electron. Notes Theor. Comput. Sci., 264 (2011), pp. 23–45.
- [21] P. DLOTKO AND R. SPECOGNA, *Critical analysis of the spanning tree techniques*, SIAM J. Numer. Anal., 48 (2010), pp. 1601–1624.
- [22] ———, *Efficient cohomology computation for electromagnetic modeling*, CMES, 60 (2010), pp. 247–277.
- [23] ———, *Cohomology in electromagnetic modeling*. arXiv:1111.2374v1, 2011.
- [24] ———, *Efficient generalized source field computation for h-oriented magnetostatic formulations*, Eur. Phys. J. Appl. Phys., 53 (2011), pp. 1–7, 20801.
- [25] ———, *A novel technique for cohomology computations in engineering practice*, Comput. Methods Appl. Mech. Engrg., (2012).
- [26] P. DLOTKO, R. SPECOGNA, AND F. TREVISAN, *Automatic generation of cuts on large-sized meshes for the $T - \Omega$ geometric eddy-current formulation*, Comput. Methods Appl. Mech. Engrg., 198 (2009), pp. 3765–3781.
- [27] P. DULAR, *Curl-conform source fields in finite element formulations. automatic construction of a reduced form*, COMPEL, 24 (2005), pp. 364–373.
- [28] 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), pp. 1398–1401.
- [29] C. FOIAS AND R. TEMAM, *Remarques sur les équations de Navier-Stokes stationnaires et les phénomènes successifs de bifurcations*, Ann. Scuola Norm. Sup. Pisa, 5 (IV) (1978), pp. 29–63.
- [30] R. GHILONI, *The Hodge decomposition theorem for general three-dimensional vector fields, without cuts*, Report UTM 731, Department of Mathematics, University of Trento, 2010.
- [31] P. W. GROSS AND P. R. KOTIUGA, *Electromagnetic Theory and Computation: a Topological Approach*, Cambridge University Press, New York, 2004.
- [32] C. S. HARROLD AND J. SIMKIN, *Cutting multiply connected domains*, IEEE Trans. Magn., 21 (1985), pp. 2495–2498.
- [33] T. HENNERON, S. CLÉNET, P. DULAR, AND F. PIRIOU, *Discrete finite element characterizations of source fields for volume and boundary constraints in electromagnetic problems*, J. Comput. Appl. Math., 215 (2008), pp. 438–447.
- [34] 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., 39 (2003), pp. 1167–1170.

- [35] R. HIPTMAIR, *Finite elements in computational electromagnetism*, Acta Numer., 11 (2002), pp. 237–339.
- [36] R. HIPTMAIR AND J. OSTROWSKI, *Generators of $H_1(\Gamma_h, \mathbb{Z})$ for triangulated surfaces: construction and classification*, SIAM J. Comput., 31 (2002), pp. 1405–1423.
- [37] L. KETTUNEN, K. FORSMAN, AND A. BOSSAVIT, *Discrete spaces for div and curl-free fields*, IEEE Trans. Magn., 34 (1998), pp. 2551–2554.
- [38] ———, *Formulation of the eddy current problem in multiply connected regions in terms of h* , Internat. J. Numer. Methods Engrg., 41 (1998), pp. 935–954.
- [39] F. KIKUCHI, *Mixed formulations for finite element analysis of magnetostatic and electrostatic problems*, Japan J. Appl. Math., 6 (1989), pp. 209–221.
- [40] P. R. KOTIUGA, *On making cuts for magnetic scalar potentials in multiply connected regions*, J. Appl. Phys., 61 (1987), pp. 3916–3918.
- [41] ———, *Toward an algorithm to make cuts for magnetic scalar potentials in finite element meshes*, J. Appl. Phys., 63 (1988), pp. 3357–3359. Erratum: *J. Appl. Phys.*, 64 (1988), 4257.
- [42] ———, *An algorithm to make cuts for scalar potentials in tetrahedral meshes based on the finite element method*, IEEE Trans. Magn., 25 (1989), pp. 4129–4131.
- [43] Y. LE MÉNACH, S. CLÉNET, AND F. PIRIOU, *Determination and utilization of the source field in 3D magnetostatic problems*, IEEE Trans. Magn., 34 (1998), pp. 2509–2512.
- [44] 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), pp. 1368–1371.
- [45] K. LIPNIKOV, G. MANZINI, F. BREZZI, AND A. BUFFA, *The mimetic finite difference method for the 3D magnetostatic field problems on polyhedral meshes*, J. Comput. Phys., 230 (2011), pp. 305–328.
- [46] I. D. MAYERGOYZ, M. V. K. CHARI, AND J. D’ANGELO, *A new scalar potential formulation for three-dimensional magnetostatic problems*, IEEE Trans. Magn., 23 (1987), pp. 3889–3894.
- [47] P. MONK, *Finite Element Methods for Maxwell’s Equations*, Oxford University Press, Oxford, 2003.
- [48] J. R. MUNKRES, *Elements of Algebraic Topology*, Addison-Wesley, Menlo Park, 1984.
- [49] J.-C. NÉDÉLEC, *Mixed finite elements in \mathbb{R}^3* , Numer. Math., 35 (1980), pp. 315–341.
- [50] R. A. NICOLAIDES AND X. WU, *Covolume solutions of three-dimensional div-curl equations*, SIAM J. Numer. Anal., 34 (1997), pp. 2195–2203.
- [51] I. PERUGIA, *A mixed formulation for 3D magnetostatic problems: theoretical analysis and face-edge finite element approximation*, Numer. Math., 84 (1999), pp. 305–326.
- [52] 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), pp. 1056–1059.
- [53] F. RAPETTI, F. DUBOIS, AND A. BOSSAVIT, *Discrete vector potentials for nonsimply connected three-dimensional domains*, SIAM J. Numer. Anal., 41 (2003), pp. 1505–1527.
- [54] Z. REN, *\mathbf{T} - Ω formulation for eddy-current problems in multiply connected regions*, IEEE Trans. Magn., 38 (2002), pp. 557–560.
- [55] Z. REN AND A. RAZEK, *Boundary edge elements and spanning tree technique in three-dimensional electromagnetic field computation*, Internat. J. Numer. Methods Engrg., 36 (1993), pp. 2877–2893.
- [56] D. ROLFSEN, *Knots and Links*, Publish or Perish, Berkeley, 1976.
- [57] J. SARANEN, *On generalized harmonic fields in domains with anisotropic nonhomogeneous media*, J. Math. Anal. Appl., 88 (1982), pp. 104–115.
- [58] H. SEIFERT AND W. THRELFALL, *A Textbook of Topology*, Academic Press, New York, 1980.
- [59] J. SIMKIN, S. C. TAYLOR, AND E. X. XU, *An efficient algorithm for cutting multiply connected regions*, IEEE Trans. Magn., 40 (2004), pp. 707–709.
- [60] J. SIMKIN AND C. W. TROWBRIDGE, *Three-dimensional nonlinear electromagnetic field computations, using scalar potentials*, IEE Proc. B, 127 (1980), pp. 368–374.
- [61] T. TARHASAARI AND L. KETTUNEN, *Topological approach to computational electromagnetism*, in “Geometric Methods for Computational Electromagnetics”, EMW Publishing, Boston, 2001.
- [62] J. P. WEBB AND B. FORGHANI, *A single scalar potential method for 3D magnetostatics using edge elements*, IEEE Trans. Magn., 25 (1989), pp. 4126–4128.