Technical Article

The Virtual Element Method for magnetostatics: Two possible approaches

Abstract — **The Virtual Element Method (VEM) is a recent and successful method for the numerical solution of partial differential equations. The methodological background of VEM is presented and its application in magnetic field analysis is shown; accordingly, the numerical solution of linear and non-linear magnetostatic problems is worked out in two- and three-dimensional domains. VEM generalizes the classical Finite Element Method allowing for more general elements (polygons and polyhedra, even non convex) and enhanced local spaces. VEM and FEM can coexist on the same mesh, ensuring the possibility of treating in a seamless way hanging nodes and mesh gluing. An original technique for handling the shape variation in a problem of optimal shape design is finally presented.**

Keywords: Virtual element method, finite element method, polygonal/polyhedral meshes, magnetostatics, permanent magnet models.

I. Introduction

The Virtual Element Method (VEM) is a recent and successful method for the numerical solution of partial differential equations [4]. VEM can be considered as an evolution of the finite element method (FEM) that can make use of very general decompositions of the computational domain into polygonal or polyhedral elements. The Virtual Element Method allows for a generic degree of accuracy, that corresponds to the polynomial degree in standard FEM, and exhibits an excellent robustness with respect to mesh distortions. As a consequence of this flexibility, VEM may be particularly useful in the presence of solutions with varying regularity, for grid adaptation, using moving meshes, and in discretizing complex geometries (just to name few).

VEM and FEM are perfectly compatible, hence it is possible (and advisable) to use Virtual Elements only in those parts of the domain where they are needed.

In the framework of magnetostatic problems, the VEM was developed both for the potential [7] and the Kikuchi [15, 16] formulations, although only in the linear regime. The subject of this article is to further develop the method for applicative purposes, such as the computation of quantities of engineering interest (like, e.g., forces and torques acting on structures placed in a magnetic field), and the generalization to the non-linear framework considering the $\mathbf{B}$-$\mathbf{H}$ curve of ferromagnetic materials.

To this end, several case studies are considered: the analysis of a C-core electromagnet used as an actuator, a four-pole permanent-magnet motor at no-load condition, the three-dimensional analysis of a cylindrical electromagnet. In order to validate the proposed approach, comparisons of both virtual and finite element potential formulations are presented and discussed. Overall, the flexibility of VEM in mesh generation for complex computational domains is tested: it is shown that such flexibility can be further exploited combining virtual and finite element approaches. An original technique for handling the shape variation in a problem of optimal shape design is finally presented; in this respect, the optimal shape design of a magnetic pole for synthesizing a uniform field profile in a region of interest is considered, which is a classical benchmark problem.

II. Potential formulation

Considering the classical potential formulation in two dimensions, we search for the scalar field A, i.e., the potential associated with the magnetic induction field $\mathbf{B} = \mathbf{curl}\, A$ such that

$$\text{curl}\,(\mu^{-1}\,\mathbf{curl}\, A) = j \quad \text{in}\, \Omega\,, \tag{1}$$

plus boundary conditions on $\partial\Omega$. The two-dimensional $\mathbf{curl}$ and the scalar curl are defined as

$$\text{curl}\,\mathbf{A} := \partial_x \mathbf{A}_2 - \partial_y \mathbf{A}_1 \quad \text{and} \quad \mathbf{curl}\, A := (\partial_y A, -\partial_x A)\,.$$

In Equation (1), the scalar μ represents the material permeability, while j is the assigned current density which in two dimensions is a scalar quantity. It turns out to be

$$\begin{aligned} \text{curl}\,(\mu^{-1}\,\mathbf{curl}\, A) &= \partial_x(-\mu^{-1}\partial_x A) - \partial_y(\mu^{-1}\partial_y A) \\ &= -\text{div}(\mu^{-1}\nabla A)\,, \end{aligned}$$

so Equation (1) becomes

$$-\text{div}(\mu^{-1}\nabla A) = j \quad \text{in}\, \Omega\,,$$

subject to appropriate boundary conditions.

III. The Virtual Element for the Poisson problem

As we have seen, the Potential formulation for Magnetostatics consists of a single Poisson equation with variable diffusion.

In this section we will describe the Virtual Element Method in the simpler case of Poisson equation with constant diffusion. The same construction can be easily extended to a piecewise constant diffusion, while for the general case we refer to [8].

Solving Poisson equation with constant diffusion and homogeneous Dirichlet boundary condition amounts to find the solution of the problem:

$$\begin{cases} -\Delta u = f & \text{in}\, \Omega \\ \quad u = 0 & \text{on}\, \partial\Omega \end{cases} \tag{2}$$

which can be written in variational form as follows:

$$\begin{cases} \text{find } u \in H^1_0(\Omega) \text{ such that} \\ \displaystyle\int_\Omega \nabla u \cdot \nabla v \,\mathbf{dx} = \int_\Omega f v \,\mathbf{dx} \quad \text{for all } v \in H^1_0(\Omega). \end{cases} \tag{3}$$

Classical conforming Finite Element methods require the following steps:

- discretize the domain Ω with non-overlapping triangles $\{T\}$;
- construct in each triangle through suitable basis functions a local space $V_h^k(T)$ made of polynomials up to degree k;
- assemble the global space $V_h^k(\Omega) \subseteq H^1_0(\Omega)$ by gluing together in a continuous fashion the local spaces, respecting global boundary conditions;

- solve the approximate variational problem

$$\begin{cases} \text{find } u_h \in V_h^k(\Omega) \text{ such that} \\ \displaystyle\int_\Omega \nabla u_h \cdot \nabla v_h \, \mathbf{dx} = \int_\Omega f v_h \, \mathbf{dx} \quad \text{for all } v_h \in V_h^k(\Omega). \end{cases} \tag{4}$$

If k is an integer and $\mathcal{O} \subseteq \mathbb{R}^2$, we define $\mathbb{P}_k(\mathcal{O})$ as the space of polynomials of degree up to k defined on $\mathcal{O}$. In the following, $\mathcal{O}$ will be a segment or a polygon.

We start with $k = 1$.

A. The Courant triangle

Given a triangle T, we define the local finite element space of degree one on T by

$$V_h^1(T) := \mathbb{P}_1(T)\,, \tag{5}$$

and if $\boldsymbol{V}_1$, $\boldsymbol{V}_2$ and $\boldsymbol{V}_3$ are the vertices of T, the *local degrees of freedom* of a function $v_h \in V_h^1(T)$ are defined by

$$\mathrm{dof}_i(v_h) := v_h(\boldsymbol{V}_i). \tag{6}$$

In other words, we are saying that a function $v_h \in V_h^1(T)$ is completely determined by its values at the vertices of the triangle.

The local basis functions $\varphi_i \in V_h^1(T)$ are defined in terms of the local degrees of freedom by the following condition:

$$\mathrm{dof}_i(\varphi_j) = \delta_{ij}. \tag{7}$$

The previous equation simply means that φ_i is the only linear function that is one on vertex $\boldsymbol{V}_i$ and zero on the other vertices. The set of functions $\{\varphi_1, \varphi_2, \varphi_3\}$ is a basis for the space $V_h^1(T)$.

We now characterize the space $V_h^1(T)$ in a different way that will be crucial for the rest of the section. Given that there exists a unique harmonic function with assigned value at the boundary, the local space $V_h^1(T)$ can also be identified by the following properties:

- for each edge e of T, $v_{h|e}$ is linear on e;
- the linear functions $v_{h|e}$ on the edges match at the vertices;
- v_h is harmonic inside, i.e. $\Delta v_h = 0$ in T.

The properties above will allow us to extend the definition of “linear triangular element” to a general polygon.

B. Extension to general polygons

The advantage of the previous characterization is that it can be extended in a seamless way to a general polygon P (even non convex). In this case we will obtain a space $V_h^1(P)$ that contains linear polynomials, together with other functions.

Given a polygon P, we define the space $V_h^1(P)$ in the following way:

- functions are linear on each edge and match at the vertices, see Fig .1;

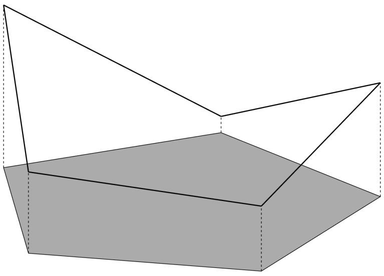


Fig. 1. Virtual functions are linear on each edge and match at the vertices

- they are harmonic inside, see Fig. 2.

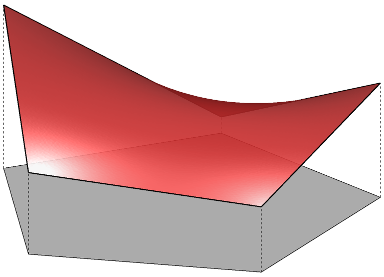


Fig. 2. Virtual functions are harmonic inside

It is clear that if P is a triangle, we recover the Courant element.

The space $V_h^1(P)$ is known in literature as *harmonic finite elements on a polygon*. The i-th local degree of freedom is still defined as the value of the function at vertex i; in fact, a function in $V_h^1(P)$ is completely determined by its values at the vertices. Hence

$$\text{dimension of } V_h^1(P) = \#\text{ of vertices of } P := N^V.$$

The local basis function φ_i takes the value one at vertex i and is zero at the other vertices, see Fig. 3.

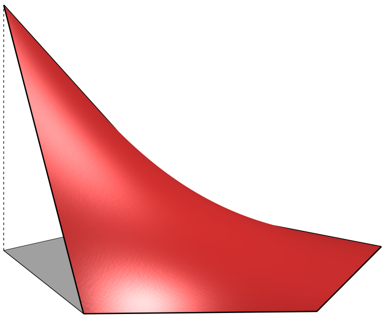


Fig. 3. Local basis function

Of course, the value of φ_i at a point inside the polygon is not readily available, since we only know that φ_i is harmonic.

The key property of the space $V_h^1(P)$ is that it *contains linear polynomials*:

$$\mathbb{P}_1(P) \subseteq V_h^1(P). \tag{8}$$

In fact, if $p_1 \in \mathbb{P}_1(P)$ then

- ✓ p_1 is linear on each edge;
- ✓ p_1 is continuous on the boundary of the polygon;
- ✓ p_1 is harmonic: $\Delta p_1 \equiv 0$.

The fact that linear polynomials are inside the space, ensures the good approximation properties of classical FEM: if we were able to compute the stiffness matrices, the method would converge at the expected rates.

C. The global space

The global finite element space is defined exactly as for classical Finite Element: the local spaces on adjacent polygons sharing an edge glue together continuously through the common edge, producing globally continuous approximation functions.

We point out that the extra non linear functions of $V_h^1(P)$, which exist if the polygon is not a triangle, ensure continuity when two elements are joined together, giving a conformal method. If we allow discontinuous functions across elements, we could define the local space on each polygon simply as $\mathbb{P}_1(P)$. This choice leads to the Discontinuous Galerkin Method (DG). The main difference with respect to VEM is that DG is non-conformal.

D. The projection Π_1^∇

We cannot compute in a cheap way the value of a function $v_h \in V_h^1(P)$ inside P starting from the degrees of freedom. In fact, we can easily recover the function on the boundary, being linear on each edge; but computing the value inside the polygon would require the solution of Laplace equation with given boundary data. Hence it is not feasible to directly compute the local stiffness matrices by quadrature.

We show that we can compute directly from the degrees of freedom a linear polynomial p_1 that approximate v_h in the following integral sense:

$$\int_P \nabla(p_1 - v_h)\cdot\nabla q_1 \,\mathbf{dx} = 0 \quad \text{for all linear polynomial } q_1\,. \tag{9}$$

In other words, p_1 is the projection of v_h with respect to the scalar product $\int_P \nabla u \cdot \nabla v \,\mathbf{dx}$. We denote the projection p_1 by $\Pi_1^\nabla v_h$. In order to show that (9) can be solved by knowing only the degrees of freedom of v_h (i.e. the value of v_h at the vertices of the polygon), we argue in the following way.

If we express the unknown $\Pi_1^\nabla v_h$ in the monomial basis of linear polynomials, namely $\{1, x, y\}$, and we let q_1 vary among $\{1, x, y\}$, we obtain a 3×3 linear system. Let $m_1 := 1$, $m_2 := x$ and $m_3 := y$; we write

$$\Pi_1^\nabla v_h = \sum_{\alpha=1}^{3} c_\alpha\, m_\alpha \quad c_\alpha = \text{unknowns}$$

and (9) becomes

$$\int_P \nabla\left(\sum_{\alpha=1}^{3} c_\alpha\, m_\alpha - v_h\right)\cdot\nabla m_\beta \,\mathbf{dx} = 0, \quad \beta = 1,2,3$$

or equivalently

$$\sum_{i=1}^{3} c_\alpha \left[\int_P \nabla m_\alpha \cdot \nabla m_\beta \,\mathbf{dx}\right] = \int_P \nabla v_h \cdot \nabla m_\beta \,\mathbf{dx},\ \beta = 1,2,3. \tag{10}$$

The 3×3 matrix $\int_P \nabla m_\alpha \cdot \nabla m_\beta \,\mathbf{dx}$ is singular, since it has the first row (and the first column) identically zero: in fact $\nabla m_1 \equiv 0$. This feature reflects the fact that (9) determines the projection $\Pi_1^\nabla v_h$ only up to a constant, or, in other words, that (9) determines only the gradient of $\Pi_1^\nabla v_h$. In order to completely recover $\Pi_1^\nabla v_h$ we need to substitute the first equation in (10) with a (computable) condition guaranteeing that when v_h is constant, $\Pi_1^\nabla v_h = v_h$. This can be achieved by asking, for instance, that

$$\sum_{i=1}^{N^V} \Pi_1^\nabla v_h(V_i) = \sum_{i=1}^{N^V} v_h(V_i).$$

The final 3×3 final linear system is then

$$\begin{cases} \displaystyle\sum_{\alpha=1}^{3} c_\alpha \sum_{i=1}^{N^V} m_\alpha(V_i) = \sum_{i=1}^{N^V} v_h(V_i) \quad \text{(first row)} \\ \displaystyle\sum_{\alpha=1}^{3} c_\alpha \left[\int_P \nabla m_\alpha \cdot \nabla m_\beta \,\mathbf{dx}\right] = \int_P \nabla v_h \cdot \nabla m_\beta \,\mathbf{dx},\ \beta = 2,3. \end{cases} \tag{11}$$

We need to show that the matrix and the right-hand-side of the linear system (11) are directly computable out of the degrees of freedom of v_h, i.e. the values of v_h at the vertices.

- The matrix of (11) clearly does not depend on v_h, and its coefficients are simple computable from the monomial basis which is explicitly given;
- as far as the right-hand-side is concerned, in the first line of (11) there are the values of v_h at the vertices, while in lines 2 and 3 there is the expression

  $$\int_P \nabla v_h \cdot \nabla m_\beta \,\mathbf{dx}, \quad \beta = 2,3.$$

  Integrating by parts, we have

  $$\begin{aligned}\int_P \nabla v_h \cdot \nabla m_\beta \,\mathbf{dx} &= \\ = -\int_P v_h\, \Delta m_\beta \,\mathbf{dx} + \int_{\partial P} v_h \frac{\partial m_\beta}{\partial n}\,\mathrm{d}s &= \\ = \int_{\partial P} v_h \frac{\partial m_\beta}{\partial n}\,\mathrm{d}s\end{aligned}$$

  since $\Delta m_\beta \equiv 0$, m_β being a linear monomial. Recalling that v_h is linear on each edge, starting from the degrees of freedom of v_h, i.e. the values of v_h at the vertices, we can easily reconstruct v_h on the boundary of P and hence compute $\displaystyle\int_{\partial P} v_h \frac{\partial m_\beta}{\partial n}\,\mathrm{d}s$.

An explicit formula for $\Pi_1^\nabla v_h$ is given by

$$\Pi_1^\nabla v_h(\mathbf{z}) = (\mathbf{z} - \overline{\boldsymbol{V}}) \cdot \frac{1}{|P|}\int_P \nabla v_h \,\mathbf{dx} + \bar{v}_h \tag{12}$$

where $\overline{\boldsymbol{V}}$ and $\bar{v}_h$ are defined as

$$\overline{\boldsymbol{V}} := \frac{1}{N^V}\sum_{i=1}^{N^V} \boldsymbol{V}_i, \quad \bar{v}_h := \frac{1}{N^V}\sum_{i=1}^{N^V} v_h(\boldsymbol{V}_i),$$

see [2] for the details. Note that

$$\nabla[\Pi_1^\nabla v_h] = \frac{1}{|P|}\int_P \nabla v_h \,\mathbf{dx},$$

and that the latter term can be directly computed from the degrees of freedom of v_h by observing that $\int_P \nabla v_h \,\mathbf{dx} = \int_{\partial P} v_h \mathbf{n}\,\mathrm{d}s$.

E. The local stiffness matrix for the Poisson equation

Given that the degrees of freedom of the basis functions φ_i are known by definition (they are all zeros except the i-th which is

1), we can easily compute $\Pi_1^\nabla \varphi_i$ and it seems to be a good idea to make the following approximation of the "true" local stiffness matrix for the Poisson equation:

$$\int_P \nabla\varphi_j \cdot \nabla\varphi_i \, \mathrm{d}\mathbf{x} \approx \int_P \nabla\Pi_1^\nabla \varphi_j \cdot \nabla\Pi_1^\nabla \varphi_i \, \mathrm{d}\mathbf{x}. \tag{13}$$

However, this is not appropriate because the $N^V \times N^V$ matrix $\int_P \nabla\Pi_1^\nabla \varphi_j \cdot \nabla\Pi_1^\nabla \varphi_i \, \mathrm{d}\mathbf{x}$ is rank-deficient, giving rise to a singular global matrix. The right rank for the local matrix $\int_P \nabla\Pi_1^\nabla \varphi_j \cdot \nabla\Pi_1^\nabla \varphi_i \, \mathrm{d}\mathbf{x}$ would be $N^V - 1$, because the constant functions are clearly in the kernel but they are ruled out by the global boundary conditions, giving at the end an invertible matrix.

Instead, it can be easily shown that the rank is 2 which is strictly less than $N^V - 1$ unless P is a triangle. For, we observe that the kernel of the projection operator $\Pi_1^\nabla : V_h^1(P) \to \mathbb{P}_1(P)$ has dimension $N^V - 3$ (since $\dim \mathbb{P}_1(P) = 3$), and the gradient annihilates the constant functions. In the global space the constant functions are ruled out by the boundary condition, but the global stiffness matrix remains singular, unless all polygons are triangles.

F. Consistency

Using (13) amounts to replacing (4) with

$$\begin{cases} \text{find } u_h \in V_h^1(\Omega) \text{ such that} \\ \displaystyle\int_\Omega \nabla\Pi_1^\nabla u_h \cdot \nabla\Pi_1^\nabla v_h \, \mathrm{d}\mathbf{x} = \int_\Omega f v_h \, \mathrm{d}\mathbf{x} \quad \text{for all } v_h \in V_h^1(\Omega) \end{cases} \tag{14}$$

where

$$\int_\Omega \nabla\Pi_1^\nabla u_h \cdot \nabla\Pi_1^\nabla v_h \, \mathrm{d}\mathbf{x} = \sum_T \int_T \nabla\Pi_1^\nabla u_h \cdot \Pi_1^\nabla \nabla v_h \, \mathrm{d}\mathbf{x}.$$

In this paper we do not discuss how to approximate the load term; we refer to [2] for the details.

As explained before, problem (14) is singular. However, if it had a unique solution u_h, then the method would be *consistent*, i.e. if the exact solution u were a global linear polynomial p_1, then $u_h = p_1$. In fact, by (3), $u = p_1$ would solve

$$\int_\Omega \nabla p_1 \cdot \nabla v_h \, \mathrm{d}\mathbf{x} = \int_\Omega f v_h \, \mathrm{d}\mathbf{x} \quad \text{for all } v_h \in V_h^1(\Omega).$$

By the definition of the projector Π_1^∇, we have

$$\int_\Omega \nabla p_1 \cdot \nabla v_h \, \mathrm{d}\mathbf{x} = \int_\Omega \nabla p_1 \cdot \nabla\Pi_1^\nabla v_h \, \mathrm{d}\mathbf{x} \tag{15}$$

and since $\Pi_1^\nabla p_1 = p_1$, we also have

$$\int_\Omega \nabla p_1 \cdot \nabla\Pi_1^\nabla v_h \, \mathrm{d}\mathbf{x} = \int_\Omega \nabla\Pi_1^\nabla p_1 \cdot \nabla\Pi_1^\nabla v_h \, \mathrm{d}\mathbf{x} \, .$$

Hence p_1 satisfies the equation

$$\int_\Omega \nabla\Pi_1^\nabla p_1 \cdot \nabla\Pi_1^\nabla v_h \, \mathrm{d}\mathbf{x} = \int_\Omega f v_h \, \mathrm{d}\mathbf{x} \quad \text{for all } v_h \in V_h^1(\Omega),$$

i.e. $u_h \equiv p_1$.

Property (15) is crucial: it ensures *consistency*. However, alone it is not enough: we need to add *stability*.

G. Stabilization

We need to supplement (14) with a *stabilization term* that

- guarantees existence and uniqueness;
- does not spoil consistency (equation (15));
- it is defined element by element.

It turns out that we can add to (14) a term of the form

$$\begin{aligned} S\big((\mathrm{I} - \Pi_1^\nabla)u_h, (\mathrm{I} - \Pi_1^\nabla)v_h\big) &= \\ &= \sum_T S_T\big((\mathrm{I} - \Pi_1^\nabla)u_h, (\mathrm{I} - \Pi_1^\nabla)v_h\big) \end{aligned} \tag{16}$$

where $S_T(\cdot,\cdot)$ is a symmetric coercive bilinear form that scales in the right way. Nota that if one of the entries is a linear polynomial p_1, since $(\mathrm{I} - \Pi_1^\nabla)p_1 \equiv 0$ the term (16) is zero so consistency is preserved.

It can be shown that under certain assumption on the mesh, we have convergence at the expected rates. There is much freedom in the choice of the local stability bilinear form $S_T(\cdot,\cdot)$. One possibility is the so-called D-recipe, i.e. to define $S_T(\cdot,\cdot)$ by

$$S_T(\varphi_i, \varphi_j) = \begin{cases} \max\left\{1, \displaystyle\int_P |\nabla\Pi_1^\nabla \varphi_i|^2 \, \mathrm{d}\mathbf{x}\right\} & \text{if } i = j \\ 0 & \text{if } i \neq j. \end{cases}$$

H. The case $k = 2$

We briefly consider the case $k = 2$. As for finite element on triangles, we need to add the middle point of each edge in order to have polynomial of degree 2 on each edge. Since the Laplacian of a degree 2 polynomial is constant, we substitute the condition $\Delta v_h \equiv 0$ with $\Delta v_h =$ constant. The degrees of freedom of a function $v_h \in V_h^2(P)$ are:

- boundary degrees of freedom: as before, the pointwise values at the vertices and at the middle point of the egdes;
- internal degrees of freedom: the mean value on P, i.e., $\displaystyle\frac{1}{|P|}\int_P v_h \, \mathrm{d}\mathbf{x}$.

Hence we have $\dim V_h^2(P) = 2N^V + 1$. Note that in this case if P is a triangle we do not recover the classical finite elements of order two but a larger space.

It is immediate to verify that $\mathbb{P}_2(P) \subseteq V_h^2(P)$.

We need to check that we can compute the Π_2^∇ projection starting from the degrees of freedom. Considering the construction above for the case $k = 1$, it turns out that we only need to be able to compute the right hand side of the linear system, i.e.

$$\int_P \nabla v_h \cdot \nabla m_\beta \, \mathrm{d}\mathbf{x}, \quad \beta = 1, 2, \ldots, 6$$

where m_β is a monomial in two variables up to degree 2 ($m_1 := 1$, $m_2 := x$, $m_3 := y$, $m_4 := x^2$, $m_5 := xy$, $m_6 := y^2$). Integrating by parts, we have as before

$$\int_P \nabla v_h \cdot \nabla m_\beta \, \mathrm{d}\mathbf{x} = -\int_P v_h \, \Delta m_\beta \, \mathrm{d}\mathbf{x} + \int_{\partial P} v_h \, \frac{\partial m_\beta}{\partial n} \, \mathrm{d}s. \tag{17}$$

This time Δm_β is zero for $\beta = 1, 2, 3$ but is constant for $\beta = 4, 5, 6$. Hence the integral of v_h can be readily computed starting from the internal degree of freedom. The rest of the construction works in the same way. Of course, formula (12) is no longer valid for $k = 2$.

I. The general case

In the general case, the functions in $v_h \in V_h^k(P)$ are polynomials of degree k on each edge of P, they match at the vertices and Δv_h is a polynomial of degree $k-2$. These conditions ensure that $\mathbb{P}_k(P) \subseteq V_h^k(P)$. The boundary degrees of freedom are the pointwise values at the vertices and at $k-1$ internal nodes on each edge (which can be taken equispaced as in classical FEM), and the internal degrees of freedom are the *moments* up to order $k-2$ in P, i.e.

$$\frac{1}{|P|}\int_P v_h m \, \mathbf{dx}, \quad m \text{ monomial up to degree } k-2. \qquad (18)$$

If we consider equation (17) in this case, we observe that Δm_β is a polynomial of degree up to $k-2$, so the term $\int_P v_h \Delta m_\beta \, \mathbf{dx}$ can be easily computed from the internal degrees of freedom of v_h. In [2] you can find all the details.

IV. Kikuchi's formulation

Let us recall the formulation introduced in [19]: find the pair $(\mathbf{H}, p)$ solution of

$$\begin{cases} \displaystyle\int_\Omega \operatorname{curl} \mathbf{H} \operatorname{curl} \mathbf{v} \, \mathbf{dx} + \int_\Omega \nabla p \cdot \mu \mathbf{v} \, \mathbf{dx} = \int_\Omega j \operatorname{curl} \mathbf{v} \, \mathbf{dx} \quad \forall \mathbf{v} \\ \displaystyle\int_\Omega \nabla q \cdot \mu \mathbf{H} \, \mathbf{dx} = 0 \quad \forall q \end{cases} \qquad (19)$$

where $\mathbf{H}$ is the magnetic field, and p (which turns out to be identically zero) plays the role of a Lagrange multiplier to enforce weakly the condition $\text{div}\mathbf{B} = 0$ (see the second equation). With the lowest order Virtual Element approximation for (19), the scalar variable p is approximated by a function linear on each edge, and harmonic in each element. The vector variable $\mathbf{H}$ is approximated by a vector $\mathbf{H}_h$ with tangential component constant on each edge e of Ω_h, constant curl and zero divergence:

- $p_{h|e} \in \mathbb{P}_1(e)\ \forall e$ of Ω_h, $\Delta p_h = 0$ in $P\ \forall P \in \Omega_h$
- $\mathbf{H}_h \cdot \mathbf{t}_{|e} \in \mathbb{P}_0(e)\ \forall e$ of Ω_h
- $\text{div}\mathbf{H}_h = 0$, $\operatorname{curl} \mathbf{H}_h \in \mathbb{P}_0(E)\ \forall P \in \Omega_h$

The degrees of freedom for p_h are the values at the vertices, while $\mathbf{H}_h$ is individuated by the constant value of the tangential component on each edge. We point out that on a triangle the VEM space for p_h coincides with the linear Lagrange Finite Element space, while the space for $\mathbf{H}_h$ coincides with the lowest-order edge element space (see [20]). In the general case of order $k \geq 2$ the definition of the spaces is

- $p_{h|e} \in \mathbb{P}_k(e)\ \forall e$ of Ω_h, $\Delta p_h \in \mathbb{P}_{k-2}(E)\ \forall E \in \Omega_h$
- $\mathbf{H}_h \cdot \mathbf{t}_{|e} \in \mathbb{P}_{k-1}(e)\ \forall e$ of Ω_h
- $\text{div}\mathbf{H}_h \in \mathbb{P}_{k-2}(E)$, $\operatorname{curl} \mathbf{H}_h \in \mathbb{P}_{k-1}(E)\ \forall E \in \Omega_h$.

For the choice of the degrees of freedom we refer to [15].

V. VEM vs FEM: a comparison

On triangles for $k = 1$ the VEM formulation for the potential coincides with the standard piecewise linear FEM, and the VEM Kikuchi formulation coincides with the lowest-order edge elements (see [20]). This is not the case for $k \geq 2$, and both VEM approaches exhibit more internal degrees of freedom than their FEM counterpart. However, through the so-called Serendipity procedure, these additional degrees of freedom can be eliminated, so that on triangles we recover exactly both standard and edge FEM ([5]-[6]).

In any case, the VEM of order k always yields an optimal convergence rate for both formulations. The VEM flexibility allows for more efficient refinements in the presence of solution singularities and thus may lead to a better "problem size to accuracy" ratio. On the other hand, since VEM are defined on the physical element and do not make use of a reference element, the calculation of the local matrices may be more costly than with FEM, although such a procedure can be naturally executed in parallel, thus reducing significantly its costs. Another important observation is that, as it happens with FEM, the VEM basis functions are local, and therefore the ensuing matrix in the linear(ized) system enjoys the same sparse structure. Moreover, since the VEM functions are polynomials of degree k on the edges and, consequently, their degrees of freedom coincide with the ones of FEM, one can use a standard FEM method combined with VEM in the same mesh.

VI. Numerical case studies

In this section we provide some numerical tests to prove the validity of the proposed method. Since the theoretical trend of the error was already proved in [15] and in [16, 14] for the two and three dimensional case, we will focus on the reliability of the method from the applicative point of view. Indeed, we consider a series of numerical experiments and we compare the results obtained via VEM and standard FEM. To achieve this goal we consider the results provided by MagNet code as a reference solutions [13]. Moreover, in the last example, where we analyze the three dimensional case, we exploit *FLUX2D* code as a reference solution [1].

In all the following examples we consider both the Kikuchi and the potential formulations and we refer to such strategies as VEM Kik and VEM pot, respectively.

A. C-core actuator

In this section we present the numerical simulation of a C-core electromagnet via VEM. The electromagnet is composed of a fixed C-shaped core and a movable plunger. A DC current of 1 A supplies the winding around the core limb which in turn excites the magnetic field lines. The overall size of the electromagnet is 80 mm×60 mm, while the cross-sectional area of the winding, which incorporates 1000 turns, is equal to 400 mm^2.

We are interested in computing the following quantities

1. B_x: the x-component of the field $\mathbf{B}$ in the mid-point of the air gap;
2. F_x: the x-component of the force $\mathbf{F}$ acting on the plunger, via Maxwell's stress tensor method,

for different position of the plunger with respect to the C-core. Since we are considering small air-gap widths, a non-linear approximation of the magnetic permeability is required. In the following experiments we define the material of both C-core and plunger as a standard laminated iron with 5 mm thickness [12].

This example presents two interesting aspects that will challenge the proposed method. First of all the electromagnet exhibits a very thin dielectric region (air-gap) located between two large ferromagnetic regions (core and plunger). Such double jump of permeability makes the discontinuity of magnetic field $\mathbf{H}$ concentrated in a very small region and this feature will challenge the accuracy of any method of field simulation.

Then, we numerically validate the usage of projection operators in a post processing procedure. Indeed, since we are interested in the computation of the force-displacement curve based on the Maxwell's stress, we need to access on both field components, but we do not have such information: the field $\mathbf{H}$ is virtual so we know it *only* via its degrees of freedom and we can approximate it via the projection operator.

In view of the simulation, we can *a priori* identify regions which are particularly interesting from the physical point of view and the remaining regions. Consequently we can generate a mesh refined in the "hot" regions and coarse in the other ones exploiting as much as possible the possibility to add hanging-nodes. On the one hand we have to refine at the corners of the C-shaped core where there are singularities of the magnetic permeability and the boundary of the plunger, where there are singularities of the magnetic field and at the air-gap between the C-core and the plunger. On the other hand we can coarse the mesh inside the two current-carrying regions forming the winding and also in the region far away from the C-core electromagnet that represents the truncated air domain.

Moving from such considerations, we build the mesh shown in Fig. 4. We observe that such mesh satisfies all the requirements we made. Indeed, it is refined according to the a-priori information we give about the physics of the problem at hand, see the details in Fig. 4 (c) and (d), and it has few elements where it is not necessary, see Fig. 4 (a) and (b). In Fig. 4 (b) we further underline that we have succeeded in having only one big element exhibiting hanging nodes, i.e., the squares simulating the winding cross section are not split, but their boundary is split in a non-uniform way.

We consider a fixed air-gap of 5 mm, we use the same VEM mesh for both Kikuchi and potential formulation, while we consider a uniform mesh composed of triangles to get a FEM solution via MagNet.

In Table I we show the values of B_x and F_x for each method and approximation degree. If we fix the method and we vary the approximation degree, we observe that such values becomes stable. Indeed, the first two digits of B_x and F_x stay the same.

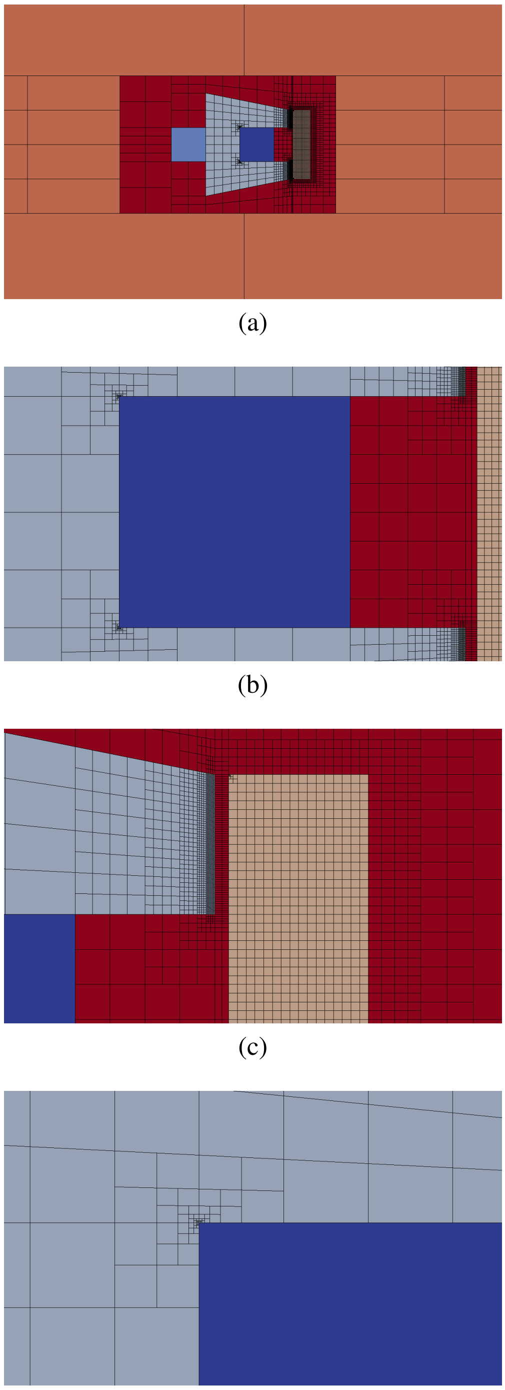
	VEM Kik	VEM Pot	FEM
degree	B_x [T]	B_x [T]	B_x [T]
1	0.1225	0.1228	0.1228
2	0.1250	0.1250	0.1227
3	0.1250	0.1250	0.1227

	VEM Kik	VEM Pot	FEM
degree	F_x [N]	F_x [N]	F_x [N]
1	-5.7325	-5.9803	-5.9898
2	-6.0334	-6.2475	-6.1091
3	-6.0976	-6.2717	-6.1254

TABLE I. C-CORE EXAMPLE: B_x [T] AND F_x [N] COMPUTED BY FEM AND VEM.

On the other hand if we fix the approximation degree and we vary the method, we see that the computed values have the same order and they are close to each other. This fact is a numerical evidence that the solutions provided by VEM are compatible with the one obtained by FEM. However, since the FEM solution is computed on a different mesh, we can not obtain exactly the same values. In particular, MagNet makes some post processing on the computed solution to get smooth values of both B_x and F_x and, consequently, its solution converges to different values with respect to the VEM ones.

Table II reports the number of degrees of freedom required to get VEM and FEM solutions.


(a)

(b)

(c)

(d)

Fig. 4. C-core example: the resulting mesh for the C-core example: (a) the whole mesh with large element around the electromagnet; (b) winding cross-section: one element characterized by several hanging-nodes; (c) refined region at the air-gap; (d) refined region around a corner of the C-core.

degree	VEM Kik		VEM Pot		FEM
1	66 607	53%	24 494	19%	124 080
2	132 217	26%	48 488	9%	495 097
3	215 449	19%	78 356	7%	1 113 053

TABLE II. C-CORE EXAMPLE: COMPARISON BETWEEN FEM AND VEM DOFS.

We observe that the amount of degrees of freedom for the two VEM approximations is in accordance with the estimates given in the previous section. However, despite the high number of degrees of freedom in the Kikuchi formulation, the flexibility in mesh generation allows us to get the result in Table I with fewer degrees of freedom than with FEM. We highlight such fact by reporting in Table II the percentage of dofs used with respect to the FEM ones.

Thanks to the previous experiments we validate the resolution provided by VEM. Now we proceed with the evaluation of B_x e F_x varying the air-gap width considering *only* the proposed VEM methods. Such data are collected in Table III.

d [mm]		0.25	0.50	1.00	2.00
B_x [T]	VEM Kik	2.0428	1.1176	0.5870	0.3015
	VEM Pot	2.0449	1.1185	0.5875	0.3017
F_x [N]	VEM Kik	-1360.5	-413.00	-117.09	-32.498
	VEM Pot	-1365.5	-416.32	-119.26	-33.433

TABLE III. C-CORE EXAMPLE: VALUES OF B_x AND F_x BY VARYING AIR-GAP WIDTHS.

Both formulations give similar results by varying the air-gap widths. Indeed B_x and F_x approximately scale as d^{-1} and d^{-2}, respectively.

In Fig. 5 we show both the vector field $\mathbf{B}$ at mesh vertices and the value of μ_r at the quadrature points. To compute the vector field $\mathbf{B}$ at a vertex ν, we make the mean of the vectors fields defined on the elements sharing the vertex ν. Such data are computed via VEM potential formulation (Kik VEM is similar so we do not show it). The behaviour of magnetic field and the magnetic permeability are the expected ones. Indeed, the vectors are properly aligned and the strength of $\mathbf{B}$ increases for small air-gap widths. Moreover, both the C-core and the plunger have a uniform μ_r for large air-gap widths, $d = 1.00$ mm and $d = 2.00$ mm, while the material starts to saturate when d is small.

B. INTERIOR PERMANENT MAGNET MOTOR

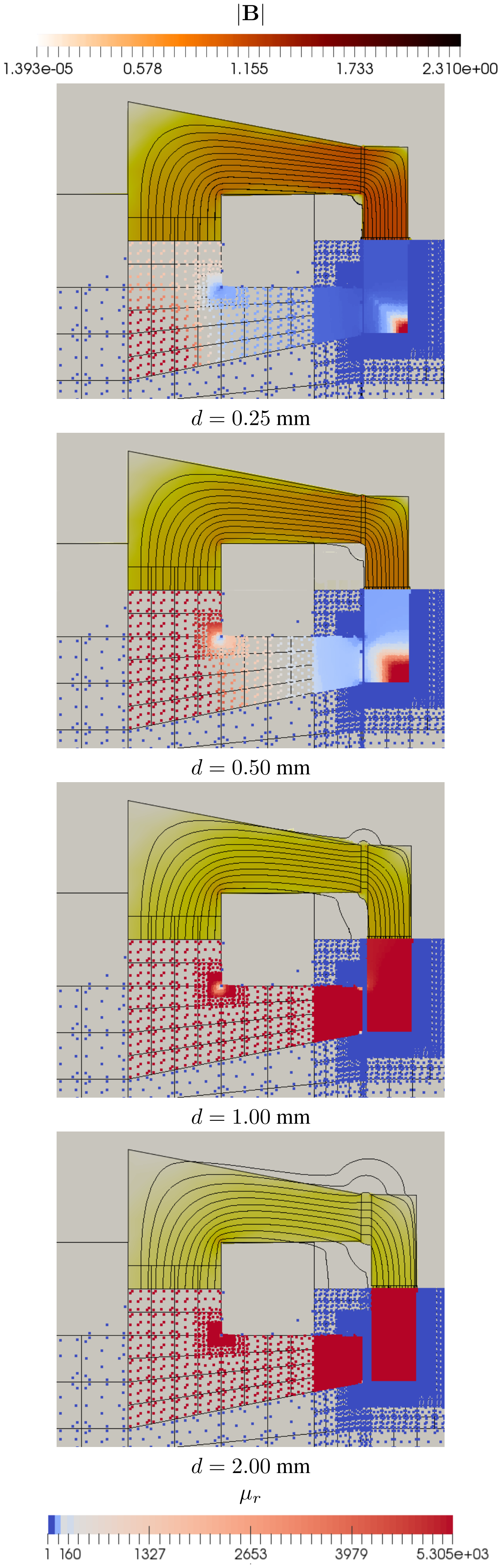
An Interior-Permanent-Magnet (IPM) motor characterized by four poles and 12 stator slots is considered as a further case study, see Fig. 6. The external and the rotor diameters are 68. mm and 30. mm, respectively, while the air-gap width is 0.5 mm. The permanent magnet exhibits a radial magnetization with remanent field equal to $\mathbf{B}_0 = 1.$ T and coercive field equal to $\mathbf{H}_c = 7.957\,10^5$ Am^{-1}.

The μ_r-$\mathbf{H}$ curve featuring the laminated magnetic core of rotor and stator is shown in Fig. 7: the saturation effect is likely to appear inside the small magnetic bridges located in the rotor region between adjacent magnets [12].

In this example we will validate the usage of the projection operators to compute the cogging torque, i.e., the torque acting on the rotor when the three-phase current in the rotor slots is zero (no-load operation). Such quantity is crucial to design a permanent magnet motor since it takes into account the tendency of the permanent magnet axis to align with the direction that corresponds to the minimum energy stored in the motor. Although its value is substantially lower than the running torque due to the on-load current, cogging torque could be responsible of annoying vibrations occurring during the normal on-load operation of a small motor like the one here considered.

The cogging torque has to be computed considering different angular positions of the rotor. To generate *all* these meshes, we can exploit the flexibility of VEM in gluing meshes. Indeed, one possible strategy is to generate stator and rotor mesh separately and then glue them together with the desired angle, see Fig. 8. As shown in Fig. 8 (a), we generate the mesh of each piece of the permanent magnet motor separately and then we glue them together.

To model the permanent magnet we proceed in two different ways according to the VEM scheme used. For the potential formulation we exploit the standard current sheet model, while for



$d = 0.25$ mm

$d = 0.50$ mm

$d = 1.00$ mm

$d = 2.00$ mm

Fig. 5. C-core example: the vector field $\mathbf{B}$, on top of each figure, and the magnetic permeability μ_r at the quadrature points, on bottom of each figure.

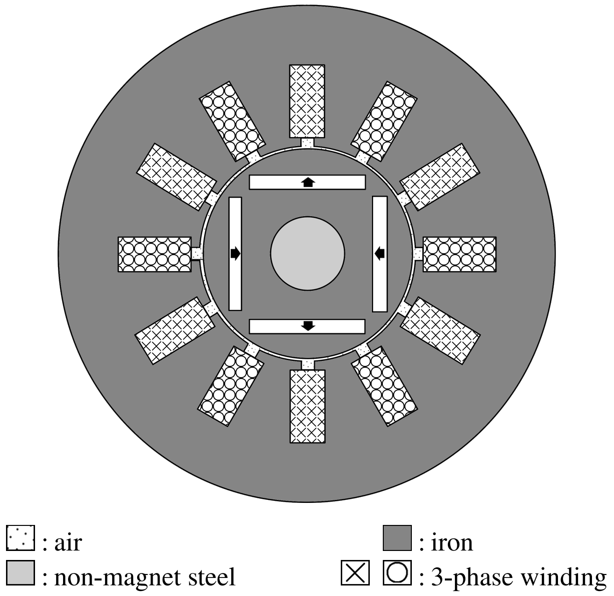


Fig. 6. Permanent magnet motor: the geometry of the 4-pole motor taken into account.

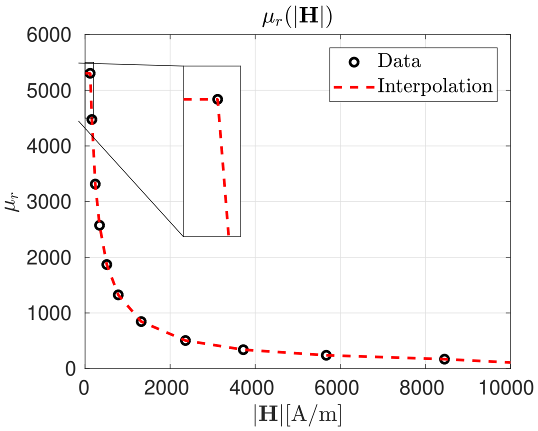


Fig. 7. Permanent magnet motor: $\mu_r - H$ curve used.

Kikuchi we follow the idea proposed in [17]. More specifically we modify the second equation of the variational formulation to take into account the presence of a permanent magnet, i.e., starting from the decomposition of the induction field $\mathbf{B}$

$$\mathbf{B} = \mu\mathbf{H} + \mathbf{B}_0 \,,$$

where $\mathbf{B}_0$ is the remanent field, we substitute

$$\int_\Omega \nabla q \cdot \mu\mathbf{H}\, \mathrm{d}\Omega = 0 \,,$$

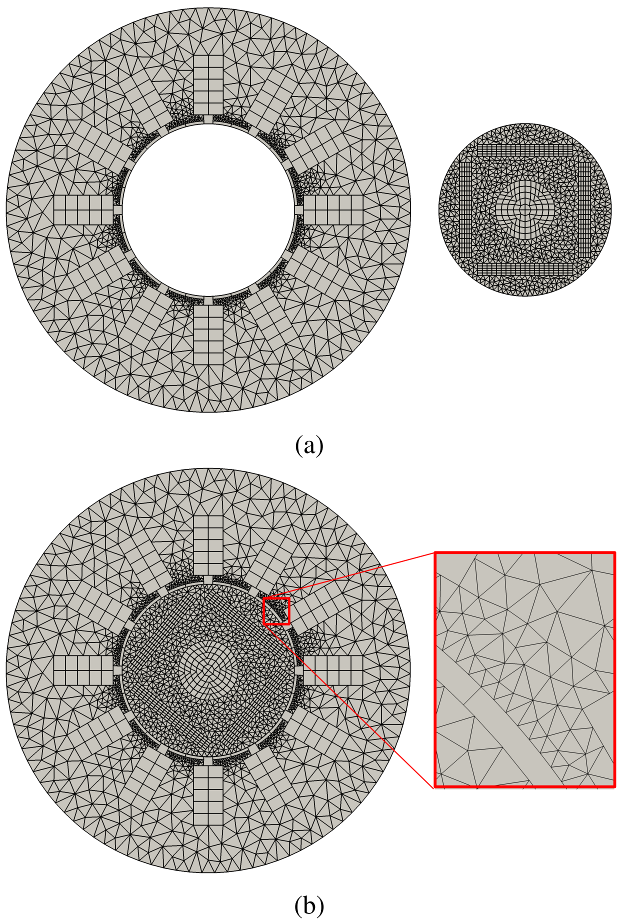
with

$$\int_\Omega \nabla q \cdot \mu\mathbf{H}\, \mathrm{d}\Omega = -\int_\Omega \nabla q \cdot \mathbf{B}_0\, \mathrm{d}\Omega \,,$$

where we recall that q is a Lagrange multiplier variable. We refer the reader to [17] for more details about such model.

Now we proceed with the computation of the cogging torque. The accurate field models provided by the projection operators of both Kikuchi and potential formulations allow us to compute the torque. In Fig. 9 we show the induction map for a specific angle, 21°, when the motor current is zero, obtained with the Kikuchi formulation; similar map holds for the Potential formulation.

To compute such quantity, we use the Maxwell stress tensor approach considering a cylindrical surface co-axially located with respect to the rotation axis as the integration surface accordingly. In Fig. 10 we show the torque-angle curve for step equal to 1°.


(a)

(b)

Fig. 8. Permanent magnet motor: (a) stator and rotor mesh, (b) mesh glued with a detail of the hanging-nodes generated by the gluing procedure.

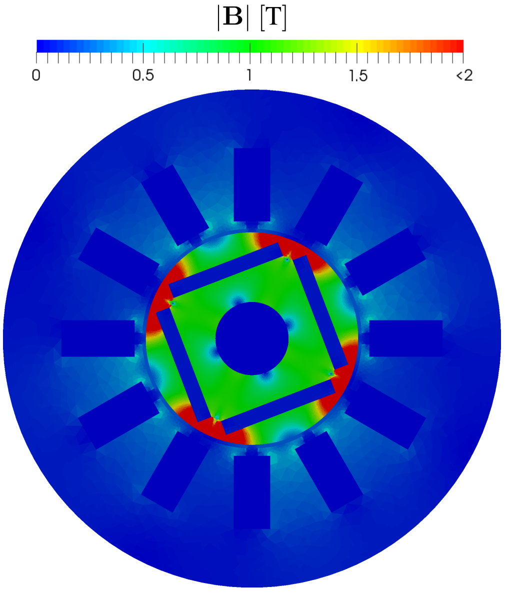


Fig. 9. Permanent magnet motor: magnetic induction map with an angle of 21°.

As expected, the torque period is equal to 30°, indeed

$$360^\circ/\mathrm{LCM}(4, 12) = 30^\circ \,,$$

where LCM is the least common multiple operator and 4 and 12 are the number of permanent magnets and slots, respectively. Moreover, it exhibits zero mean value over the period. Once again there is a good agreement between Kikuchi and potential virtual element formulations.

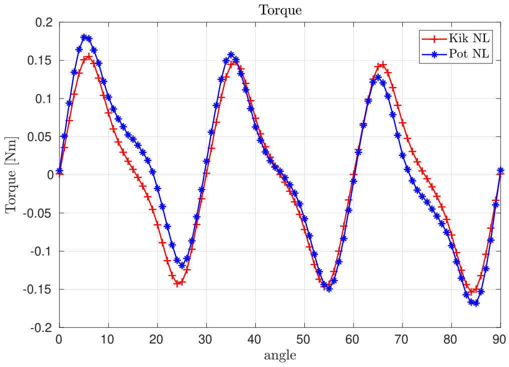


Fig. 10. Permanent magnet motor: values of torque for different position of the rotor.

C. Optimal shape design of an electromagnet

In this section we propose an innovative technique for handling the shape variation of a magnet. More specifically we focus on the optimal desing of a magnet for applications in clinical hypethermia.

Actually, magnetic fluids have a good potential for clinical hyperthermia due to their capability of concentrating power. In fact, the heat generated by magnetic nanoparticles suspended in a fluid depends at least on the squared value of an applied time-harmonic field; so, nanoparticles excited by an AC field can be exploited to treat tumoral tissues [1-4]. Actually, experiments with Magnetic Fluid Hyperthermia (MFH) have shown that human tumour cells are homogeneously inactivated after AC magnetic field excitation of magnetic fluids injected in the tumour region [5]. Since the power density generated by nanoparticles is a function of the applied magnetic field [4-6], field uniformity and field strength are a twofold prerequisite for homogeneously heating the tumour tissue at a therapeutic temperature, without damaging the neighbouring healthy tissues.

A typical device for clinical MFH is characterized by a magnetic core made of ferrite and it exhibits three limbs. Two series-connected current-carrying coils are wound on the central limb which has a wide air-gap, where the patient is accommodated during the treatment.

A quarter of the model geometry here considered is shown in Fig. 11: a ferrite core fills in region $\mathcal{F}$, while an air-gap 30 cm high and 20 cm long incorporates the region of interest $\Gamma_{\mathcal{I}}$, i.e., a path along which the degree of uniformity of flux density is controlled. The non-linear $\mathbf{B}$-$\mathbf{H}$ curve of the ferrite considered in the model exhibits an initial value of relative permeability equal to 1800, and a saturation flux density of 490 mT. The complementary domain includes the winding cross-section $\mathcal{J}$, which is composed of 16 turns and carries a sinusoidal current of 150 Arms at 100 kHz, and an air region, $\mathcal{A}$.

The design challenge is to shape the magnetic pole in such a way that the prescribed field takes place in the region of interest, see Fig. 11. Such problem is usually reformulated as an inverse problem solved by means of a numerical optimisation technique. Indeed, a suitable functional is defined to measure the discrepancy between actual and the prescribed flux density in the region of interest, $\Gamma_{\mathcal{I}}$. In this case the unknown is the pole shape, $\Gamma_{\mathcal{P}}$, which minimises such functional.

In this preliminary study we do not solve explicitly an optimization problem, but we show how to exploit the flexibility in mesh generation of VEM to tackle such procedure. In Fig. 12 we show the mesh of the computational domain. To generate such mesh we exploit the possibility of gluing meshes. We create four pieces for each region: two triangular meshes for the air and the magnet, and three structured quadrilateral meshes for the current, the region of interest and the optimising region, see Fig. 11.

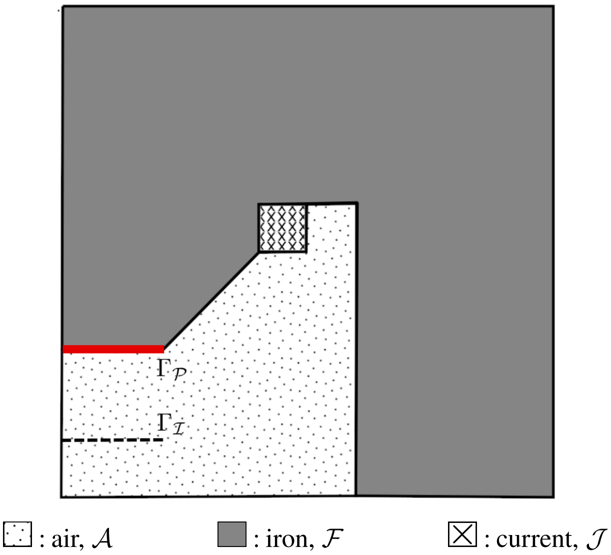


Fig. 11. Optimal shape design: domain to consider. We highlight in red the shape-varying line, $\Gamma_{\mathcal{P}}$, and with a dotted line the path where we compute the y-component of the $\mathbf{B}$ field, $\Gamma_{\mathcal{I}}$.

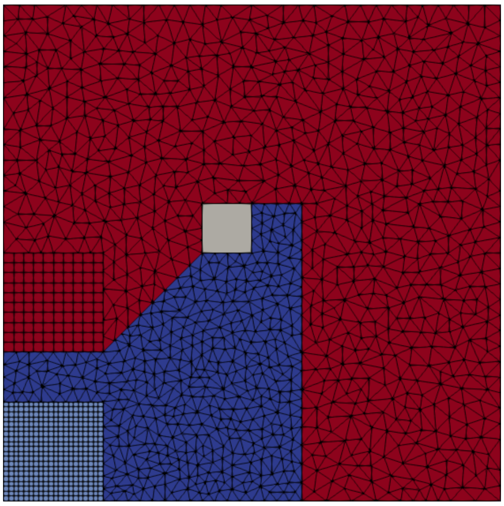


Fig. 12. Optimal shape design: mesh of the computational domain.

The key idea of the VEM optimisation procedure is to start from a uniform mesh composed, e.g. of quadrilateral elements regularly spaced, see Fig. 13 left, and then introduce suitable cuts by segments whose end-points are located at the vertexes or on edges of the background mesh, see Fig. 13 right. Consequently the original quadrilateral element is cut in a pair of polygons that are naturally handled by VEM and there is no need of sub-triangulate or re-build the mesh. Such piece-wise segment represents the air-ferrite boundary that is the magnet pole we would like to modify in order to obtain the prescribed field at the region of interest.

In Fig. 15 we show the y-component along the reference line $\Gamma_{\mathcal{I}}$ highlighted in Fig. 11 for both designs shown in Fig. 13 and in Fig. 14: we show the vector $\mathbf{B}$ field map for both configurations.

Also in this case we exploit the projection operator to compute the magnetic field. As it was expected from the study in [22], the design (b) of Fig. 13 is more uniform with respect to the one provided by the design (a).

From the results in Fig. 15 and the mesh represented in Fig. 13 we

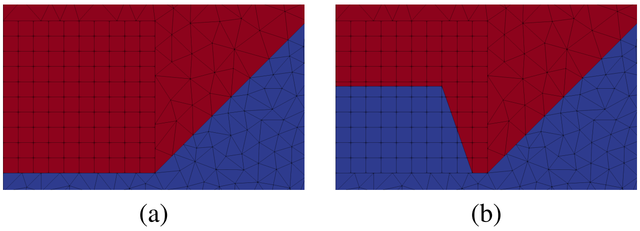

(a) (b)

Fig. 13. Optimal shape design: on the left the initial profile of the magnet, on the right a new configuration.

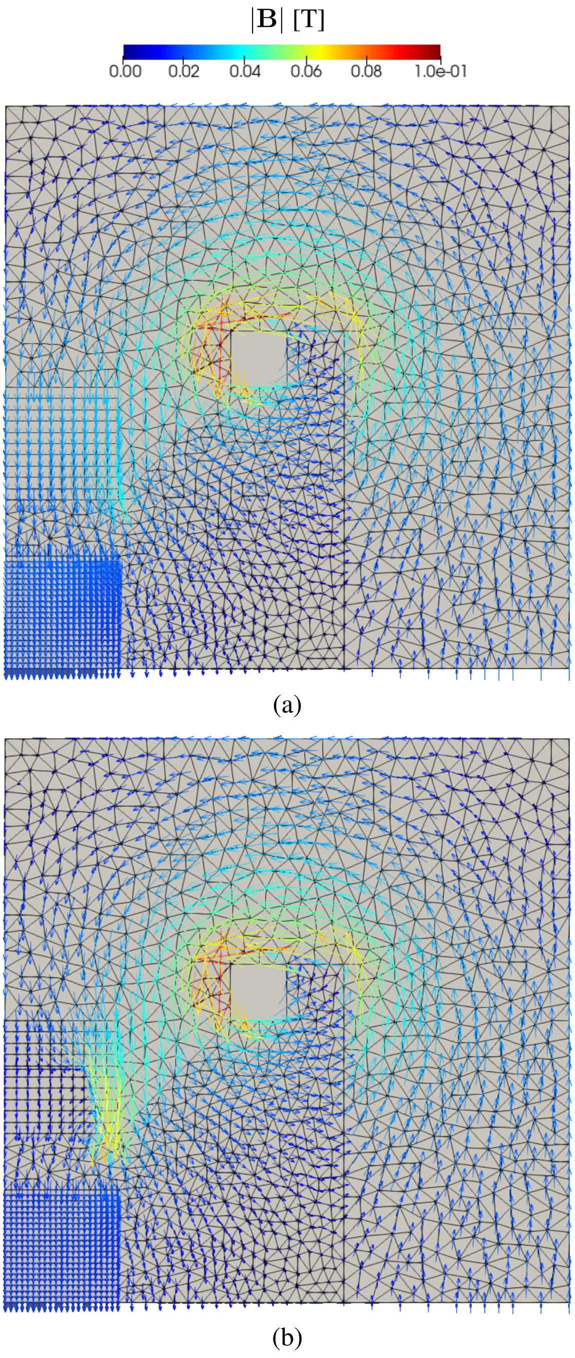
infer that an optimisation procedure is possible with VEM and that the presence of arbitrary shaped polygons do not corrupt the whole numerical simulation. However, the proposed strategy is really flexible and overcomes the common issue of a FEM shape optimisation problem.

Indeed, in standard FEM based approaches, all the problem of handling a deformed triangular meshes arises [21]. Since the elements may become too large it has to be split to have the same discrete approximation of the field. The triangles can be split in three new triangles [18] without any update of neighbouring elements or, alternatively, in four new triangles using the edge midpoints as new nodes and updating the neighbouring triangles. In both cases too stretched triangles can be generated so the whole simulation my suffer for the presence of such badly-shaped elements. However, in the new approach here proposed the background mesh determines the detail of the $\mathbf{B}$ field and the added segments do not generate elements with larger diameter by construction. Moreover such fine structured mesh is *independent* on all the other regions. Indeed one can make a structured very fine mesh and glue with the mesh of the whole domain without caring about grading element size. Finally, about the FEM issue related to badly shaped elements, it is already provided in many papers that VEM is more robust with respect element distortion so the presence of stretched and deformed elements do not corrupt the whole simulation, see Fig. 13.

A second FEM approach is proposed in [22]. Here the authors presents two strategies for updating the profile of the pole shape were implemented. In the *fixed-grid strategy* the region to synthesize is subdivided into several square sub-regions, whose material property can be attributed to either air or ferrite. This way, the unknown profile is approximated by means of a staircase-shaped line, and a structured mesh discretizes the synthesis region: the material permeability (either ferrite or air) is updated, and the field analysis is repeated. This way, the FE mesh does not change, while the distribution of magnetic permeability changes, i.e., in this approach we have a fixed topology but variable material. Alternatively, in the *variable-grid strategy*, the unknown profile is approximated by means of a polygonal line defined by a number of moving nodes, and the FE mesh is accordingly regenerated. The advantage of the proposed VEM approach with respect to the variable-grid strategy is that we do not need to regenerate the whole mesh: we have only to *update* the position of the piece-wise segment inside a specific region. This procedure is localized and it is much easier than a mesh generation procedure. Moreover, it produces a piece-wise segment by construction and we do not have a ladder-like profile as it may happen in the fixed-grid strategy.

D. A 3D analysis of a cylindrical electromagnet

Now we move to the three dimensional setting. In the previous subsection we do not show how to modify the virtual element



(a)

(b)

Fig. 14. Optimal shape design: field $\mathbf{B}$ arrow maps for the designs (a) and (b).

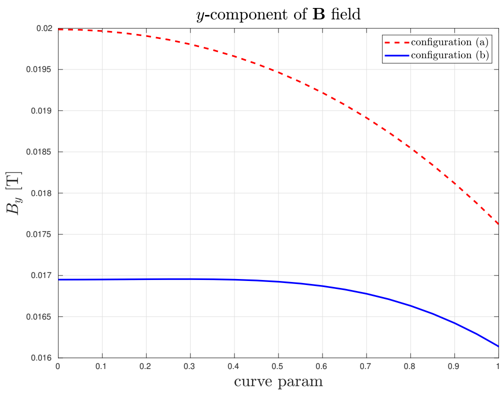


Fig. 15. Optimal shape design: y-component of the $\mathbf{B}$ field along the path $\Gamma_{\mathcal{I}}$ for the designs (a) and (b).

spaces in 3D, we refer the reader to [3] and to [9, 14] for the potential and Kikuchi formulations, respectively.

We consider a typical benchmark proposed in [10, 11]. The geometry consists of a ferromagnetic cylindrical core, $\mathcal{C}$, surrounded by a toroidal coil with a rectangular cross section, $\mathcal{T}$, with air, $\mathcal{A}$, around these two structures. In Fig. 16 we show a meridian cross section of the domain where we specify the dimensions of the cylindrical core, the toroidal coil and the bounding box of the domain.

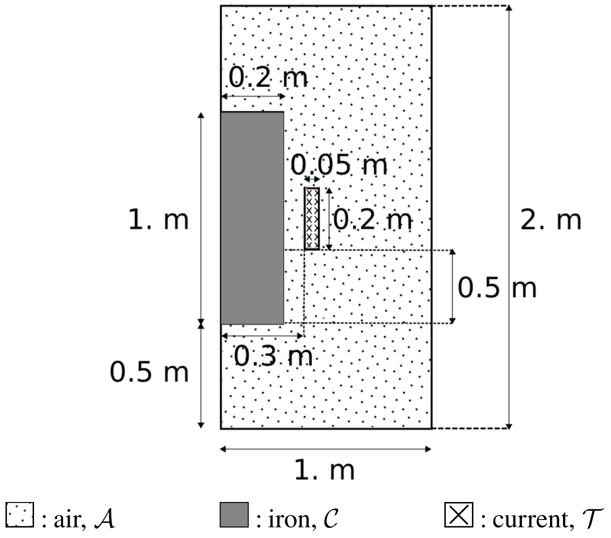


Fig. 16. Cylindrical electromagnet: meridian cross section of the computational domain taken into account.

The aim of this problem is to find the magnetic flux of this configuration when a constant current is carried by the toroidal coil. We have a constant current of 1 A along the toroidal coil. Then, we set the following values of the relative magnetic permeability: $\mu_{\mathcal{T}} = 1.0$ for the coil, $\mu_{\mathcal{C}} = 1.0\,10^4$ in the the ferromagnetic cored and $\mu_{\mathcal{A}} = 1.0$ for the air around.

To solve such problem we use only the Kikuchi VEM formulation. Then, to build the computational domain we extrude a two-dimensional mesh constrained to the circles which defines the cylindrical core and the toroidal coil. The resulting mesh is shown in Fig. 17.

In Fig. 18 we compare the solution provided in [10], the pictures on the right, and the one obtained by the proposed method, the ones on the left. More specifically, in Fig. 18, we show the modulus of the magnetic flux density. This is only a qualitative comparison between the field provided by VEM and the one in [10], but it is clear that the proposed method is aligned with the FEM result obtained in the reference book.

Finally, to have a more quantitative validation of this example, we compute the magnetic energy

$$W := \int_{\mathcal{D}} \mathbf{B} \cdot \mathbf{H} \,\mathrm{dx} = \int_{\mathcal{D}} \mu |\mathbf{H}|^2 \,\mathrm{dx}\,,$$

for each sub-domain $\mathcal{C}$, $\mathcal{T}$ and $\mathcal{A}$, exploiting the assumption of linear material. We consider a sequence of three nested meshes of the domain at hand to verify the convergence rate of such energies. We refer to these meshes as mesh1, mesh2 and mesh3, the first mesh being the one associated with the larger mesh-size. Moreover, since we can not have a direct access to the field $\mathbf{H}$, we exploit once again the projector operator to compute such integral.

Since we do not have the exact solution of this problem, we proceed as [11], i.e. we consider as exact values of the magnetic energies the values obtained by the *FLUX2D* code on a very fine mesh. The numerical solution of *FLUX2D* follows a scalar potential formulation and it exploits the symmetry of the domain via a two-dimensional cylindrical coordinate system.

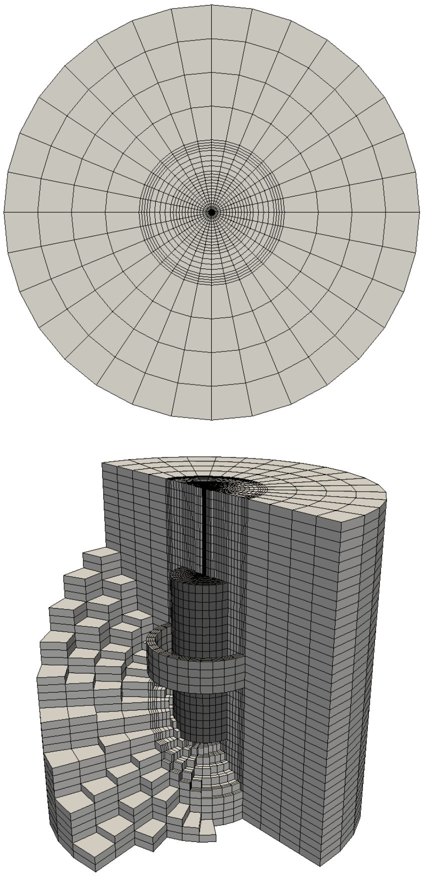


Fig. 17. Cylindrical electromagnet: on top the two-dimensional mesh which we extrude to make the three-dimensional one. On the bottom a clip of the 3D mesh where we highlight the cylindrical core and the toroidal coil.

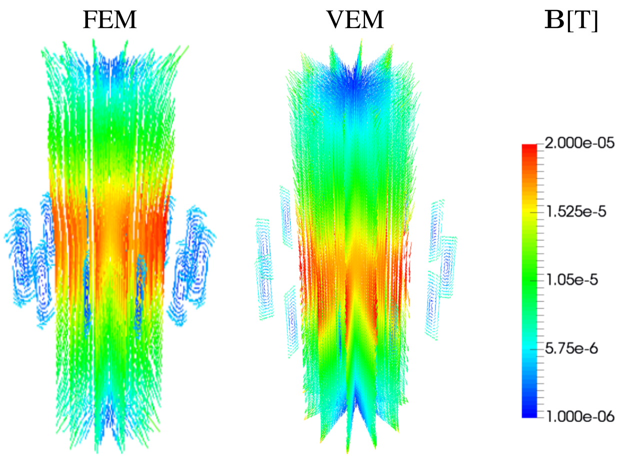


Fig. 18. Cylindrical electromagnet: on the left we show a picture from [10] representing the modulus of the magnetic flux density in a section of the cylindrical coil. On the right we show the same data obtained with the proposed approach.

The data are collected in Table IV. The method behaves as expected and the values of W on each sub-domain converge to the ones provided by *FLUX2D* .

VII. Conclusions

The paper summarizes the theoretical aspects behind the Virtual Element Method and presents several results after its application in linear and non-linear magnetostatics; both potential and

W [J]		$\mathcal{A}$	$\mathcal{C}$	$\mathcal{T}$
FLUX2D		9.09e-07	4.73e-10	3.61e-08
VEM	mesh1	9.70e-07	7.54e-10	3.29e-08
	mesh2	9.22e-07	5.53e-10	3.81e-08
	mesh3	9.11e-07	4.98e-10	3.75e-08

TABLE IV. CYLINDRICAL ELECTROMAGNET: COMPARISON BETWEEN THE VALUES OF W COMPUTED VIA *FLUX2D* AND THE PROPOSED VEM METHOD.

Kikuchi mixed formulation are considered. In particular, the results from the VEM and FEM approximation of the potential formulation are compared. Moreover, the fields B and H are approximated by suitable projection operators which are also exploited to compute some derived quantities like forces and torques via the Maxwell's stress tensor.

Notably, the flexibility of VEM in mesh generation for complex computational domains is tested: it is shown that such flexibility can be further exploited combining virtual and finite element approaches. Indeed, one can use local spaces over elements characterized by hanging nodes and polygonal geometries combined with standard finite elements over triangles and squares. Eventually, an original technique for handling the shape variation in a problem of optimal shape design is presented. The promising results in both mesh generation and field analysis encourage further investigations on magnetic field synthesis.

REFERENCES

[1] Altair Flux, Altair Engineering. https://www.altair.com/flux/.

[2] L. Beirão da Veiga, F. Brezzi, L. D. Marini, and A. Russo. The hitchhiker's guide to the virtual element method. *Math. Models Methods Appl. Sci.*, 24(08):1541–1573, 2014.

[3] L. Beirão da Veiga, F. Dassi, and A. Russo. High-order virtual element method on polyhedral meshes. *Comput. Math. Appl.*, 74(5):1110–1122, 2017.

[4] L. Beirão da Veiga, F. Brezzi, A. Cangiani, G. Manzini, L. D. Marini, and A. Russo. Basic principles of virtual element methods. *Math. Models Methods Appl. Sci.*, 23(1):199–214, 2013.

[5] L. Beirão da Veiga, F. Brezzi, L. D. Marini, and A. Russo. Serendipity nodal VEM spaces. *Comp. Fluids*, 141:2–12, 2016.

[6] L. Beirão da Veiga, F. Brezzi, L. D. Marini, and A. Russo. Serendipity face and edge VEM spaces. *Rend. Lincei Mat. Appl.*, 28(1):143–180, 2017.

[7] L. Beirão da Veiga, F. Brezzi, L. D. Marini, and A. Russo. Virtual element approximations of the vector potential formulation of magnetostatic problems. *SMAI-Journal of computational mathematics*, 4:399–416, 2018.

[8] L Beirão da Veiga, F Brezzi, LD Marini, and A Russo. Virtual element method for general second-order elliptic problems on polygonal meshes. *Mathematical Models and Methods in Applied Sciences*, 26(04):729–750, 2016.

[9] L. Beirão da Veiga, F. Brezzi, F. Dassi, L.D. Marini, and A. Russo. Lowest order virtual element approximation of magnetostatic problems. *Computer Methods in Applied Mechanics and Engineering*, 332:343 – 362, 2018.

[10] A. Bermúdez, D. Gómez, and P. Salgado. *Mathematical Models and Numerical Simulation in Electromagnetism*, volume 74 of *Unitext*. Springer, 2014.

[11] A. Bermúdez, R. Rodríguez, and P. Salgado. A finite element method for the magnetostatic problem in terms of scalar potentials. *SIAM Journal on Numerical Analysis*, 46(3):1338–1363, 2008.

[12] N. Bianchi. *Electrical machine analysis using finite elements*. CRC press, 2017.

[13] A Siemens Business. MagNet, version 7.9, mentor graphics. https://www.mentor.com/products/mechanical/magnet/.

[14] L. Beirão da Veiga, F. Brezzi, F. Dassi, L. D. Marini, and A. Russo. A family of three-dimensional virtual elements with applications to magnetostatics. *SIAM Journal on Numerical Analysis*, 56(5):2940–2962, January 2018.

[15] L. Beirão da Veiga, F. Brezzi, F. Dassi, L.D. Marini, and A. Russo. Virtual element approximation of 2D magnetostatic problems. *Computer Methods in Applied Mechanics and Engineering*, 327:173 – 195, 2017. Advances in Computational Mechanics and Scientific Computation—the Cutting Edge.

[16] L. Beirão da Veiga, F. Brezzi, F. Dassi, L.D. Marini, and A. Russo. Lowest order virtual element approximation of magnetostatic problems. *Computer Methods in Applied Mechanics and Engineering*, 332:343–362, April 2018.

[17] F. Dassi, P. Di Barba, and A. Russo. Virtual element method and permanent magnet simulations: potential and mixed formulations. To appear in IET-SMT.

[18] Antoine Henrot and Michel Pierre. *Shape variation and optimization*. 2018.

[19] F. Kikuchi. Mixed formulations for finite element analysis of magnetostatic and electrostatic problems. *Japan J. Appl. Math.*, 6:209–221, 1989.

[20] J.-C. Nédélec. A new family of mixed finite elements in $\mathbf{R}^3$. *Numer. Math.*, 50(1):57–81, 1986.

[21] Olivier Pironneau. Optimal shape design for elliptic systems. In *System Modeling and Optimization*, pages 42–66. Springer, 1982.

[22] Slawomir Wiak, P. Di Barba, F. Dughiero, and E. Sieni. Non-parametric optimal shape design of a magnetic device for biomedical applications. *COMPEL - The international journal for computation and mathematics in electrical and electronic engineering*, 31(5):1358–1367, 2012.

AUTHORS NAME AND AFFILIATION

Franco Dassi, Dipartimento di Matematica e Applicazioni, Università di Milano–Bicocca, Via Cozzi 55, I-20153, Milano, Italy, franco.dassi@unimib.it

Paolo Di Barba, Dipartimento di Ingegneria Industriale e dell'Informazione, Via Adolfo Ferrata 1, 27100 Pavia, Italy, paolo.dibarba@unipv.it

Alessandro Russo, Dipartimento di Matematica e Applicazioni, Università di Milano–Bicocca, Via Cozzi 55, I-20153, Milano, Italy, IMATI-CNR, Via Ferrata 5/A, 27100 Pavia, Italy, alessandro.russo@unimib.it.