Journal of Computational Physics xxx (2010) xxx-xxx



Contents lists available at ScienceDirect

Journal of Computational Physics



journal homepage: www.elsevier.com/locate/jcp

The mimetic finite difference method for the 3D magnetostatic field problems on polyhedral meshes

K. Lipnikov^{a,*}, G. Manzini^{b,c}, F. Brezzi^d, A. Buffa^b

^a Los Alamos National Laboratory, Theoretical Division, MS B284, Los Alamos, NM 87545, USA

^b Istituto di Matematica Applicata e Tecnologie Informatiche (IMATI) – CNR, via Ferrata 1, I-27100 Pavia, Italy

^c Centro di Simulazione Numerica Avanzata (CeSNA)-IUSS Pavia, v.le Lungo Ticino Sforza 56, I-27100 Pavia, Italy

^d Istituto Universitario di Studi Superiori, Pavia, Italy

ARTICLE INFO

Article history: Received 18 March 2010 Received in revised form 30 August 2010 Accepted 6 September 2010 Available online xxxx

Keywords: Div-curl equations Magnetostatics Mimetic finite differences Polyhedral mesh

ABSTRACT

We extend the mimetic finite difference (MFD) method to the numerical treatment of magnetostatic fields problems in mixed *div-curl* form for the divergence-free magnetic vector potential. To accomplish this task, we introduce three sets of degrees of freedom that are attached to the vertices, the edges, and the faces of the mesh, and two discrete operators mimicking the curl and the gradient operator of the differential setting. Then, we present the construction of two suitable quadrature rules for the numerical discretization of the domain integrals of the *div-curl* variational formulation of the magnetostatic equations. This construction is based on an *algebraic consistency condition* that generalizes the usual construction of the inner products of the MFD method. We also discuss the linear algebraic form of the resulting MFD scheme, its practical implementation, and discuss existence and uniqueness of the numerical solution by generalizing the concept of *logically rectangular or cubic meshes* by Hyman and Shashkov to the case of unstructured polyhedral meshes. The accuracy of the method is illustrated by solving numerically a set of academic problems and a realistic engineering problem.

Published by Elsevier Inc.

1. Introduction

Mimetic discretizations for the numerical resolution of Partial Differential Equations (PDE) have been proposed to the research community since the beginning of the eighties. These methods were originally aimed at preserving the fundamental properties of physical and mathematical models such as conservation laws, solution symmetries and positivity, as well as some fundamental identities and theorems of vector and tensor calculus, e.g., Gauss–Green's identities. Among major advantages that mimetic formulations offer is the possibility of using polyhedral meshes, which may be more efficient in partitioning the computational domain. In fact, a complex three-dimensional (3D) geometry is easily modeled with mixed types of mesh elements such as pentahedrons, prisms and tetrahedrons that can be obtained by collapsing some of the elements of a structured hexahedral or prismatic mesh to conform and adapt it to the physical domain. Polyhedral meshes appear often in numerical applications using Lagrangian meshes, moving mesh methods, and mesh reconnection methods, where elements may develop non-convex shapes and have non-planar faces due to the specifics of the flow dynamics and adaptive mesh refinement. The use of polyhedral meshes relaxes the requirement of maintaining the mesh conformity, which may

* Corresponding author. Tel.: +1 505 667 1719; fax: +1 505 665 5757.

E-mail addresses: lipnikov@lanl.gov (K. Lipnikov), Marco.Manzini@imati.cnr.it (G. Manzini), brezzi@imati.cnr.it (F. Brezzi), annalisa@imati.cnr.it (A. Buffa).

0021-9991/\$ - see front matter Published by Elsevier Inc. doi:10.1016/j.jcp.2010.09.007

K. Lipnikov et al./Journal of Computational Physics xxx (2010) xxx-xxx

result in an excessive refinement, and of treating possible hanging nodes in some special way. In fact, locally refined meshes and non-matching meshes can be treated as conformal polyhedral meshes with degenerate elements, i.e., elements with a zero angle between adjacent edges or faces.

These issues motivate the development of discretization methods in variational form like finite elements (FE) suited to general polygonal and polyhedral meshes. Various approaches to extend FE methods to non-traditional elements (pyramids, polyhedrons, etc.) have been developed over the last decade, see, e.g., [25,33,35,46,47]). A straightforward approach is the construction of a set of basis functions for a general polygonal element. However, this approach may be difficult to be pursued as it requires an extensive geometrical analysis. For instance, in [33] an auxiliary simplicial partition is used in each polygonal element to simplify the construction of the FE basis functions.

The mimetic finite difference (MFD) method presented, analyzed and tested in [44,31,12–14,10] combines the analytical power of FE methods with the flexibility provided by polygonal and polyhedral meshes. The MFD method uses a suitable discrete version of the Gauss–Green relations, i.e., a discrete integration by parts formula, to build the stiffness and mass matrices for the scheme's unknowns. Since no explicit representation of the approximate solution through basis functions is required inside mesh elements, practical implementation of the MFD method is relatively simple for polygonal and polyhedral meshes. The MFD method has been successfully employed in the numerical solution of linear diffusion problems [10,13,29], convection–diffusion equations [17], electromagnetic problems [29], linear elasticity equation [3] and for modeling fluid flows [1,15,36]. The original MFD method is a low-order method, but miscellaneous approaches were developed towards higher-order methods; cf. [43,6,26,4]. We also mention the development of a posteriori estimators for diffusion problems [2,5] and the post-processing methodology analyzed and tested in [16].

Other successful approaches for elliptic problems on unstructured polygonal and polyhedral meshes that are related to the mimetic methodology are in the large family of finite volume techniques. For example, the Discrete Duality Finite Volume (DDFV) formulation [27,19,20] uses definitions of discrete operators on staggered meshes that are connected by duality relations. We also mention the gradient type scheme of [24] and the mixed-finite volume scheme proposed in [22] The connection between these methods and the MFD method for diffusion problem in mixed form is investigated in [23].

In this work, we propose and investigate how to extend the MFD method to the numerical treatment of magnetostatic field problems. To this purpose, Maxwell's equations for the steady magnetic field are reformulated in the *div-curl* mixed form through the introduction of the magnetic vector potential **u** satisfying the Coulomb gauge, i.e., the solenoidal condition $\operatorname{div}(\mathbf{u}) = 0$ [34, Chapter IV]. As pointed out in [9], the mimetic discretizations are intimately connected to the geometric structure of Maxwell's equations. Several other papers in the literature investigate this concept. It is worth mentioning the pioneering work on mimetic discretizations that was carried out in [29–32] in the framework of logically rectangular and logically cubic meshes. In the finite element context, the seminal paper is surely that on Whitney forms [8]. As an extensive overview of the results presented in the literature is beyond the scope of the present work, we refer the interested reader to [38] for a detailed treatment of such topics. A significant contribution to the numerical discretizations based on the mimetic approach and the geometric structure of differential forms is found in [9]. In this work, the mimetic degrees of freedom are a discretization of *co-chains*, and inner products are derived from the introduction of a lifting operator into a discrete consistency condition that yields a numerical integration by parts formula. The papers [7,37] are significant contributions to the literature based on algebraic topology concepts towards a unified formulation of finite element, finite difference and finite volume methods. It is also worth mentioning the seminal paper [49], which is a central-difference scheme in space and time originally devised for time-dependent Maxwell equations, and the covolume methods developed in [40-42] for div-curl systems. A proof of the second-order convergence is given that does not rely on connections to variational formulations. These covolume methods require two orthogonal meshes to approximate the electric and magnetic fields. To this purpose, the dual Delaunay-Voronoi diagram is a natural choice; for example, in three dimensions, every edge of a Voronoi diagram is orthogonal to the correspondent face of the Delaunay triangulation, and vice verse. A recent extension to arbitrary two-dimensional (2D) meshes, which is based on the DDFV approach and does not require any orthogonality property between the primal and the dual mesh, is found in [21]. We finally point out that, on tetrahedral or on hexahedral grids the mixed finite element approach, based on the Nédélec edge elements [39], will surely be one of the many possible variants allowed in our approach, in the spirit of [28]; but the extension of mixed finite elements to a general decomposition is definitely cumbersome.

In the above works, we can find many of the ideas also considered in our approach. More precisely, the mimetic formulation that we investigate in this paper relies upon a set of degrees of freedom that are topologically attached to the vertices, the edges, and the faces of a mesh. Then, we define two discrete operators mimicking curl and gradient that act on edge and node degrees of freedom, respectively. The next ingredient is provided by two suitable quadrature rules for the numerical discretization of volume integrals on the computational domain, which make use of edge and face degrees of freedom. Using the discrete curl and gradient operators and these quadrature rules, we provide a numerical discretization of the bilinear forms in the variational formulation of magnetostatic equations that can be seen as a variant of the one proposed in [9]. The derivation of the quadrature rules that are mentioned above is, thus, the crucial step in the formulation of this scheme and is discussed in detail in the paper. This derivation is based on an *algebraic consistency condition* that generalizes the usual construction of the inner products in the MFD methods. When this condition is satisfied, each quadrature rule takes the form of a vector–matrix–vector multiplication for the vectors of degrees of freedom, where the matrix is given by formulas provided in the paper. We also mention the paper [48], where an algebraic consistency of mimetic [44] and covolume methods on triangular meshes is introduced and analyzed.

K. Lipnikov et al./Journal of Computational Physics xxx (2010) xxx-xxx

We also present the linear algebraic form of the MFD method and demonstrate theoretically its well-posedness, i.e., existence and uniqueness of the numerical solution. Finally, we illustrate the performance of the method through numerical experiments that show the accuracy of the approximate solution on a set of academic problems and present the application of the method to a realistic engineering problem.

The paper is organized as follows. In Section 2, we discuss the variational form of the magnetostatic field problem. In Section 3, we discuss the derivation of this MFD method. In Section 4, we analyze the well-posedness of the discretization. In Section 5, we present the results of our numerical experiments. In Section 6, we offer final remarks and discuss the perspective for future work.

2. 3D magnetostatic field problem

We are interested in solving the magnetostatic problem

$$\begin{aligned} & \text{curl}(\boldsymbol{H}) = \boldsymbol{J} \quad \text{in } \Omega, \\ & \text{div}(\boldsymbol{\mu}\boldsymbol{H}) = \boldsymbol{0} \quad \text{in } \Omega, \\ & \boldsymbol{H} \times \boldsymbol{n} = \boldsymbol{g}' \quad \text{on } \Gamma \end{aligned}$$
 (1) (2)

for the unknown vector variable **H**, the magnetic field intensity. We assume that **J**, the divergence-free current density, μ , the magnetic permeability *tensor*, and **g**', a vector-valued boundary function, are given. From a physical standpoint, the domain Ω should be the whole space \Re^3 , and the magnetic field should satisfy a radiation condition like $\mathbf{H} \to 0$ at infinity instead of (3). In practice, we assume that Ω is a bounded, simply-connected domain in \Re^3 with the Lipschitz boundary Γ , and replace the radiation condition by the boundary condition (3), where **n** is the unit vector orthogonal to Γ . The tensor coefficient μ may be discontinuous. However, the tangential component of **H** and the normal component of μ **H** are continuous across the possible interfaces of discontinuity of μ .

Condition (2) allows us to introduce the vector potential \boldsymbol{u} such that $\operatorname{curl}(\boldsymbol{u}) = \mu \boldsymbol{H}$. The choice of \boldsymbol{u} is not unique as we can always add the gradient of a scalar function to the vector potential \boldsymbol{u} and leave the relation with \boldsymbol{H} unaltered. To obtain a weak formulation that admits a unique solution we consider the *Coulomb gauge*, which leads to a divergence-free vector potential. More precisely, we require the vector field \boldsymbol{u} to be the solution of the set of equations:

$$\operatorname{curl}(\mu^{-1}\operatorname{curl}(\boldsymbol{u})) = \boldsymbol{J} \quad \text{in } \Omega, \tag{4}$$

$$\operatorname{div}(\boldsymbol{u}) = \boldsymbol{0} \quad \text{in } \Omega, \tag{5}$$

$$\boldsymbol{u} \times \boldsymbol{n} = \boldsymbol{g} \quad \text{in } \Gamma. \tag{6}$$

We derive the variational formulation for this problem in the following steps. First, we introduce the vector Sobolev space

$$H(\operatorname{curl},\Omega) = \left\{ \boldsymbol{v} \in (L^2(\Omega))^3 \quad \operatorname{such} \text{ that } \operatorname{curl}(\boldsymbol{v}) \in (L^2(\Omega))^3 \right\}.$$
(7)

The class of admissible weak solutions for the vector potential is given by

$$H_{\mathbf{g}}(\operatorname{curl},\Omega) = \{ \boldsymbol{\nu} \in H(\operatorname{curl},\Omega) \quad \text{such that } \boldsymbol{\nu} \times \boldsymbol{n} = \boldsymbol{g} \quad \text{on } \Gamma \},$$
(8)

while vector-valued test functions will be taken in space $H_0(\operatorname{curl}, \Omega)$ that is defined by setting g = 0 in (8). We do not explicitly require the vector fields in $H_g(\operatorname{curl}, \Omega)$ to be divergence-free. Instead, we will take into account the solenoidal constraint (5) through the introduction of the Lagrangian multiplier p, which belongs to the scalar Sobolev space

$$H_0^1(\Omega) = \Big\{ q \in L^2(\Omega), \nabla q \in (L^2(\Omega))^3, \text{ with } q = 0 \text{ on } \Gamma \Big\}.$$
(9)

Next, we define the bilinear forms

$$\forall \boldsymbol{u}, \, \boldsymbol{v} \in H(\operatorname{curl}, \Omega) := \int_{\Omega} \mu^{-1} \operatorname{curl}(\boldsymbol{u}) \cdot \operatorname{curl}(\boldsymbol{v}) dV, \tag{10}$$

$$\forall \boldsymbol{v} \in H(\operatorname{curl}, \Omega), \quad \forall \boldsymbol{q} \in H_0^1(\Omega): \quad \mathcal{B}(\boldsymbol{v}, \boldsymbol{q}) := \int_{\Omega} \boldsymbol{v} \cdot \nabla \boldsymbol{q} dV \tag{11}$$

and denote the L^2 inner product between vector fields by:

$$\forall \boldsymbol{u}, \boldsymbol{v} \in (L^2(\Omega))^3: \quad (\boldsymbol{u}, \boldsymbol{v}) := \int_{\Omega} \boldsymbol{u} \cdot \boldsymbol{v} \, dV. \tag{12}$$

It is not difficult to see that the following variational problem:

Find $(\boldsymbol{u}, \boldsymbol{p}) \in H_{\mathbf{g}}(\operatorname{curl}, \Omega) \times H_0^1(\Omega)$ such that :

$$\mathcal{A}(\boldsymbol{u},\boldsymbol{v}) + \mathcal{B}(\boldsymbol{v},\boldsymbol{p}) = (\boldsymbol{J},\boldsymbol{v}) \quad \forall \boldsymbol{v} \in H_{\boldsymbol{0}}(\operatorname{curl},\Omega),$$
(13)

$$\mathcal{B}(\boldsymbol{u},\boldsymbol{q}) = \boldsymbol{0} \quad \forall \boldsymbol{q} \in H^1_0(\Omega). \tag{14}$$

K. Lipnikov et al./Journal of Computational Physics xxx (2010) xxx-xxx

is a variational formulation problem (4)–(6). Indeed, under suitable assumptions on the regularity of μ , the well-posedness of (13) and (14) can be proved in the framework of the classical theory for saddle-point problems [11]. Moreover, if $(\boldsymbol{u}, p) \in H_g(\operatorname{curl}, \Omega) \times H_0^1(\Omega)$ is the solution of (13) and (14), choosing $\boldsymbol{v} = \nabla p \in H_0(\operatorname{curl}, \Omega)$ in (13) we have that $\mathcal{A}(\boldsymbol{u}, \nabla p) = 0$ due to the differential identity $\operatorname{curl} \circ \nabla = 0$ and therefore,

$$\int_{\Omega} \boldsymbol{J} \cdot \nabla \boldsymbol{p} d\boldsymbol{V} = -\int_{\Omega} \boldsymbol{p} \operatorname{div}(\boldsymbol{J}) d\boldsymbol{V} + \int_{\Gamma} \boldsymbol{p} \boldsymbol{n} \cdot \boldsymbol{J} d\boldsymbol{S} = \boldsymbol{0}$$
(15)

since **J** is a divergence-free field and p is zero on the boundary Γ . Thus, Eq. (13) becomes:

$$\mathcal{B}(\nabla p, p) = \int_{\Omega} |\nabla p|^2 dV = 0$$

from which it easily follows that p = 0 in Ω , using the homogeneous Dirichlet condition $p_{ir} = 0$.

3. Mimetic discretization

Let \mathcal{T}_h be a partition of the computational domain Ω into m_P polyhedra, m_F planar faces, m_E straight edges and m_N nodes (also called "vertices"). We denote: the set of faces by F, a face by f and its area by |f|; the set of edges by E, an edge by e and its length by |e|; the set of vertices by \vee and a vertex by \vee . Consistently with this notation, if P is a polyhedron of \mathcal{T}_h , its volume is denoted by |P|. The sub-index h, which labels the mesh \mathcal{T}_h , is the *mesh size*, i.e., the characteristic length of the mesh, and is defined, as $h = \max_{e \in E} |e|$.

Each face and each edge in the mesh is endowed with orientation, fixed once and for all by prescribing a unit tangent vector \mathbf{t}_{e} to the edge e and a unit normal vector \mathbf{n}_{f} to the face f. The mutual orientation of the edge e with respect to the face f is reflected via the number $\sigma_{f,e} = \pm 1$. The positive sign corresponds to a counterclockwise orientation of the edge e when an observer is located at the tip of the vector \mathbf{n}_{f} .

The mimetic finite difference method that we aim to develop in this section is formulated for a family of meshes $\{\mathcal{T}_h\}_h$ with decreasing *h*. The meshes in $\{\mathcal{T}_h\}_h$ may contain very general shaped elements and even non-convex elements are admissible. Nonetheless, a few minimal assumptions are usually imposed on elemental shape to avoid some pathological situations. We assume that each partition in $\{\mathcal{T}_h\}_h$ is conformal, i.e., intersection of any two distinct elements P₁ and P₂ of a given \mathcal{T}_h is either empty, or a few mesh points, or a few mesh edges, or a few mesh faces (two adjacent elements may share more than one edge or more than one face). Following [10], we assume that for each mesh \mathcal{T}_h there exists a sub-partition obtained by decomposing each polyhedron in a uniformly bounded number of tetrahedra, whose union is a conformal and regular mesh in the sense of Ciarlet [18]. We point out that this last assumption only requires to know that such a sub-mesh *exists*, a fact that can be easily verified in most cases, but not to construct it in practical implementations. According to these assumptions, the MFD method can be applied on a wide range of meshes. Finally, we approximate the coefficient μ^{-1} by a constant tensor inside each mesh element.

Let us briefly describe the formal construction of our mimetic discretization. The numerical approximation to problem (13) and (14) requires to discretize scalar and vector functions, which are, respectively, elements of $H_0^1(\Omega)$ and $H(\text{curl}, \Omega)$, the bilinear forms $\mathcal{A}(\boldsymbol{u}, \boldsymbol{v})$ and $\mathcal{B}(\boldsymbol{u}, \boldsymbol{v})$, and the right-hand side integral ($\boldsymbol{J}, \boldsymbol{v}$) of Eq. (13). The bilinear forms $\mathcal{A}(\boldsymbol{u}, \boldsymbol{v})$ and $\mathcal{B}(\boldsymbol{u}, \boldsymbol{v})$, and the right-hand side integral ($\boldsymbol{J}, \boldsymbol{v}$) of Eq. (13). The bilinear forms $\mathcal{A}(\boldsymbol{u}, \boldsymbol{v})$ and $\mathcal{B}(\boldsymbol{u}, \boldsymbol{v})$ involve two differential operators, *curl* and *gradient*, for which a *mimetic* discretization is to be provided.

We begin by introducing the degrees of freedom of our mimetic discretization. Even if we discretize *two* types of fields, i.e., scalar and vector fields, for a reason that will be clear throughout this section we consider *three* different types of *grid functions* (see Fig. 1):

- node functions, defined by one number per mesh vertex;
- edge functions, defined by one number per mesh edge;
- face functions, defined by one number per mesh face.

A node function can be interpreted as the *collection of the values of a scalar function* at mesh vertices. An edge function can be interpreted as *the collection of the values of the tangential component of a vector function* averaged along mesh edges. A face function can be interpreted as *the collection of values of the normal component of a vector function* averaged over mesh faces. Therefore, node functions are discrete representations of scalar fields, while edge and face functions are discrete representations of vector fields. We will make this last statement formally precise throughout the rest of this section. Node, edge, and face functions are at the same time *grid functions*, since they uniquely map grid items like nodes, edges, and faces to real numbers, and *algebraic vectors*, since linear algebraic operations such as matrix–vector multiplication can be performed on them. For example, any $q \in N$ can be interpreted as a *discrete scalar field* as well as a *linear algebraic vector*.

For simplicity of notation, we denote continuous and discrete *scalar* fields by letters in normal font, and continuous and discrete *vector* fields by letters in bold font. Therefore, at the notational level, we make no distinction between fields in the continuum and discrete fields as the nature of these quantities can be determined contextually without any ambiguity. Thus, the symbol q may denote either a spatially dependent scalar function defined on $\mathbf{x} \in \Omega$, or an algebraic vector of degrees-of-freedom associated to mesh items like vertices, edges or faces. Likewise, the symbol \mathbf{v} may denote either a spatially



Fig. 1. Restrictions of the degrees of freedom of a grid function in (a) $\mathcal{V}_{|P}$, (b) $\mathcal{E}_{|P}$, (c) $\mathcal{F}_{|P}$ for a (cubic) polyhedron P (the edge orientation in (b) is arbitrary).

dependent vector-valued function defined on $\mathbf{x} \in \Omega$, or an algebraic vector of degrees of freedom associated to mesh items like vertices, edges or faces. However, we will denote the scalar product between vectors by " $\mathbf{u} \cdot \mathbf{v}$ " when \mathbf{u} and \mathbf{v} are vector fields in the continuum, and by " $\mathbf{u}^{\mathsf{T}}\mathbf{v}$ " when \mathbf{u} and \mathbf{v} are algebraic vectors of degrees of freedom.

We denote the linear space of all possible node functions by \mathcal{N} . Let $q \in H^1(\Omega) \cap C^0(\overline{\Omega})$. Its degrees of freedom in \mathcal{N} , denoted by $q^{\mathcal{N}}$, are the values taken by q at the mesh vertices, i.e., for any $v \in V$ we have that $q^{\mathcal{N}}_{|v} = q(\mathbf{x}_v)$, where \mathbf{x}_v is the position vector of the vertex v. It is also convenient to consider the linear subspace $\mathcal{N}_0 \subset \mathcal{N}$ which is formed by all the node functions whose value is zero at the boundary nodes.

We denote the linear space of edge functions by \mathcal{E} . Let $\boldsymbol{v} \in H(\operatorname{curl}, \Omega) \cap (C^0(\overline{\Omega}))^3$. Its degrees of freedom in \mathcal{E} , denoted by $\boldsymbol{v}^{\mathcal{E}} = (\boldsymbol{v}_{|_{\mathsf{e}}|_{\mathsf{e}}}^{\mathcal{E}})_{\mathsf{e}}$, are given by

$$\forall \mathbf{e} \in \mathsf{E}: \quad \boldsymbol{v}_{|\mathbf{e}}^{\varepsilon} = \frac{1}{|\mathbf{e}|} \int_{\mathbf{e}} \boldsymbol{v} \cdot \boldsymbol{t}_{\mathbf{e}} \, dL. \tag{16}$$

As for node functions, we will find it convenient to consider the linear subspace \mathcal{E}_{g} formed by all the functions $\boldsymbol{v} \in \mathcal{E}$ such that for every boundary edge $e \subset \Gamma$ the corresponding value v_e equals the average on e of the tangential component of the vector \boldsymbol{g} . The linear subspace \mathcal{E}_0 is immediately derived by setting $\boldsymbol{g} = 0$.

We denote the linear space of face functions by \mathcal{F} . Let $\boldsymbol{v} \in H(\operatorname{div}, \Omega) \cap (L^{s}(\Omega))^{3}$, where s > 2. Its degrees of freedom in \mathcal{F} , denoted by $\boldsymbol{v}^{\mathcal{F}} = (\boldsymbol{v}_{\mathrm{if}}^{\mathcal{F}})_{\mathrm{f}}$, are given by:

$$\forall \mathbf{f} \in \mathsf{F}: \quad \boldsymbol{v}_{|\mathsf{f}}^{\mathcal{F}} = \frac{1}{|\mathsf{f}|} \int_{\mathsf{f}} \boldsymbol{v} \cdot \boldsymbol{n}_{\mathsf{f}} \, dS. \tag{17}$$

Throughout the paper we will make use of the *restrictions* of an edge or a face function to special subsets of edges and faces. More precisely, let \boldsymbol{v} denote a vector field defined in the continuum setting on the computational domain Ω , and $\boldsymbol{v}^{\varepsilon} \in \mathcal{E}$ its degrees of freedom defined for all the edges of E. Then,

• $\boldsymbol{v}_{\text{f}}^{\varepsilon} = (\boldsymbol{v}_{\text{e}})_{e \in \partial F}$ denotes the subset of values of $\boldsymbol{v}^{\varepsilon}$ attached to the edges e that form the polygonal boundary of the face f; • $\boldsymbol{v}_{\text{F}}^{\varepsilon} = (\boldsymbol{v}_{\text{e}})_{e \in \partial P}$ denotes the subset of values of $\boldsymbol{v}^{\varepsilon}$ attached to the edges e that form the boundary of the polyhedron P.

On its turn, let $\boldsymbol{v}^{\mathcal{F}} \in \mathcal{F}$ be a face function. Consistently with the previous notation,

• $\boldsymbol{v}_{\mathsf{P}}^{\mathcal{F}} = (v_{\mathsf{f}})_{\mathsf{f} \in \partial \mathsf{P}}$ denotes the subset of values of $\boldsymbol{v}^{\mathcal{F}}$ attached to the faces f that form the boundary of the polyhedron P.

The collection of edge and face restrictions may be given the algebraic structure of a linear space, which is denoted by the self-explanatory symbols $\mathcal{E}_{[r]}, \mathcal{E}_{[P]}$ and $\mathcal{F}_{[P]}$ (see Fig. 1).

Remark 3.1. As pointed out in [7,9], we could complete this construction by the introduction of the linear space \mathcal{P} of cellbased functions, i.e., those functions that are defined by attaching one number to each polyhedron P. Up to a suitable rescaling of the quantities defined above, it is possible to re-interpret the entire setting in terms of *k*-cochains or 3D discrete *k*-forms, where k = 0 corresponds to $\mathcal{N}, k = 1$ to $\mathcal{E}, k = 2$ to \mathcal{F} , and k = 3 to \mathcal{P} . However the investigation of connections and analogies with algebraic topology concepts is beyond the scope of our work.

Using N, \mathcal{E} , and \mathcal{F} , we define the discrete operators \mathcal{GRAD} and \mathcal{CURL} that mimic the two differential operators ∇ and curl, respectively.

• The discrete operator \mathcal{GRAD} maps any discrete scalar field of \mathcal{N} into a discrete vector field of \mathcal{E} , and is defined by:

K. Lipnikov et al./Journal of Computational Physics xxx (2010) xxx-xxx

$$\forall q \in \mathcal{N}: \quad \left(\mathcal{GRAD}(q)\right)_{|\mathsf{e}} = \frac{q_{\mathsf{v}_2} - q_{\mathsf{v}_1}}{|\mathsf{e}|}, \quad \forall \mathsf{e} = (\mathsf{v}_1, \mathsf{v}_2) \in \mathsf{E}, \tag{18}$$

where q_{v_1} and q_{v_2} are the values of the node function q at the vertices v_1 and v_2 , these latters being connected by the oriented edge $e = (v_1, v_2)$ having length |e|.

• The discrete operator \mathcal{CURL} maps any discrete vector field of \mathcal{E} into a discrete vector field of \mathcal{F} , and is defined by:

$$\forall \boldsymbol{\nu} \in \mathcal{E}: \quad \left(\mathcal{CURL}(\boldsymbol{\nu})\right)_{|\mathsf{f}} = \frac{1}{|\mathsf{f}|} \sum_{\mathsf{e} \in \partial \mathsf{f}} |\mathsf{e}| \sigma_{\mathsf{f},\mathsf{e}} \nu_{\mathsf{e}}, \quad \forall \mathsf{f} \in \mathsf{F}, \tag{19}$$

where $v_{\rm e}$ is the value of the edge function \boldsymbol{v} that is attached to the edge e.

Remark 3.2. By construction it immediately follows that $CURL \circ GRAD = 0$, which mimics the differential identity of calculus curl $\circ \nabla = 0$.

We now assume that two quadrature formulas for the volume integrals of the bilinear forms (10) and (11) are available with the following properties: they are first-order accurate and act, respectively, on the whole sets of face and edge degrees of freedom. The construction of these quadrature rules is the crucial point in the derivation of an accurate mimetic discretization. We will discuss this issue in great details in the next subsection. For the moment, we introduce two quadrature rules as follows:

$$\int_{\Omega} \boldsymbol{u} \cdot \boldsymbol{v} dV = \left[\boldsymbol{u}^{\mathcal{E}}, \boldsymbol{v}^{\mathcal{E}} \right]_{\mathcal{E}} + \mathcal{O}(h), \tag{20}$$

$$\int_{O} \boldsymbol{\mu}^{-1} \boldsymbol{u} \cdot \boldsymbol{v} dV = \left[\boldsymbol{u}^{\mathcal{F}}, \boldsymbol{v}^{\mathcal{F}} \right]_{\mathcal{F}} + \mathcal{O}(h), \tag{21}$$

where \boldsymbol{u} and \boldsymbol{v} are sufficiently regular vector fields, and, according to (16) and (17, $\boldsymbol{v}^{\mathcal{E}}, \boldsymbol{v}^{\mathcal{E}} \in \mathcal{E}$ and $\boldsymbol{u}^{\mathcal{F}}, \boldsymbol{v}^{\mathcal{F}} \in \mathcal{F}$ are the *edge* and the *face* degrees of freedom of $\boldsymbol{u}, \boldsymbol{v}$, respectively. Using these quadrature formulas, it is straightforward to define the discrete bilinear forms

$$\forall \boldsymbol{u}, \boldsymbol{v} \in \mathcal{E}: \quad \mathcal{A}_{h}(\boldsymbol{u}, \boldsymbol{v}) := [\mathcal{CURL}(\boldsymbol{u}), \mathcal{CURL}(\boldsymbol{v})]_{\mathcal{F}}, \tag{22}$$

$$\forall \boldsymbol{v} \in \mathcal{E}, \quad \boldsymbol{q} \in \mathcal{N} : \quad \mathcal{B}_{h}(\boldsymbol{v}, \boldsymbol{q}) := [\boldsymbol{v}, \mathcal{GRAD}(\boldsymbol{q})]_{\mathcal{E}}, \tag{23}$$

which are, indeed, our mimetic approximations of the bilinear forms (10) and (11). Similarly, we consider the discretization of the integral of (12) (to be used in the right-hand side of (13)) through the edge-based quadrature formula:

$$\int_{\Omega} \boldsymbol{J} \cdot \boldsymbol{\nu} \boldsymbol{d} \boldsymbol{V} = \left[\boldsymbol{J}^{\mathcal{E}}, \, \boldsymbol{\nu}^{\mathcal{E}} \right]_{\mathcal{E}} + \mathcal{O}(\boldsymbol{h}), \tag{24}$$

which involves the discrete edge-based vector fields J^{ε} , $v^{\varepsilon} \in \mathcal{E}$.

Eventually, the formulation of the mimetic discretization of (13) and (14) is given by

Find $(\mathbf{u}_h, p_h) \in \mathcal{E}_{\mathbf{g}} \times \mathcal{N}_0$ such that :

$$\mathcal{A}_{h}(\boldsymbol{u}_{h},\boldsymbol{v}) + \mathcal{B}_{h}(\boldsymbol{v},p_{h}) = \left[\boldsymbol{J}^{\varepsilon},\boldsymbol{v}\right]_{\varepsilon} \quad \forall \boldsymbol{v} \in \mathcal{E}_{0},$$

$$(25)$$

$$\mathcal{B}_h(\boldsymbol{u}_h, q) = \mathbf{0} \quad \forall q \in \mathcal{N}_0.$$

Note that the Dirichlet conditions for the discrete solution fields \boldsymbol{u}_h and p_h are automatically taken into account because these fields belong to \mathcal{E}_g and \mathcal{N}_0 , respectively.

3.1. Linear algebraic formulation

In this subsection, we present the linear algebraic formulation that arises from the mimetic finite difference discretization (25) and (26). This construction is devised in four steps. First, we note that both discrete operators \mathcal{GRAD} and \mathcal{CURL} are linear; hence, their action on node-and edge-based discrete fields can be represented as a matrix–vector multiplication. Let \mathbb{G} be the $m_{\mathcal{E}} \times m_{\mathcal{N}}$ matrix such that

$$\forall q \in \mathcal{N}: \quad \mathcal{GRAD}(q) = \mathbb{G}q, \tag{27}$$

which implies that the component of the discrete gradient of the node-based vector q attached to the edge e is obtained by the scalar product between the row of \mathbb{G} associated to e and the vector of numbers q. Similarly, we define the $m_{\mathcal{F}} \times m_{\mathcal{E}}$ matrix \mathbb{C} that yields the discrete curl of an edge-based vector $\boldsymbol{v} \in \mathcal{E}$ through a matrix–vector multiplication:

$$\forall \boldsymbol{v} \in \mathcal{E}: \quad \mathcal{CURL}(\boldsymbol{v}) = \mathbb{C}\boldsymbol{v}. \tag{28}$$

In accordance with the definitions (18) and (19), the non-zero entries of \mathbb{G} on the row associated to an edge $e \in E$ are equal to $\pm 1/|e|$, while the non-zero entries of \mathbb{C} on the row associated to a face $f \in F$ are equal to $\sigma_{f,e}|e|/|f|$. It is easy to see that up to a

Please cite this article in press as: K. Lipnikov et al., The mimetic finite difference method for the 3D magnetostatic field problems on polyhedral meshes, J. Comput. Phys. (2010), doi:10.1016/j.jcp.2010.09.007

K. Lipnikov et al./Journal of Computational Physics xxx (2010) xxx-xxx

suitable (left diagonal) rescaling by edge lengths and face areas, \mathbb{G} and \mathbb{C} are, respectively, the *incidence matrix* describing the *edge-node connectivity* and the *incidence matrix* describing the *face-edge connectivity* of mesh \mathcal{T}_h .

The second step consists in reformulating the two quadrature rules (20) and (21) as a vector-matrix-vector multiplication. To this purpose, we define the $m_{\mathcal{E}} \times m_{\mathcal{E}}$ matrix $\mathbb{M}_{\mathcal{E}}$ such that

$$\forall \boldsymbol{u}, \boldsymbol{v} \in \mathcal{E}: \quad [\boldsymbol{u}, \boldsymbol{v}]_{\mathcal{F}} = \boldsymbol{v}^{\mathsf{T}} \mathbb{M}_{\mathcal{E}} \boldsymbol{u}. \tag{29}$$

Likewise, we define the $m_\mathcal{F} imes m_\mathcal{F}$ matrix $\mathbb{M}_\mathcal{F}$ such that

$$\forall \boldsymbol{u}, \boldsymbol{v} \in \mathcal{F}: \quad [\boldsymbol{u}, \boldsymbol{v}]_{\mathcal{F}} = \boldsymbol{v}^{T} \mathbb{M}_{\mathcal{F}} \boldsymbol{u}. \tag{30}$$

Since both quadrature formulas are an approximation of an L^2 inner product between the vector-valued functions **u** and **v**, we assume that $M_{\mathcal{E}}$ and $M_{\mathcal{F}}$ are inner product matrices for the linear spaces \mathcal{E} and \mathcal{F} , respectively. This assumption implies that $M_{\mathcal{E}}$ and $M_{\mathcal{F}}$ must be symmetric and positive definite. The construction of these matrices is performed locally by defining, for any element $P \in \mathcal{T}_h$, suitable elemental matrices that act on the restriction of the degrees of freedom to the element and then assembling all the elemental contributions in a finite element fashion. This procedure is detailed in the next sub-section.

Using the discrete gradient and curl operators (18) and (19), and the matrices introduced in (29) and (30), we reformulate, in the third step, the bilinear forms (22) and (23) and the quadrature term of the right-hand side of (24) as follows:

$$\forall \boldsymbol{u}, \boldsymbol{v} \in \mathcal{E}: \quad \mathcal{A}_h(\boldsymbol{u}, \boldsymbol{v}) = \boldsymbol{v}^T \mathbb{A} \boldsymbol{u}, \quad \text{where } \mathbb{A} = \mathbb{C}^T \mathbb{M}_{\mathcal{F}} \mathbb{C}, \tag{31}$$

$$\forall \boldsymbol{v} \in \mathcal{E}, \quad \forall \boldsymbol{q} \in \mathcal{N}: \quad \mathcal{B}_{h}(\boldsymbol{v}, \boldsymbol{q}) = \boldsymbol{v}^{\mathrm{T}} \mathbb{B}^{\mathrm{T}} \boldsymbol{q}, \quad \text{where } \mathbb{B} = \mathbb{G}^{\mathrm{T}} \mathbb{M}_{\mathcal{E}}, \tag{32}$$

$$\forall \mathbf{J} \in H(\operatorname{curl}, \Omega) \cap (\mathcal{C}^0(\bar{\Omega}))^3, \quad \forall \mathbf{v} \in \mathcal{E} : \quad [\mathbf{J}^{\mathcal{E}}, \mathbf{v}]_{\mathcal{E}} = \mathbf{v}^T \ \mathbb{M}_{\mathcal{E}} \mathbf{J}^{\mathcal{E}}. \tag{33}$$

In the fourth and final step, we reformulate the mimetic scheme (25) and (26) in the linear algebraic form:

Find
$$(\boldsymbol{u}_h, \boldsymbol{p}_h) \in \mathcal{E}_{\boldsymbol{g}} \times \mathcal{N}_0$$
 such that :

$$\boldsymbol{\nu}^{T} \mathbb{A} \boldsymbol{u}_{h} + \boldsymbol{\nu}^{T} \mathbb{B}^{T} \boldsymbol{p}_{h} = \boldsymbol{\nu}^{T} \mathbb{M}_{\mathcal{E}} \boldsymbol{J}^{\mathcal{E}} \quad \forall \boldsymbol{\nu} \in \mathcal{E}_{0},$$
(34)

$$q^{I} \mathbb{B} \boldsymbol{u}_{h} = \boldsymbol{0} \quad \forall \boldsymbol{q} \in \mathcal{N}_{0}.$$

We directly impose the Dirichlet boundary conditions on the resulting linear system by eliminating equations for boundary edges and boundary nodes and by modifying properly the remaining equations to take into account that $u_h \in \mathcal{E}_g$ and $p_h \in \mathcal{N}_0$. The elimination of the boundary equations leads to a saddle-point problem for a vector of internal degrees of freedom that we denote by $(\widetilde{u_h}^T, \widetilde{p_h}^T)^T$. The reduced problem takes the form

$$\begin{bmatrix} \widetilde{A} & \widetilde{B}^T \\ \widetilde{B} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \widetilde{\boldsymbol{u}_h} \\ \widetilde{\boldsymbol{p}_h} \end{bmatrix} = \begin{bmatrix} \widetilde{M_s} \widetilde{\boldsymbol{j}}^\varepsilon + \widetilde{\boldsymbol{g}}_h \\ \mathbf{0} \end{bmatrix},$$
(36)

where $\widetilde{\mathbb{A}}$ and $\widetilde{\mathbb{B}}$ are the sub-matrices of \mathbb{A} and \mathbb{B} that correspond to the internal degrees of freedom, and the reduced righthand side $\mathbb{M}_{\boldsymbol{\ell}} \boldsymbol{J}^{\boldsymbol{\ell}}$ is modified by the vector $\widetilde{\boldsymbol{g}}_h$ to take into account the boundary values of \boldsymbol{u}_h .

Our MFD method uses one unknown per mesh edge, as the finite element method (FEM) with the lowest-order Nedelec basis functions, plus one unknown per mesh node. Since the MFD method works on arbitrary polyhedral meshes, it may use smaller total number of unknowns than the FEM on an equivalent tetrahedral partition. The covolume methods in [42,40] use one unknown per edge and one unknown per face on tetrahedral meshes and expected to be more computationally expensive since the number of faces in a tetrahedral mesh is usually much bigger than the number of vertices. The 3D DDFV formulation proposed recently in [19] for non-linear scalar diffusion problems is not restricted to meshes of tetrahedra. Nonetheless, its formulation requires one unknown per mesh node, edge, face and cell for each scalar variable and a straightforward extension to Maxwell's equations seems impractical.

3.2. Local construction of matrices $M_{\mathcal{E}}$ and $M_{\mathcal{F}}$

In this section, we describe how the matrices $\mathbb{M}_{\mathcal{E}}$ and $\mathbb{M}_{\mathcal{F}}$ are built by assembling local matrices defined for each polyhedron P of the mesh. Since the argument is the same for both matrices $\mathbb{M}_{\mathcal{E}}$ and $\mathbb{M}_{\mathcal{F}}$, we present the detailed derivation for the former matrix, while, for the latter, we will give just the final formulas which are useful for the software implementation.

Let \boldsymbol{u} and \boldsymbol{v} denote two sufficiently regular vector fields defined on Ω . According to (16), let $\boldsymbol{u}_{|P}^{\varepsilon} := (\boldsymbol{u}_{e})_{e \in \partial P}$ denote the degrees of freedom of \boldsymbol{u} for the $m_{\varepsilon,P}$ edges of the polyhedron P (the same definition holds for $\boldsymbol{v}_{|P}^{\varepsilon}$). We write the numerical integration over a single polyhedron P as

$$\int_{\mathsf{P}} \boldsymbol{u} \cdot \boldsymbol{\nu} dV = \left[\boldsymbol{u}_{|\mathsf{P}}^{\mathcal{E}}, \boldsymbol{\nu}_{|\mathsf{P}}^{\mathcal{E}} \right]_{\mathcal{E},\mathsf{P}} + |\mathsf{P}|\mathcal{O}(h).$$
(37)

The quadrature rule in (37) can be expressed in matrix form through the $m_{\mathcal{E},\mathsf{P}} \times m_{\mathcal{E},\mathsf{P}}$ symmetric and positive definite matrix $\mathbb{M}_{\mathcal{E},\mathsf{P}}$ which acts on the local degrees-of-freedom:

K. Lipnikov et al./Journal of Computational Physics xxx (2010) xxx-xxx

$$\left[\boldsymbol{u}_{|\mathsf{P}}^{\mathcal{E}}, \boldsymbol{v}_{|\mathsf{P}}^{\mathcal{E}}\right]_{\mathcal{E},\mathsf{P}} := \left(\boldsymbol{v}_{|\mathsf{P}}^{\mathcal{E}}\right)^{T} \mathbb{M}_{\mathcal{E},\mathsf{P}} \boldsymbol{u}_{|\mathsf{P}}^{\mathcal{E}}.$$
(38)

Then, we split the left-hand side integral of Eq. (20) into the sum of the polyhedral contributions and apply the quadrature formula given by combining (37) and (38):

$$\int_{\Omega} \boldsymbol{u} \cdot \boldsymbol{v} dV = \sum_{\mathsf{P} \in \mathcal{T}_{h}} \int_{\mathsf{P}} \boldsymbol{u} \cdot \boldsymbol{v} dV = \sum_{\mathsf{P} \in \mathcal{T}_{h}} \left(\left[\boldsymbol{u}_{|\mathsf{P}}^{\mathcal{E}}, \boldsymbol{v}_{|\mathsf{P}}^{\mathcal{E}} \right]_{\mathcal{E},\mathsf{P}} + |\mathsf{P}|\mathcal{O}(h) \right) = \sum_{\mathsf{P} \in \mathcal{T}_{h}} \left(\left(\boldsymbol{v}_{|\mathsf{P}}^{\mathcal{E}} \right)^{T} \mathbb{M}_{\mathcal{E},\mathsf{P}} \boldsymbol{u}_{|\mathsf{P}}^{\mathcal{E}} \right) + \mathcal{O}(h).$$
(39)

By comparing (20) and (39), we get the following expression for the global quadrature formula:

$$\left[\boldsymbol{u}^{\mathcal{E}}, \boldsymbol{v}^{\mathcal{E}}\right]_{\mathcal{E}} := \sum_{\mathsf{P}\in\mathcal{T}_{h}} \left[\boldsymbol{u}^{\mathcal{E}}_{|\mathsf{P}}, \boldsymbol{v}^{\mathcal{E}}_{|\mathsf{P}}\right]_{\mathcal{E},\mathsf{P}},\tag{40}$$

which also takes the equivalent matrix form:

$$(\boldsymbol{v}^{\mathcal{E}})^{T}\mathbb{M}_{\mathcal{E}}\boldsymbol{u}^{\mathcal{E}} = \sum_{\mathsf{P}\in\mathcal{T}_{h}} (\boldsymbol{v}_{|\mathsf{P}}^{\mathcal{E}})^{T}\mathbb{M}_{\mathcal{E},\mathsf{P}}\boldsymbol{u}_{|\mathsf{P}}^{\mathcal{E}}.$$
(41)

Let $\mathbb{S}_{\mathcal{E},\mathsf{P}}$ be the *restriction matrix* that provides the edge degrees of freedom of a polyhedron P when it is applied to an edge function of \mathcal{E} , i.e., $\boldsymbol{v}_{ip}^{\mathcal{E}} = \mathbb{S}_{\mathcal{E},\mathsf{P}} \boldsymbol{v}^{\mathcal{E}}$. The size of $\mathbb{S}_{\mathcal{E},\mathsf{P}}$ is $m_{\mathcal{E},\mathsf{P}} \times m_{\mathcal{E}}$. Using this definition in (41), we get:

$$(\boldsymbol{\nu}^{\mathcal{E}})^{T}\mathbb{M}_{\mathcal{E}}\boldsymbol{u}^{\mathcal{E}} = (\boldsymbol{\nu}^{\mathcal{E}})^{T} \left(\sum_{\mathsf{P}\in\mathcal{T}_{h}} \mathbb{S}_{\mathcal{E},\mathsf{P}}^{T}\mathbb{M}_{\mathcal{E},\mathsf{P}} \mathbb{S}_{\mathcal{E},\mathsf{P}}\right) \boldsymbol{u}^{\mathcal{E}},\tag{42}$$

which implies that

$$\mathbb{M}_{\mathcal{E}} = \sum_{\mathsf{P}\in\mathcal{T}_{h}} \mathbb{S}_{\mathcal{E},\mathsf{P}}^{T} \mathbb{M}_{\mathcal{E},\mathsf{P}} \mathbb{S}_{\mathcal{E},\mathsf{P}}.$$
(43)

Repeating this argument for the left-hand side integral of (21) leads to a similar formula that allows us to assemble the global matrix $\mathbb{M}_{\mathcal{F}}$ from the polyhedral matrices $\mathbb{M}_{\mathcal{F},P}$:

$$\mathbb{M}_{\mathcal{F}} = \sum_{\mathsf{P}\in\mathcal{T}_h} \mathbb{S}_{\mathcal{F},\mathsf{P}}^{\mathcal{T}} \mathbb{M}_{\mathcal{F},\mathsf{P}} \mathbb{S}_{\mathcal{F},\mathsf{P}}.$$
(44)

In this formula, $\mathbb{S}_{\mathcal{F},\mathsf{P}}$ is the *restriction matrix* that gives the face degrees of freedom of polyhedron P, denoted by $\boldsymbol{v}_{\mathcal{F}}^{\mathcal{F}}$, when it is applied to a face function of \mathcal{F} , i.e., $\boldsymbol{v}_{\mathcal{F}}^{\mathcal{F}} = \mathbb{S}_{\mathcal{F},\mathsf{P}} \boldsymbol{v}^{\mathcal{F}}$. The size of $\mathbb{S}_{\mathcal{F},\mathsf{P}}$ equals to $m_{\mathcal{F},\mathsf{P}} \times m_{\mathcal{F}}$, where $m_{\mathcal{F},\mathsf{P}}$ is the number of faces of P. It is worth noting that the polyhedral matrix $\mathbb{M}_{\mathcal{F},\mathsf{P}}$ provides a numerical integration formula which is first-order accurate for the volume integral (21) defined on P:

$$\int_{\mathsf{P}} \boldsymbol{\mu}^{-1} \boldsymbol{u} \cdot \boldsymbol{\nu} dV = (\boldsymbol{\nu}_{|\mathsf{P}}^{\mathcal{F}})^{\mathsf{T}} \mathbb{M}_{\mathcal{F},\mathsf{P}} \boldsymbol{u}_{|\mathsf{P}}^{\mathcal{F}} + |\mathsf{P}| \mathcal{O}(h).$$
(45)

In view of (45), the local matrix $\mathbb{M}_{\mathcal{F},\mathsf{P}}$ must contain information about the magnetic permeability tensor μ on P.

The general procedure for the construction of the local matrices $\mathbb{M}_{\mathcal{E},\mathsf{P}}$ and $\mathbb{M}_{\mathcal{F},\mathsf{P}}$ relies upon the *algebraic consistency condition*, which we formulate now.

Definition 3.1. Let \mathbb{M} be an $m \times m$ symmetric and positive definite matrix, and \mathbb{N} and \mathbb{R} two full rank $m \times d$ matrices for d = 2 or 3. We say that the matrices \mathbb{M} , \mathbb{N} and \mathbb{R} satisfy the algebraic consistency condition when:

(i)
$$\mathbb{MN} = \mathbb{R};$$
 (46)

(ii) $\mathbb{N}^T \mathbb{R}$ is a symmetric and positive definite matrix. (47)

At first sight, the name *algebraic consistency condition* might look mysterious, as it is difficult to see what it has to do with *consistency*. However we shall see in the following subsections that, in practice, we will choose, in each particular case, matrices \mathbb{R} and \mathbb{N} such that (46) will imply that the scalar product induced by \mathbb{M} coincides with the "exact scalar product" on scalar functions (or vectors), so that some sort of "patch test" is satisfied. In particular, as we will discuss in the next subsections, the algebraic consistency condition will stem from an $\mathcal{O}(h)$ accurate approximation of a *Gauss–Green* relation. Therefore, the matrix \mathbb{R} is not uniquely determined because any approximation of the Gauss–Green formula provides an acceptable \mathbb{R} . One possible realization is found in [9]. When matrices \mathbb{N} and \mathbb{R} are available, matrix \mathbb{M} is derived from Proposition 3.1 below. Therefore, the "game" that we systematically play to construct the mimetic finite difference method is the following: using approximation arguments, we choose an appropriate matrix \mathbb{N} and determine the correct matrix \mathbb{R} from a discrete Gauss–Green formula. The simplest choice for \mathbb{N} consists in the interpolation of the canonical basis vectors of \mathfrak{R}^3 or \mathfrak{R}^2 . With this choice the product $\mathbb{N}^T\mathbb{R}$ has an explicit form. The matrix \mathbb{M} that is provided by Proposition 3.1 for the pair of matrices (\mathbb{N}, \mathbb{R}) is then used for numerical integration.

Proposition 3.1. Let \mathbb{N} and \mathbb{R} be two full rank $m \times d$ matrices for d = 2 or 3 and $m \ge d + 1$ be such that the $d \times d$ matrix $\mathbb{N}^T \mathbb{R}$ is symmetric and positive definite. Then, a possible form of \mathbb{M} satisfying (46) is given by:

Please cite this article in press as: K. Lipnikov et al., The mimetic finite difference method for the 3D magnetostatic field problems on polyhedral meshes, J. Comput. Phys. (2010), doi:10.1016/j.jcp.2010.09.007

K. Lipnikov et al./Journal of Computational Physics xxx (2010) xxx-xxx

$$\mathbb{M} = \mathbb{R}(\mathbb{N}^T \mathbb{R})^{-1} \mathbb{R}^T + \gamma Trace \left(\mathbb{R}(\mathbb{N}^T \mathbb{R})^{-1} \mathbb{R}^T \right) \mathbb{P},$$
(48)

9

with

$$\mathbb{P} = \mathbb{I} - \mathbb{N}(\mathbb{N}^T \mathbb{N})^{-1} \mathbb{N}^T \tag{49}$$

and γ being a strictly positive real number (in our experiments, γ is just 1/m). The matrix \mathbb{M} given by (48) and (49) is symmetric and positive definite.

Proof. A straightforward calculation shows that the matrix \mathbb{M} given by (48) satisfies condition (46), and its symmetry is an obvious consequence of the form given by (48). To prove that \mathbb{M} is positive definite, we observe that for any discrete vector \boldsymbol{v} of appropriate size *m* there holds:

$$\boldsymbol{v}^{\mathsf{T}} \mathbb{M} \, \boldsymbol{v} = \boldsymbol{v}^{\mathsf{T}} \mathbb{R} (\mathbb{N}^{\mathsf{T}} \mathbb{R})^{-1} \mathbb{R}^{\mathsf{T}} \, \boldsymbol{v} + \gamma \operatorname{Trace} \left(\mathbb{R} (\mathbb{N}^{\mathsf{T}} \mathbb{R})^{-1} \mathbb{R}^{\mathsf{T}} \right) \, \boldsymbol{v}^{\mathsf{T}} \mathbb{P} \, \boldsymbol{v} \ge \mathbf{0}.$$
(50)

The first term of the right-hand side of (50) is non negative since $(\mathbb{N}^T \mathbb{R})^{-1}$ is positive definite, but may be zero when \boldsymbol{v} belongs to the (non-trivial) kernel of \mathbb{R}^T . The second term of the right-hand side of (50) is non-negative since it is the product of three non-negative numbers: γ , which is strictly positive by hypothesis, the trace of $\mathbb{R}(\mathbb{N}^T \mathbb{R})^{-1} \mathbb{R}^T$, which is the sum of non-negative eigenvalues, and $\boldsymbol{v}^T \mathbb{P} \boldsymbol{v}$, which is non-negative because \mathbb{P} is an orthogonal projector. It is left to show that $\boldsymbol{v}^T \mathbb{M} \boldsymbol{v} = 0$ implies that $\boldsymbol{v} = 0$. Condition $\boldsymbol{v}^T \mathbb{M} \boldsymbol{v} = 0$ implies that the equations $\mathbb{R}^T \boldsymbol{v} = 0$ and $\mathbb{P} \boldsymbol{v} = 0$ hold separately. On its turn, the latter condition implies that the vector \boldsymbol{v} belongs to the linear sub-space of \mathfrak{R}^m spanned by the columns of \mathbb{N} , i.e., there exists a vector $\boldsymbol{\phi} \in \mathfrak{R}^d$ such that $\boldsymbol{v} = \mathbb{N} \boldsymbol{\phi}$. We obtain:

$$\mathbf{0} = \mathbb{R}^T \boldsymbol{\nu} = \mathbb{R}^T \mathbb{N} \boldsymbol{\phi} = \mathbb{N}^T \mathbb{R} \boldsymbol{\phi} \tag{51}$$

from which it follows that $\phi = 0$, because $\mathbb{N}^T \mathbb{R}$ is symmetric and positive definite by hypothesis and, hence, non-singular. Therefore, we have that $\boldsymbol{v} = 0$. This proves the assertion of the proposition. \Box

3.2.1. 3D consistency relation for edge functions

In the mimetic setting discussed at the end of the previous sub-section, we choose the matrix \mathbb{N} as follows. Let us consider the three constant vectors that form the canonical basis of \mathfrak{R}^3 , i.e., $\mathbf{c}_1 = (1,0,0)^T$, $\mathbf{c}_2 = (0,1,0)^T$, $\mathbf{c}_3 = (0,0,1)^T$. For each polyhedron P, we set the column vector \mathbf{N}_j for j = 1, 2, 3, which is the *j*th column of the matrix \mathbb{N} , equal to the degrees of freedom of \mathbf{c}_j for the edges of P through the relation:

$$\boldsymbol{N}_{j} = (\boldsymbol{c}_{j})_{|\mathrm{P}}^{\varepsilon}.$$
(52)

Therefore, the entries of \mathbb{N} are given by

$$\forall \mathbf{e} \in \mathsf{P} \quad \text{and } j \in \{1, 2, 3\}: \quad \mathbb{N}_{\mathbf{e}, j} = \frac{1}{|\mathbf{e}|} \int_{\mathbf{e}} \boldsymbol{c}_{j} \cdot \boldsymbol{t}_{\mathbf{e}} dS.$$
(53)

After substituting $\boldsymbol{u} = \boldsymbol{c}_j$, equations (37) and (38) imply that

$$\int_{\mathsf{P}} \boldsymbol{c}_{j} \cdot \boldsymbol{\nu} dV = \left(\boldsymbol{\nu}_{|\mathsf{P}}^{\mathcal{E}}\right)^{T} \mathbb{M}_{\mathcal{E},\mathsf{P}} \boldsymbol{N}_{j} + |\mathsf{P}| \mathcal{O}(h).$$
(54)

The rest of this sub-section is devoted to derivation a proper expression for the columns R_j of the matrix \mathbb{R} so that the quadrature rule (54) can also take the form:

$$\int_{\mathsf{P}} \boldsymbol{c}_{j} \cdot \boldsymbol{\nu} dV = \left(\boldsymbol{\nu}_{|\mathsf{P}}^{\mathcal{E}}\right)^{\mathsf{T}} \boldsymbol{R}_{j} + |\mathsf{P}|\mathcal{O}(h).$$
(55)

Comparing (54) and (55) yields the algebraic consistency condition of the form " $\mathbb{M}_{\mathcal{E},P}\mathbb{N} = \mathbb{R}$ " that allows us to derive the matrix $\mathbb{M}_{\mathcal{E},P}$ from Proposition 3.1.

To this aim, let c be a constant vector field on the polyhedron P and note that we can always set $c = \operatorname{curl}(p^1)$ with $p^1(x) = (1/2)c \times (x - x_P)$ for every $x \in P$. Here x_P is the center of gravity of the polyhedron P. All the components of the vector field p^1 are linear scalar functions on P, p^1 is a divergence-free vector function and is orthogonal to the constant vectors with respect to the L^2 inner product defined on the polyhedron P. Let us define

$$\langle \operatorname{curl}(\boldsymbol{v}) \rangle_{\mathsf{P}} = \frac{1}{|\mathsf{P}|} \int_{\mathsf{P}} \operatorname{curl}(\boldsymbol{v}) dV.$$
 (56)

From the orthogonality of p^1 to the constant vectors, the Cauchy–Schwarz inequality and the fact that for any sufficiently regular function ϕ the cell average $\langle \phi \rangle_P$ is a first-order approximation of ϕ on P, we obtain that

$$\int_{\mathsf{P}} \boldsymbol{p}^{1} \cdot \operatorname{curl}(\boldsymbol{v}) dV = \int_{\mathsf{P}} \boldsymbol{p}^{1} \cdot (\operatorname{curl}(\boldsymbol{v}) - \langle \operatorname{curl}(\boldsymbol{v}) \rangle_{\mathsf{P}}) dV \leqslant \|\boldsymbol{p}^{1}\|_{(L^{2}(\mathsf{P}))^{3}} \|\operatorname{curl}(\boldsymbol{v}) - \langle \operatorname{curl}(\boldsymbol{v}) \rangle\|_{(L^{2}(\mathsf{P}))^{3}} = |\mathsf{P}|\mathcal{O}(h).$$
(57)

K. Lipnikov et al./Journal of Computational Physics xxx (2010) xxx-xxx

Using $\boldsymbol{c} = \operatorname{curl}(\boldsymbol{p}^1)$, we get the following development:

$$\int_{P} \mathbf{c} \cdot \mathbf{v} dV = \int_{P} \operatorname{curl}(\mathbf{p}^{1}) \cdot \mathbf{v} dV \quad (\text{integrate by parts})$$

$$= \int_{P} \mathbf{p}^{1} \cdot \operatorname{curl}(\mathbf{v}) dV + \int_{\partial P} \mathbf{n}_{P} \cdot (\mathbf{p}^{1} \times \mathbf{v}) dS \quad (\text{use estimate (57)})$$

$$= |P|\mathcal{O}(h) + \int_{\partial P} \mathbf{n}_{P} \cdot (\mathbf{p}^{1} \times \mathbf{v}) dS \quad (\text{split the boundary integral})$$

$$= |P|\mathcal{O}(h) + \sum_{f \in \partial P} \int_{f} \mathbf{n}_{P,f} \cdot (\mathbf{p}^{1} \times \mathbf{v}) dS \quad (\text{use } \mathbf{a} \cdot (\mathbf{b} \times \mathbf{c}) = \mathbf{b} \cdot (\mathbf{c} \times \mathbf{a}))$$

$$= |P|\mathcal{O}(h) + \sum_{f \in \partial P} \int_{f} \mathbf{v} \cdot (\mathbf{n}_{P,f} \times \mathbf{p}^{1}) dS. \quad (58)$$

In the previous development we introduced \mathbf{n}_{P} , which is the unit vector orthogonal to the polyhedron's boundary ∂P , and $\mathbf{n}_{P,f}$, which is the unit vector orthogonal to the polyhedron's face $f \in \partial P$. Both vectors \mathbf{n}_{P} and $\mathbf{n}_{P,f}$ point out of P. It is worth noting that the previous development is exact if \boldsymbol{v} or curl(\boldsymbol{v}) are constant vectors on P because in both cases the volume integral on P, which is neglected to obtain the first-order accurate approximation (58), is zero. This remark is crucial in proving that $\mathbb{N}^T \mathbb{R}$ is a positive definite matrix, as we will show at the end of this sub-section.

The next step consists in reformulating the face integrals of the right-hand side of (58) in a 2D way. To do so, we first note that the vector $\mathbf{n}_{P,f} \times \mathbf{p}^1$ is parallel to face f. Thus, it is convenient to split \mathbf{v} into the sum of its parallel and perpendicular component with respect to f, i.e., $\mathbf{v} = \mathbf{v}^{\parallel} + \mathbf{v}^{\perp}$. Using such a decomposition, we readily get:

$$\int_{f} \boldsymbol{v} \cdot (\boldsymbol{n}_{\mathsf{P},\mathsf{f}} \times \boldsymbol{p}^{1}) dS = \int_{f} (\boldsymbol{v}^{\parallel} + \boldsymbol{v}^{\perp}) \cdot (\boldsymbol{n}_{\mathsf{P},\mathsf{f}} \times \boldsymbol{p}^{1}) dS = \int_{f} \boldsymbol{v}^{\parallel} \cdot (\boldsymbol{n}_{\mathsf{P},\mathsf{f}} \times \boldsymbol{p}^{1}) dS.$$
(59)

The right-most integral of Eq. (59) can be reformulated using the following notation:

$$\hat{\mathbf{v}}_{f} = \boldsymbol{v}^{\parallel} \quad \text{and} \; \hat{\mathbf{p}}_{f}^{1} = \boldsymbol{n}_{\mathsf{P},f} \times \boldsymbol{p}^{1}.$$
 (60)

We have:

$$\int_{f} \boldsymbol{\nu}^{\parallel} \cdot \left(\boldsymbol{n}_{\mathsf{P},\mathsf{f}} \times \boldsymbol{p}^{1}(\boldsymbol{x})\right) dS = \int_{f} \hat{\mathbf{v}}_{\mathsf{f}} \cdot \hat{\boldsymbol{p}}_{\mathsf{f}}^{1} dS.$$
(61)

In a local coordinate system associated with the face f, the vector functions $\hat{\mathbf{v}}_f$ and $\hat{\mathbf{p}}_f^I$ have only two non-zero components. Thus, the latter integral is reduced to a 2D integral. We complete this derivation by assuming (for the moment) that a quadrature formula is available for the numerical integration of the right-hand side of (61). This quadrature formula, whose construction is detailed in the next sub-section, is required: (i) to be first-order accurate; (ii) to depend on the edge degrees of freedom of the integrand terms related to the face f; (iii) to be exact when $\hat{\mathbf{v}}_f$ is constant on f. This numerical integration rule is formally written as:

$$\int_{\mathsf{f}} \hat{\mathbf{v}}_{\mathsf{f}}(\boldsymbol{\xi}) \cdot \hat{\mathbf{p}}_{\mathsf{f}}^{\mathbf{1}}(\boldsymbol{\xi}) dS = \left[(\hat{\mathbf{v}}_{\mathsf{f}})_{|\mathsf{f}}^{\mathcal{E}}, (\hat{\mathbf{p}}_{\mathsf{f}}^{\mathbf{1}})_{|\mathsf{f}}^{\mathcal{E}} \right]_{\mathcal{E},\mathsf{f}} + |\mathsf{f}|\mathcal{O}(h).$$
(62)

Assumptions (i)–(iii) imply that the right-hand side of (62) can take the form of a vector–matrix–vector multiplication through the introduction of a suitable face matrix $\mathbb{M}_{\mathcal{E},f}$:

$$\left[\left(\hat{\mathbf{v}}_{\mathsf{f}} \right)_{|\mathsf{f}}^{\mathcal{E}}, \left(\hat{\mathbf{p}}_{\mathsf{f}}^{\mathsf{1}} \right)_{|\mathsf{f}}^{\mathcal{E}} \right]_{\mathcal{E},\mathsf{f}} = \left(\left(\hat{\mathbf{v}}_{\mathsf{f}} \right)_{|\mathsf{f}}^{\mathcal{E}} \right)^{T} \mathbb{M}_{\mathcal{E},\mathsf{f}} \left(\hat{\mathbf{p}}_{\mathsf{f}}^{\mathsf{1}} \right)_{|\mathsf{f}}^{\mathcal{E}}.$$
(63)

The matrix $M_{\mathcal{E},f}$ that is defined for the face f will be derived in the next subsection from a 2D consistency relation for the faces by applying again Proposition 3.1.

Now, substituting back all expressions from (63) to (58) leads to the integration rule:

$$\int_{\mathsf{P}} \boldsymbol{c} \cdot \boldsymbol{\nu} dV = \sum_{\mathsf{f} \in \partial \mathsf{P}} \left((\hat{\mathbf{v}}_{\mathsf{f}})_{|\mathsf{f}}^{\mathcal{E}} \right)^T \mathbb{M}_{\mathcal{E},\mathsf{f}} (\hat{\mathbf{p}}_{\mathsf{f}}^1)_{|\mathsf{f}}^{\mathcal{E}} + |\mathsf{P}|\mathcal{O}(h).$$
(64)

We can go one step further in this development by introducing the *restriction* matrix $S_{\varepsilon,P,f}$ that extracts the edge degrees of freedom of face f when applied to an edge function of $\mathcal{E}_{|P}$. The size of $S_{\varepsilon,P,f}$ equals $m_{\varepsilon,f} \times m_{\varepsilon,P}$, where $m_{\varepsilon,f}$ is the number of edges of face f and $m_{\varepsilon,P}$ the number of edges of polyhedron P. Therefore, there holds that $\boldsymbol{v}_{f}^{\varepsilon} = S_{\varepsilon,P,f} \boldsymbol{v}_{|P}^{\varepsilon}$, and, since the degrees of freedom of $(\hat{\mathbf{v}}_{t})_{f}^{\varepsilon}$ coincide with the degrees of freedom of \boldsymbol{v} for the same edges, we obtain

$$\int_{\mathsf{P}} \boldsymbol{c} \cdot \boldsymbol{\boldsymbol{\nu}} dV = \sum_{\mathsf{f} \in \partial \mathsf{P}} \left(\left(\hat{\mathbf{v}}_{\mathsf{f}} \right)_{|\mathsf{f}}^{\mathcal{E}} \right)^T \mathbb{M}_{\mathcal{E},\mathsf{f}} \left(\hat{\mathbf{p}}_{\mathsf{f}}^{\mathsf{1}} \right)_{|\mathsf{f}}^{\mathcal{E}} + |\mathsf{P}|\mathcal{O}(h) = \left(\boldsymbol{\boldsymbol{\nu}}_{|\mathsf{P}}^{\mathcal{E}} \right)^T \sum_{\mathsf{f} \in \partial \mathsf{P}} \mathbb{S}_{\mathcal{E},\mathsf{P},\mathsf{f}}^T \mathbb{M}_{\mathcal{E},\mathsf{f}} \left(\hat{\mathbf{p}}_{\mathsf{f}}^{\mathsf{1}} \right)_{|\mathsf{f}}^{\mathcal{E}} + |\mathsf{P}|\mathcal{O}(h).$$
(65)

The formula for \mathbb{R} is derived from (65). To this purpose, we just use the vectors \mathbf{c}_j for j = 1, 2, 3, introduced at the beginning of this section, instead of \mathbf{c} , and let $\hat{\mathbf{p}}_{i,j}^1$ be defined by (60) using $\mathbf{p}_i^1(\mathbf{x}) = (1/2)\mathbf{c}_j \times (\mathbf{x} - \mathbf{x}_P)$ instead of $\mathbf{p}^1(\mathbf{x})$. We have that

K. Lipnikov et al./Journal of Computational Physics xxx (2010) xxx-xxx

$$\int_{\mathsf{P}} \boldsymbol{c}_{j} \cdot \boldsymbol{\boldsymbol{\nu}} dV = \left(\boldsymbol{\boldsymbol{\nu}}_{|\mathsf{P}}^{\mathcal{E}}\right)^{T} \sum_{\mathsf{f} \in \partial \mathsf{P}} \mathbb{S}_{\mathcal{E},\mathsf{P},\mathsf{f}}^{T} \mathbb{M}_{\mathcal{E},\mathsf{f}} \left(\hat{\mathbf{p}}_{\mathsf{f},j}^{1}\right)_{|\mathsf{f}}^{\mathcal{E}} + |\mathsf{P}|\mathcal{O}(h) = \left(\boldsymbol{\boldsymbol{\nu}}_{|\mathsf{P}}^{\mathcal{E}}\right)^{T} \boldsymbol{\boldsymbol{R}}_{j} + |\mathsf{P}|\mathcal{O}(h), \tag{66}$$

11

where

$$\boldsymbol{R}_{j} := \sum_{f \in \partial \mathsf{P}} \mathbb{S}_{\mathcal{E},\mathsf{P},\mathsf{f}}^{T} \mathbb{M}_{\mathcal{E},\mathsf{f}} \left(\hat{\boldsymbol{p}}_{\mathsf{f},j}^{1} \right)_{|\mathsf{f}}^{\mathcal{E}}.$$
(67)

Comparing (66) with (54), neglecting O(h) terms, and using the arbitrariness of \boldsymbol{v} , we get the algebraic consistency relation in the form required by Proposition 3.1:

for
$$j \in \{1, 2, 3\}$$
: $\mathbb{M}_{\mathcal{E}, \mathcal{P}} \mathbf{N}_j = \mathbf{R}_j.$ (68)

We are left to prove that $\mathbb{N}^T \mathbb{R}$ is a symmetric and positive definite matrix. This is a consequence of the fact that the numerical integration formulas developed so far are exact when \boldsymbol{v} is a constant vector on P. Let us substitute $\boldsymbol{v} = \boldsymbol{c}_i$ for i = 1, 2, 3 in (66). We obtain:

$$\delta_{ij}|\mathsf{P}| = \int_{\mathsf{P}} \boldsymbol{c}_{j} \cdot \boldsymbol{c}_{i} dV = \left((\boldsymbol{c}_{i})_{|\mathsf{P}}^{\mathcal{E}} \right)^{\mathsf{T}} \boldsymbol{R}_{j} = \boldsymbol{N}_{i}^{\mathsf{T}} \boldsymbol{R}_{j}, \tag{69}$$

where $\delta_{ij} = 1$ for i = j and $\delta_{ij} = 0$ for $i \neq j$. Eq. (69) implies that $\mathbb{N}^T \mathbb{R} = |\mathsf{P}|\mathbb{I}$, from which it is obvious that $\mathbb{N}^T \mathbb{R}$ is symmetric and positive definite.

It is worth noting that the previous derivation is indeed related to a very precise *discrete Gauss–Green relation* or discrete integration-by-parts formula. In fact, substituting (62) into (61), the resulting expression into (59) and then into (58), and taking $\mathbf{c} = \operatorname{curl}(\mathbf{p}^1)$, we get:

$$\int_{\mathsf{P}} \boldsymbol{\nu} \cdot \operatorname{curl}(\boldsymbol{p}^{1}) dV = \sum_{\mathsf{f} \in \partial \mathsf{P}} \left[(\boldsymbol{\nu})_{|\mathsf{f}}^{\mathcal{E}}, \left(\boldsymbol{n}_{\mathsf{P},\mathsf{f}} \times \boldsymbol{p}^{1} \right)_{|\mathsf{f}}^{\mathcal{E}} \right]_{\mathcal{E},\mathsf{f}} + |\mathsf{P}|\mathcal{O}(h).$$
(70)

Applying the quadrature to the integral in the left-hand side and neglecting the O(h) terms, we get the following definition.

Definition 3.2. Let $(\mathbb{P}_1(\mathsf{P})/\Re)^3$ denote the space of linear vectors orthogonal to constant vector fields on the polyhedron P. The discrete Gauss–Green relation for edge functions defined on the polyhedron P takes the form:

$$\forall \boldsymbol{v} \in \mathcal{E}_{|\mathsf{P}}, \quad \forall \boldsymbol{p}^{1} \in \left(\mathbb{P}_{1}(\mathsf{P})/\Re\right)^{3}: \quad \left[\boldsymbol{v}, \left(\operatorname{curl}(\boldsymbol{p}^{1})\right)_{|\mathsf{P}}^{\mathcal{E}}\right]_{\mathcal{E},\mathsf{P}} := \sum_{\mathsf{f} \in \partial^{\mathsf{P}}} \left[\boldsymbol{v}_{|\mathsf{f}}, \left(\boldsymbol{n}_{\mathsf{P},\mathsf{f}} \times \boldsymbol{p}^{1}\right)_{|\mathsf{f}}^{\mathcal{E}}\right]_{\mathcal{E},\mathsf{f}}.$$
(71)

Now, recalling that in the present setting $N_j = (c_j)_{|p|}^{\mathcal{E}}$ and using the discrete Gauss–Green relation (71), we get the alternative derivation of the algebraic consistency condition:

$$\forall \boldsymbol{v} \in \mathcal{E}_{|\mathsf{P}}: \quad \boldsymbol{v}^{\mathsf{T}} \mathbb{M}_{\mathcal{E},\mathsf{P}} \boldsymbol{N}_{j} = \left[\boldsymbol{v}, (\boldsymbol{c}_{j})_{|\mathsf{P}}^{\mathcal{E}}\right]_{\mathcal{E},\mathsf{P}} = \left[\boldsymbol{v}, \operatorname{curl}(\boldsymbol{p}_{j}^{1})_{|\mathsf{P}}^{\mathcal{E}}\right]_{\mathcal{E},\mathsf{P}} \equiv \boldsymbol{v}^{\mathsf{T}} \boldsymbol{R}_{j}, \tag{72}$$

where \mathbf{R}_{i} is defined through the last identity.

We finally emphasize that the whole derivation is based on the existence of the integration formula (62) with the properties listed therein. We will investigate this issue in the next sub-section.

3.2.2. 2D consistency relation for edge functions

As in the 3D case, we first choose the matrix \mathbb{N} and then determine the correct matrix \mathbb{R} such that a consistency condition of the form " $\mathbb{M}_{\mathcal{E},f}\mathbb{N} = \mathbb{R}$ " holds. The matrix $\mathbb{M}_{\mathcal{E},f}$ that we need for the numerical integration formula (63), follows from Proposition 3.1.

Let us consider a face f and introduce a 2D system of coordinates $\boldsymbol{\xi} = (\xi_1, \xi_2)^T$ associated with it. We define in the plane of f the 2D unit vectors $\boldsymbol{n}_{f,e}$ and $\boldsymbol{t}_{f,e}$ that are orthogonal and tangential, respectively, to the edge e of f. We assume that the orientation is such that det($\boldsymbol{n}_{f,e} \times \boldsymbol{t}_{f,e}$) > 0. For simplicity, we use the same notation \boldsymbol{t}_e for the 2D tangential vector that defines the unique global orientation of the edges.

Let us consider the two constant vectors that form the canonical basis of the vector space \Re^2 , i.e., $\mathbf{c}_1 = (1,0)^T$, $\mathbf{c}_2 = (0,1)^T$. The *j*th column N_j of the matrix \mathbb{N} for j = 1, 2 is given formally by

$$\boldsymbol{N}_{j} = \left(\boldsymbol{c}_{j}\right)_{| \boldsymbol{f}}^{\mathcal{E}}.$$
(73)

The entries of the matrix \mathbb{N} are calculated as follows:

$$\forall \mathsf{e} \in \partial \mathsf{f} \quad \text{and } j \in \{1, 2\}: \quad \mathbb{N}_{\mathsf{e},j} = \frac{1}{|\mathsf{e}|} \int_{\mathsf{e}}^{\mathsf{c}} \boldsymbol{c}_j \cdot \boldsymbol{t}_{\mathsf{e}} \, dL. \tag{74}$$

Now, we derive an expression for the columns R_j so that the quadrature rule

ARTICLE IN PRESS K. Lipnikov et al. / Journal of Computational Physics xxx (2010) xxx-xxx

$$\int_{f} \boldsymbol{c}_{j} \cdot \boldsymbol{\nu} dS = \left(\boldsymbol{\nu}_{|f}^{\mathcal{E}}\right)^{T} \mathbb{M}_{\mathcal{E}, f} \boldsymbol{N}_{j} + |f| \mathcal{O}(h)$$
(75)

can also take the form

$$\int_{f} \boldsymbol{c}_{j} \cdot \boldsymbol{\boldsymbol{\nu}} dS = \left(\boldsymbol{\boldsymbol{\nu}}_{|f}^{\mathcal{E}}\right)^{T} \boldsymbol{R}_{j} + |f| \mathcal{O}(h)$$
(76)

for every generic and sufficiently regular vector-valued function \pmb{v} defined on f.

In order to do so, we introduce the 2D operators

$$\forall \psi \in H^{1}(f): \quad \operatorname{Curl}_{\xi}(\psi) = \left(-\frac{\partial \psi}{\partial \xi_{2}}, \frac{\partial \psi}{\partial \xi_{1}}\right)^{T}, \tag{77}$$

$$\forall \boldsymbol{\phi} = (\phi_1, \phi_2) \in H(\operatorname{curl}, f): \quad \operatorname{curl}_{\boldsymbol{\xi}}(\boldsymbol{\phi}) = \frac{\partial \phi_1}{\partial \xi_2} - \frac{\partial \phi_2}{\partial \xi_1}.$$
(78)

Let $\mathbf{c} = (c_1, c_2)^T$ be a constant vector field defined on face f. We take the scalar function $p^1(\xi) = -c_1(\xi_2 - \xi_{f,2}) + c_2(\xi_1 - \xi_{f,1})$ where $\xi_f = (\xi_{f,1}, \xi_{f,2})^T$ is the local position vector of the center of gravity of face f. It is easy to see that $p^1(\xi)$ is orthogonal to the constant functions with respect to the $L^2(f)$ inner product. Note that $\mathbf{c} = \operatorname{Curl}_{\xi}(p^1)$. Using this fact and integratingby-parts, we obtain:

$$\int_{f} \boldsymbol{c} \cdot \boldsymbol{\nu} dS = \int_{f} \operatorname{Curl}_{\xi}(p^{1}) \cdot \boldsymbol{\nu} dS = \int_{f} p^{1} \operatorname{Curl}_{\xi}(\boldsymbol{\nu}) dS + \int_{\partial f} p^{1} \boldsymbol{t}_{f} \cdot \boldsymbol{\nu} dL,$$
(79)

where t_f is the unit vector tangent to ∂f , the two-dimensional boundary of face f. Let us now define

$$\langle \operatorname{curl}_{\xi}(\boldsymbol{v}) \rangle_{\mathrm{f}} = \frac{1}{|\mathrm{f}|} \int_{\mathrm{f}} \operatorname{curl}_{\xi}(\boldsymbol{v}) dS.$$

From the orthogonality of p^1 to the constant functions, the Cauchy–Schwarz inequality and the fact that for any sufficiently regular function ψ the cell average $\langle \psi \rangle_f$ is a first-order approximation of ψ on f, we obtain that

$$\int_{f} p^{1} \operatorname{curl}_{\xi}(\boldsymbol{v}) dS = \int_{f} p^{1} (\operatorname{curl}_{\xi}(\boldsymbol{v}) - \langle \operatorname{curl}_{\xi}(\boldsymbol{v}) \rangle_{f}) dS \leqslant \|p^{1}\|_{(L^{2}(f))^{2}} \|\operatorname{curl}_{\xi}(\boldsymbol{v}) - \langle \operatorname{curl}_{\xi}(\boldsymbol{v}) \rangle_{f}\|_{(L^{2}(f))^{2}} = |f|\mathcal{O}(h).$$

$$(80)$$

Using (80) into (79) yields

$$\int_{f} \boldsymbol{c} \cdot \boldsymbol{\boldsymbol{\nu}} dS = |\mathbf{f}| \mathcal{O}(h) + \sum_{\mathbf{e} \in \partial f} \int_{\mathbf{e}} p^{1} \boldsymbol{t}_{\mathbf{f},\mathbf{e}} \cdot \boldsymbol{\boldsymbol{\nu}} dL.$$
(81)

Now, we consider the first-order approximation

$$\int_{\mathsf{e}} p^{1} \boldsymbol{t}_{\mathsf{f},\mathsf{e}} \cdot \boldsymbol{\nu} dL = |\mathsf{e}| p^{1}(\boldsymbol{\xi}_{\mathsf{e}}) \left(\frac{1}{|\mathsf{e}|} \int_{\mathsf{e}} \boldsymbol{t}_{\mathsf{f},\mathsf{e}} \cdot \boldsymbol{\nu} dS \right) + |\mathsf{e}| \mathcal{O}(h) = |\mathsf{e}| p^{1}(\boldsymbol{\xi}_{\mathsf{e}}) \sigma_{\mathsf{f},\mathsf{e}} \boldsymbol{\nu}_{|\mathsf{e}}^{\mathcal{E}} + |\mathsf{e}| \mathcal{O}(h), \tag{82}$$

where ξ_e is the mid-point of e and the sign $\sigma_{f,e} = t_{f,e} \cdot t_e$ takes into account the orientation of the edge e with respect to the face f. We end the derivation by taking $c_j = (c_{j,1}, c_{j,2})^T$ for j = 1, 2 introduced at the beginning of this sub-section instead of the generic constant vector c and the scalar polynomial $p_j^1(\xi) = -c_{j,1}(\xi_2 - \xi_{f,2}) + c_{j,2}(\xi_1 - \xi_{f,1})$. Quadrature formula (75) is obtained by defining the component of the column vector R_j for j = 1, 2 associated to the edge e, or, equivalently, the entry (e, j) of the matrix \mathbb{R} , by:

$$\forall \mathsf{e} \in \mathsf{f} \quad \text{and } j \in \{1, 2\}: \quad \left(\mathbf{R}_{j}\right)_{|_{\mathsf{e}}} = \mathbb{R}_{\mathsf{e},j} := |\mathsf{e}|\sigma_{\mathsf{f},\mathsf{e}} p^{1}(\boldsymbol{\xi}_{\mathsf{e}}). \tag{83}$$

In order to use Proposition 3.1, we are left to show that $\mathbb{N}^T \mathbb{R}$ is symmetric and positive definite. Note that the numerical integration rule in (81) is exact when \boldsymbol{v} is a constant vector on f (or its curl is zero). Setting $\boldsymbol{v} = \boldsymbol{c}_i$ for i = 1, 2 and noting that $\boldsymbol{c}_i \cdot \boldsymbol{c}_j = \delta_{ij}$ yields

$$\delta_{ij}|\mathbf{f}| = \int_{\mathbf{f}} \mathbf{c}_i \cdot \mathbf{c}_j dS = \left((\mathbf{c}_i)_{|\mathbf{f}}^{\mathcal{E}} \right)^T \mathbf{R}_j = \mathbf{N}_i^T \mathbf{R}_j, \tag{84}$$

which immediately implies the relation $\mathbb{N}^T \mathbb{R} = |f|\mathbb{I}$, and, consequently, that $\mathbb{N}^T \mathbb{R}$ is symmetric and positive definite.

The previous derivation is related to a *discrete Gauss–Green formula* or a discrete integration-by-parts formula. Substituting (82) into (81), and taking $\mathbf{c} = \text{Curl}_{\xi}(p^1)$ yields:

$$\int_{f} \boldsymbol{\nu} \cdot \operatorname{Curl}_{\boldsymbol{\xi}}(\boldsymbol{p}^{1}) dS = |f| \mathcal{O}(h) + \sum_{e \in \partial f} |e| \boldsymbol{p}^{1}(\boldsymbol{\xi}_{e}) \sigma_{f,e} \boldsymbol{\nu}_{|e}^{\mathcal{E}}.$$
(85)

Applying the quadrature rule to the integral in the left-hand side of (85) and neglecting the O(h) terms, we get the following definition.

Please cite this article in press as: K. Lipnikov et al., The mimetic finite difference method for the 3D magnetostatic field problems on polyhedral meshes, J. Comput. Phys. (2010), doi:10.1016/j.jcp.2010.09.007

K. Lipnikov et al./Journal of Computational Physics xxx (2010) xxx-xxx

Definition 3.3. Let $\mathbb{P}_1(f)/\mathfrak{R}$ denote the space of linear scalar functions orthogonal to constant scalar fields on the polygonal face f. The discrete Gauss–Green relation for the edge functions defined on the face f takes the form:

$$\forall \boldsymbol{\nu} \in \mathcal{E}_{|\mathsf{f}}, \quad \forall p^1 \in \mathbb{P}_1(\mathsf{f})/\mathfrak{R}: \quad \left[\boldsymbol{\nu}, \left(\operatorname{Curl}_{\boldsymbol{\xi}}(p^1)\right)_{|\mathsf{f}}^{\mathcal{E}}\right]_{\mathcal{E},\mathsf{f}} := \sum_{\mathsf{e} \in \partial \mathsf{f}} |\mathsf{e}| p^1(\boldsymbol{\xi}_\mathsf{e}) \sigma_{\mathsf{f},\mathsf{e}} \nu_{\mathsf{e}}. \tag{86}$$

Now, recalling that in the present setting $N_j = (c_j)_{|_f}^{\mathcal{E}}$ and using the discrete Gauss–Green relation (86), we get the alternative derivation of the algebraic consistency condition:

$$\forall \boldsymbol{\nu} \in \mathcal{E}_{|\mathsf{f}}: \quad \boldsymbol{\nu}^{\mathsf{T}} \mathbb{M}_{\mathcal{E},\mathsf{f}} \boldsymbol{N}_{j} = \left[\boldsymbol{\nu}, \left(\boldsymbol{c}_{j} \right)_{|\mathsf{f}}^{\mathcal{E}} \right]_{\mathcal{E},\mathsf{f}} = \left[\boldsymbol{\nu}, \left(\operatorname{Curl}_{\boldsymbol{\xi}}(\boldsymbol{p}_{j}^{1}) \right)_{|\mathsf{f}}^{\mathcal{E}} \right]_{\mathcal{E},\mathsf{f}} \equiv \boldsymbol{\nu}^{\mathsf{T}} \boldsymbol{R}_{j}, \tag{87}$$

where \mathbf{R}_i is defined through the last identity.

Finally, we note that this derivation is related to the construction of the mimetic inner product for discrete fluxes that was proposed in [14]. Definition 3.3 can be identified with the *consistency condition* (*S*2) that is used to derive the MFD scheme for the 2D diffusion equation in [14]. To do so, it is sufficient to re-interpret the 2D curl as the divergence of a 2D rotated field, and the edge degrees of freedom as the fluxes of the rotated field.

3.2.3. 3D consistency relation for face functions

As in the two preceding sub-sections, we choose the matrix \mathbb{N} in the algebraic relation " $\mathbb{M}_{\mathcal{F},\mathbb{P}}\mathbb{N} = \mathbb{R}$ " and search for a suitable matrix \mathbb{R} such that the numerical integration rule (45) holds. The matrix $\mathbb{M}_{\mathcal{F},\mathbb{P}}$ will then be given by Proposition 3.1. An important connection, which is discussed at the end of this sub-section, exists with a discrete Gauss–Green formula. We also remark that the development discussed in this section is related to that in [14] for the mimetic discretization of the elliptic equation.

Let us assume that μ^{-1} is constant on the polyhedron P and consider three constant vectors that form a basis in \Re^3 , i.e., $\mathbf{c}_1 = \mu(1,0,0)^T$, $\mathbf{c}_2 = \mu(0,1,0)^T$, and $\mathbf{c}_3 = \mu(0,0,1)^T$. For each polyhedron P and j = 1,2,3 we require the column vector \mathbf{N}_j , which is the *j*th column of matrix \mathbb{N} , to be equal to the degrees of freedom of \mathbf{c}_i for the faces of P:

$$\mathbf{N}_j = (\mathbf{c}_j)_{|\mathsf{P}}^{\mathcal{F}}.$$
(88)

More precisely, the entries of \mathbb{N} are given by

$$\forall \mathbf{f} \in \partial \mathsf{P}, \quad \mathbf{j} \in \{1, 2, 3\}: \quad \mathbb{N}_{\mathbf{f}, \mathbf{j}} = \frac{1}{|\mathbf{f}|} \int_{\mathbf{f}} \mathbf{n}_{\mathbf{f}} \cdot \mathbf{c}_{\mathbf{j}} dS.$$
(89)

The rest of this section is devoted to the derivation of a proper expression for the columns R_j of matrix \mathbb{R} such that the quadrature rule

$$\int_{\mathsf{P}} \mu^{-1} \boldsymbol{c}_{j} \cdot \boldsymbol{v} dV = \left(\boldsymbol{v}_{|\mathsf{P}}\right)^{\mathsf{T}} \mathbb{M}_{\mathcal{F},\mathsf{P}} \boldsymbol{N}_{j} + |\mathsf{P}| \mathcal{O}(h)$$
(90)

can also take the form

$$\int_{\mathsf{P}} \mu^{-1} \boldsymbol{c}_{j} \cdot \boldsymbol{\nu} dV = \left(\boldsymbol{\nu}_{|\mathsf{P}}^{\mathcal{F}}\right)^{\mathsf{T}} \boldsymbol{R}_{j} + |\mathsf{P}|\mathcal{O}(h)$$
(91)

for every generic vector-valued function $\boldsymbol{\nu}$ defined on P. The comparison between (90) and (91) reveals the first part the algebraic consistency condition, i.e., $\mathbb{M}_{\mathcal{F},P} \mathbb{N} = \mathbb{R}$. The second part is proved below. Now, the matrix $\mathbb{M}_{\mathcal{F},P}$ used in the quadrature formula (45) is given by Proposition 3.1.

Let \boldsymbol{c} be a constant vector. Since μ^{-1} has been taken constant on P, we have that $\boldsymbol{c} = \mu \nabla p^1(\boldsymbol{x})$ where $p^1(\boldsymbol{x}) = \mu^{-1} \boldsymbol{c} \cdot (\boldsymbol{x} - \boldsymbol{x}_{\mathsf{P}})$. Let us define

$$\langle \operatorname{div}(\boldsymbol{v}) \rangle_{\mathsf{P}} = \frac{1}{|\mathsf{P}|} \int_{\mathsf{P}} \operatorname{div}(\boldsymbol{v}) dV.$$
 (92)

From the orthogonality of p^1 to the constant scalar fields, the Cauchy–Schwarz inequality and the fact that for any sufficiently regular function ϕ the cell average $\langle \phi \rangle_{P}$ is a first-order approximation of ϕ on P, we obtain the inequality:

$$\int_{\mathsf{P}} p^{1} \operatorname{div}(\boldsymbol{v}) dV = \int_{\mathsf{P}} p^{1} (\operatorname{div}(\boldsymbol{v}) - \langle \operatorname{div}(\boldsymbol{v}) \rangle_{\mathsf{P}}) dV \leqslant \|p^{1}\|_{(L^{2}(\mathsf{P}))^{3}} \|\operatorname{div}(\boldsymbol{v}) - \langle \operatorname{div}(\boldsymbol{v}) \rangle_{\mathsf{P}}\|_{(L^{2}(\mathsf{P}))^{3}} = |\mathsf{P}|\mathcal{O}(h).$$
(93)

Using the previous expression for the vector field \boldsymbol{c} , we get the following development:

$$\int_{P} \mu^{-1} \boldsymbol{c} \cdot \boldsymbol{v} dV = \int_{P} \nabla p^{1} \cdot \boldsymbol{v} dV \quad \text{(integrate by parts)} \\ = -\int_{P} p^{1} \operatorname{div}(\boldsymbol{v}) dV + \int_{\partial P} p^{1} \boldsymbol{n}_{P} \cdot \boldsymbol{v} dS \quad \text{(use estimate (93))} \\ = |P|\mathcal{O}(h) + \int_{\partial P} p^{1} \boldsymbol{n}_{P} \cdot \boldsymbol{v} dS \quad \text{(split the boundary integral)} \\ = |P|\mathcal{O}(h) + \sum_{f \in \partial P} \int_{f} p^{1} \boldsymbol{n}_{P,f} \cdot \boldsymbol{v} dS. \tag{94}$$

K. Lipnikov et al./Journal of Computational Physics xxx (2010) xxx-xxx

Then, we consider the following first-order approximation for each face integral that appears in the summation term of (94):

$$\int_{f} p^{1} \boldsymbol{n}_{\mathsf{P},\mathsf{f}} \cdot \boldsymbol{\nu} dS = \left(p^{1}(\boldsymbol{x}_{\mathsf{f}}) + \mathcal{O}(h) \right) \int_{f} \boldsymbol{n}_{\mathsf{P},\mathsf{f}} \cdot \boldsymbol{\nu} dS = p^{1}(\boldsymbol{x}_{\mathsf{f}}) |\mathsf{f}| \sigma_{\mathsf{P},\mathsf{f}} \boldsymbol{\nu}_{|\mathsf{f}}^{\mathcal{F}} + \mathcal{O}(h) \int_{f} \boldsymbol{n}_{\mathsf{P},\mathsf{f}} \cdot \boldsymbol{\nu} dS, \tag{95}$$

where we used (17) to introduce $\boldsymbol{v}_{|f}^{\mathcal{F}}$, and the sign $\sigma_{\mathsf{P},\mathsf{f}} = \boldsymbol{n}_{\mathsf{P},\mathsf{f}} \cdot \boldsymbol{n}_{\mathsf{f}}$ to take into account the orientation of f with respect to $\partial \mathsf{P}$. Substituting (95) into (94) yields

$$\int_{\mathsf{P}} \mu^{-1} \boldsymbol{c} \cdot \boldsymbol{\boldsymbol{\nu}} dV = |\mathsf{P}|\mathcal{O}(h) + \sum_{\mathsf{f} \in \partial \mathsf{P}} \left(|\mathsf{f}| \sigma_{\mathsf{P},\mathsf{f}} \boldsymbol{p}^{1}(\boldsymbol{x}_{\mathsf{f}}) \boldsymbol{\boldsymbol{\nu}}_{|\mathsf{f}}^{\mathcal{F}} + \mathcal{O}(h) \int_{\mathsf{f}} \boldsymbol{n}_{\mathsf{P},\mathsf{f}} \cdot \boldsymbol{\boldsymbol{\nu}} dS \right) = |\mathsf{P}|\mathcal{O}(h) + \sum_{\mathsf{f} \in \partial \mathsf{P}} |\mathsf{f}| \sigma_{\mathsf{P},\mathsf{f}} \boldsymbol{p}^{1}(\boldsymbol{x}_{\mathsf{f}}) \boldsymbol{\boldsymbol{\nu}}_{|\mathsf{f}}^{\mathcal{F}}.$$
(96)

Taking c_j instead of c in (96), considering the corresponding scalar polynomial $p_j^1(\mathbf{x})$ and renumbering locally the $m_{\mathcal{F},\mathsf{P}}$ faces of $\partial\mathsf{P}$ from 1 to $m_{\mathcal{F},\mathsf{P}}$ yields

$$\int_{\mathsf{P}} \mu^{-1} \boldsymbol{c}_{j} \cdot \boldsymbol{v} dV = \left(\boldsymbol{v}_{|\mathsf{f}}^{\mathcal{F}}\right)^{\mathsf{T}} \boldsymbol{R}_{j} + |\mathsf{P}|\mathcal{O}(h), \tag{97}$$

where the component of the column vector \mathbf{R}_j associated with the face f, or, equivalently, the (f, j)-entry of the matrix \mathbb{R} is given by

$$\boldsymbol{R}_{f,j} = \mathbb{R}_{f,j} = \sigma_{\mathsf{P},f} |f| \boldsymbol{p}_j^1(\boldsymbol{x}_f).$$
(98)

We are now left to show that $\mathbb{N}^T \mathbb{R}$ is a symmetric and positive definite matrix. As for the previous cases, the crucial point is that the integration formulas (94) and (95) are exact when the vector field \boldsymbol{v} is constant on P. Let us now take \boldsymbol{c}_j instead of \boldsymbol{c} and set $\boldsymbol{v} = \boldsymbol{c}_i$ in (94), and recall that μ is assumed to be constant on P. We obtain the following development:

$$|\mathsf{P}|\boldsymbol{\mu}_{|jj} = |\mathsf{P}| \left(\boldsymbol{\mu}^{-1} \boldsymbol{c}_{i}\right) \cdot \boldsymbol{\mu} \left(\boldsymbol{\mu}^{-1} \boldsymbol{c}_{j}\right) = \int_{\mathsf{P}} \left(\boldsymbol{\mu}^{-1} \boldsymbol{c}_{i}\right) \cdot \boldsymbol{c}_{j} dV = \left(\left(\boldsymbol{c}_{i}\right)_{|\mathsf{P}}^{\mathcal{F}}\right)^{T} \boldsymbol{R}_{j} = \boldsymbol{N}_{i}^{T} \boldsymbol{R}_{j}, \tag{99}$$

from which it follows that $\mathbb{N}^T \mathbb{R} = |\mathsf{P}|\mu$, and, eventually that $\mathbb{N}^T \mathbb{R}$ is a symmetric and positive definite matrix.

The previous derivation is related to a *discrete Gauss–Green formula* or a discrete integration-by-parts formula. Taking $\mathbf{c} = \mu \nabla p^1$ in (96) yields:

$$\int_{\mathsf{P}} \boldsymbol{\nu} \cdot \nabla p^1 dV = |\mathsf{P}|\mathcal{O}(h) + \sum_{\mathsf{f} \in \partial \mathsf{P}} |\mathsf{f}| \sigma_{\mathsf{P},\mathsf{f}} p^1(\boldsymbol{x}_\mathsf{f}) \boldsymbol{v}_{|\mathsf{f}}^{\mathcal{F}}.$$
(100)

Applying the quadrature rule to the integral in the left-hand side and neglecting the O(h) terms, we get the following definition.

Definition 3.4. Let $\mathbb{P}_1(P)/\Re$ denote the space of linear scalar functions orthogonal to the constant scalar fields defined on the polyhedron P. The discrete Gauss–Green relation for the face functions defined on the polyhedron P takes the form:

$$\forall \boldsymbol{\nu} \in \mathcal{F}_{|\mathsf{P}}, \quad \forall \boldsymbol{p}^{1} \in \mathbb{P}_{1}(\mathsf{P})/\Re: \quad \left[\boldsymbol{\nu}, \left(\mu\nabla(\boldsymbol{p}^{1})\right)_{|\mathsf{P}}^{\mathcal{F}}\right]_{\mathcal{F},\mathsf{P}} := \sum_{\mathsf{f} \in \partial\mathsf{P}} |\mathsf{f}|\boldsymbol{p}^{1}(\boldsymbol{x}_{\mathsf{f}})\boldsymbol{\nu}_{\mathsf{f}}. \tag{101}$$

Now, recalling that in the present setting $N_j = (c_j)_{|p}^{\mathcal{F}}$ and using the previous discrete Gauss–Green relation, we get the alternative derivation of the algebraic consistency condition:

$$\forall \boldsymbol{v} \in \mathcal{F}_{|\mathsf{P}}: \quad \boldsymbol{v}^{\mathsf{T}} \mathbb{M}_{\mathcal{F},\mathsf{P}} \boldsymbol{N}_{j} = \left[\boldsymbol{v}, \left(\boldsymbol{c}_{j}\right)_{|\mathsf{P}}^{\mathcal{F}}\right]_{\mathcal{F},\mathsf{P}} = \left[\boldsymbol{v}, \left(\mu \nabla(p_{j}^{1})\right)_{|\mathsf{P}}^{\mathcal{F}}\right]_{\mathcal{F},\mathsf{P}} \equiv \boldsymbol{v}^{\mathsf{T}} \boldsymbol{R}_{j}, \tag{102}$$

where \mathbf{R}_i is defined through the last identity.

3.3. Local implementation

The implementation of the mimetic finite difference method requires formally the construction and solution of linear system (36). To this purpose, we first have to calculate:

- the discrete gradient operator \mathcal{GRAD} in (18) and its matrix representation \mathbb{G} in (27);
- the discrete curl operator \mathcal{CURL} in (19) and its matrix representation \mathbb{C} in (28);
- the edge matrix $M_{\mathcal{E}}$ of the quadrature formula (29);
- the face matrix $M_{\mathcal{F}}$ of the quadrature formula (30).

Then, we have to calculate the two matrices $\mathbb{A} = \mathbb{C}^T \mathbb{M}_F \mathbb{C}$ and $\mathbb{B} = \mathbb{G}^T \mathbb{M}_{\mathcal{E}}$ (see (31)) that form the coefficient matrix of system (36). Even if \mathbb{A} and \mathbb{B} are defined by multiplying global matrices, a local construction is still possible by calculating the four matrices $\mathbb{G}_{|P}, \mathbb{O}_{|P}, \mathbb{M}_{\mathcal{E},P}$ and $\mathbb{M}_{\mathcal{F},P}$ for each polyhedron P, forming the elemental matrices $\mathbb{A}_{|P}$ and $\mathbb{B}_{|P}$ and then assembling all local contributions. We describe this algorithm in the rest of this subsection.

Please cite this article in press as: K. Lipnikov et al., The mimetic finite difference method for the 3D magnetostatic field problems on polyhedral meshes, J. Comput. Phys. (2010), doi:10.1016/j.jcp.2010.09.007

K. Lipnikov et al./Journal of Computational Physics xxx (2010) xxx-xxx

Local construction of the matrix A. For every polyhedron P do the following:

- 1. calculate the matrix \mathbb{N} by using formulas (88) and (89); and the matrix \mathbb{R} by using formula (98);
- 2. calculate the matrix $\mathbb{M}_{\mathcal{F},P}$ by using Proposition 3.1;
- 3. calculate the *local curl matrix* \mathbb{C}_{P} by applying (19) to the faces of the polyhedron P;
- 4. calculate the local matrix $\mathbb{A}_{|P} = (\mathbb{C}_{|P})^T \mathbb{M}_{\mathcal{F},P} \mathbb{C}_{|P}$ by direct multiplication;
- 5. accumulate the local matrices to the global matrix A:

$$\mathbb{A} = \sum_{\mathsf{P} \in \mathcal{T}_h} \left(\mathbb{S}_{\mathcal{F},\mathsf{P}} \right)^T \mathbb{A}_{|\mathsf{P}} \mathbb{S}_{\mathcal{F},\mathsf{P}}$$

where $\mathbb{S}_{\mathcal{F},P}$ is the same as in (44).

Local construction of the matrix B and of the right-hand side of (36). For every polyhedron P do the following:

- 1. calculate the (polyhedron) matrix \mathbb{N} by using formulas (52) and (53); and the (polyhedron) matrix \mathbb{R} by accumulating the face contributions as follows. For any face in ∂P :
 - (a) calculate the (face) matrix \mathbb{N} by using formulas (73) and (74); the (face) matrix \mathbb{R} by using formula (83);
 - (b) calculate the matrix $\mathbb{M}_{\mathcal{E},f}$ by Proposition 3.1;
- (c) accumulate the face contribution to the (polyhedron) matrix \mathbb{R} in accordance with (67);
- 2. calculate matrix $M_{\mathcal{E},P}$ by Proposition 3.1;
- 3. calculate the *local gradient matrix* $\mathbb{G}_{|P}$ by applying (18) to the edges of the polyhedron P;
- 4. calculate the local matrix $\mathbb{B}_{|P} = \mathbb{G}_{|P}^T \mathbb{M}_{\mathcal{E},P}$ and the local right-hand side vector $\mathbb{M}_{\mathcal{E},P} \mathbf{J}_{|P}^{\mathcal{E}}$ by direct multiplication;
- 5. accumulate the local contribution to the global matrix $\mathbb B$ by using the formulas:

$$\mathbb{B} = \sum_{\mathsf{P}\in\mathcal{T}_{h}} (\mathbb{S}_{\mathcal{E},\mathsf{P}})^{T} \mathbb{B}_{|\mathsf{P}} \mathbb{S}_{\mathcal{N},\mathsf{P}}, \tag{104}$$
$$\mathbb{M}_{\mathcal{E}} \boldsymbol{J}^{\mathcal{E}} = \sum_{\mathsf{P}\in\mathcal{T}_{h}} (\mathbb{S}_{\mathcal{E},\mathsf{P}})^{T} \mathbb{M}_{\mathcal{E},\mathsf{P}} \boldsymbol{J}_{|\mathsf{P}}^{\mathcal{E}}. \tag{105}$$

In both equations, $\mathbb{S}_{\mathcal{E},\mathsf{P}}$ is the same as in (43), and $\mathbb{S}_{\mathcal{N},\mathsf{P}}$ is the restriction matrix that extracts the degrees of freedom for the nodes of P from a node function of \mathcal{N} . Note that $\mathbb{S}_{\mathcal{N},\mathsf{P}}$ has size $m_{\mathcal{N},\mathsf{P}} \times m_{\mathcal{N}}$, where $m_{\mathcal{N},\mathsf{P}}$ is the number of vertices in P .

4. Well-posedness of the method

In this section we investigate the well-posedness of this mimetic finite difference method. The existence and uniqueness of the mimetic solution (\boldsymbol{u}_h, p_h) is proved in Theorem 4.1. The proof is based on the fact that an edge function \boldsymbol{u} can have zero discrete curl, i.e., $CURL(\boldsymbol{u}) = 0$, if and only if \boldsymbol{u} is the discrete gradient of a node function. This property is proved in Lemma 4.1 for simply connected meshes.

Definition 4.1. We say that \mathcal{T}_h is a simply connected mesh if for any closed path χ without inner loops formed by a subset of the mesh edges E there exists a subset F_{χ} of the mesh faces F such that for any mesh edge e that belongs to a face f of F_{χ} there are only two possible cases: either e belongs to χ or there is another face \tilde{f} in F_{χ} that shares this edge.

In a simply connected mesh, it is always possible to connect the edges of a closed path through the mesh faces by forming a "discrete" surface that has the given path as boundary and *does not have any inner hole*. Fig. 2 shows two possible situations. In both plots of this figure, the edges of the mesh path χ (here displayed as a sequence of arrows) are connected through an internal mesh surface using mesh faces. No holes are present in the left plot, while, in the right plot, the discrete surface contains an inner hole delimited by the gray boundary. The surface considered in the right plot cannot be used to qualify the mesh as simply connected. Note that in two dimensions, a 2D mesh containing a portion depicted in the right plot can never be simply connected, while in three dimensions other discrete surfaces may exist that connect the edges of χ . It is easy to realize that the property of the mesh of being simply connected reflects the topological property of the domain of being *contractable*. For a contractable domain, any reasonable mesh is expected to be simply connected. Thus, the requirement of being simply connected does not introduce a new constraint on the admissible meshes that are used to formulate the current mimetic discretization. Note also that any *logically rectangular or cubic* mesh in the sense of Hyman–Shashkov [30–32] is simply connected. Lemma 4.1 is an extension of a similar result for logically rectangular or cubic meshes [30–32] to unstructured polyhedral meshes.

Lemma 4.1. Let \mathcal{T}_h be a simply connected mesh, and $\mathbf{u} \in \mathcal{E}$ be an edge function. Then, $\mathcal{CURL}(\mathbf{u}) = 0$ if and only if there exists a node function $q \in \mathcal{N}$ such that $\mathbf{u} = \mathcal{GRAD}(q)$. The node function q is unique once its value in any node has been set.

Proof. Let $\boldsymbol{u} = \mathcal{GRAD}(q)$ for some node function q. Then, Remark 3.2 implies that $\mathcal{CURL}(\boldsymbol{u}) = 0$.

Please cite this article in press as: K. Lipnikov et al., The mimetic finite difference method for the 3D magnetostatic field problems on polyhedral meshes, J. Comput. Phys. (2010), doi:10.1016/j.jcp.2010.09.007

(103)

K. Lipnikov et al./Journal of Computational Physics xxx (2010) xxx-xxx



Fig. 2. Admissible (a) and non-admissible (b) discrete surfaces according to the requirements of Definition 4.1. In both left and right plots the closed path that bounds the discrete surface is formed by the edges indicated by the straight arrows that connect the sequence of nodes with a circle. The closed path in plot (a) is equal to the sum of the elementary circuits of its inner mesh faces while in plot (b) the most inner circuit, here denoted by the dashed polygonal line, must also be added.

Let $\mathcal{CURL}(u) = 0$. Let $\chi \subset E$ be any subset of the mesh edges that form a *closed* path without loops. Condition $\mathcal{CURL}(u) = 0$ implies that

$$\sum_{e\in\chi} |e|\sigma_{\chi,e}u_e = 0, \tag{106}$$

where $\sigma_{\chi,e}$ is the sign that takes into account the orientation of $e \in \chi \cap \partial f$ with respect to the orientation of the face f, and u_e is the value of u attached to edge e. To prove this fact, let us consider the subset F_{χ} of the mesh faces F that connect the edges of χ in accordance with Definition 4.1. It is easy to see that

$$0 = \sum_{f \in F_{\chi}} |f| (\mathcal{CURL}(\boldsymbol{u}))_{f} = \sum_{f \in F_{\chi}} \sum_{e \in \partial f} \sigma_{f,e} |e| u_{e} = \sum_{e \in \chi} |e| \sigma_{\chi,e} u_{e},$$
(107)

because any "internal" edge of F_{χ} gives two contributions of opposite sign that mutually eliminates; see again Fig. 2. It is worth pointing out that Eq. (107) is just a discrete version of the Stokes theorem ("curl" theorem) applied to the "discrete surface" F_{χ} with boundary χ . Now, we select a node \tilde{v} of the mesh \mathcal{T}_h and we set a real value for this node, i.e., $q_{\tilde{v}}$. Then, for any other node of the mesh we consider an open path χ_v that connects \tilde{v} to v and define the value of the nodal function qattached to v by the formula:

$$q_{v} = q_{\bar{v}} + \sum_{e \in \chi_{v}} |e| \sigma_{\chi_{v},e} u_{e}.$$
(108)

The crucial point of this definition is that the value of q_v for any $v \in \mathcal{T}_h$ provided by (108) is actually *independent of the path* χ_v . In fact, if we consider a different path χ'_v still connecting \tilde{v} to v, the union of this two paths forms the closed path $\chi = \chi_v \cup \chi'_v$. Eq. (106) implies that

$$\sum_{e \in \chi_{\nu}} |e| \sigma_{\chi_{\nu},e} u_{e} = \sum_{e \in \chi_{\nu}'} |e| \sigma_{\chi_{\nu}',e} u_{e},$$
(109)

where the signs $\sigma_{\chi_v,e}$ and $\sigma_{\chi',e}$ take into account the fact the two paths χ_v and χ'_v are run in opposite sense. The derivation of node function q in (108) implies that

$$\forall \mathsf{e} = (\mathsf{v}_1, \mathsf{v}_2) \in \mathsf{E}, \quad u_\mathsf{e} = \frac{q_{\mathsf{v}_2} - q_{\mathsf{v}_1}}{|\mathsf{e}|} = \left(\mathcal{GRAD}(q)\right)_\mathsf{e},\tag{110}$$

where the edge e is oriented from node v_1 to node v_2 (so that $\sigma_{\chi_v,e} = 1$) and q_{v_1} and q_{v_2} are the values that q takes at the two nodes v_1 and v_2 . Therefore, the node function q is uniquely determined by the edge values u_e and by the value $q_{\tilde{v}}$ at the first mesh node \tilde{v} . Note that this latter can always be taken on the boundary and set equal to zero. \Box

The well-posedness of our numerical method relies on the following result.

Theorem 4.1. Let T_h be a simply connected mesh. Then, problem (34) and (35) admits a unique solution.

Proof. Let $(\boldsymbol{u}_h, p_h) \in \mathcal{E}_0 \times \mathcal{N}_0$ be the solution of the homogeneous problem obtained by imposing $\boldsymbol{J} = 0$ in (34). We will prove that $\boldsymbol{u}_h = 0$ and $p_h = 0$. To this purpose, let us consider Eq. (34) with $\boldsymbol{v} = \boldsymbol{u}_h$ and Eq. (35) with $\boldsymbol{q} = p_h$. We get

$$\boldsymbol{u}_{h}^{T} \mathbb{A} \boldsymbol{u}_{h} + \boldsymbol{u}_{h}^{T} \mathbb{B}^{T} \boldsymbol{p}_{h} = (\boldsymbol{u}_{h}^{T} \mathbb{C}^{T}) \mathbb{M}_{\mathcal{F}}(\mathbb{C} \boldsymbol{u}_{h}) = \boldsymbol{0},$$

which implies that $\mathbb{C}\boldsymbol{u}_h = 0$ because $\mathbb{M}_{\mathcal{F}}$ is a positive definite matrix. Substituting $\mathbb{C}\boldsymbol{u}_h = 0$ and $\mathbb{B}^T = \mathbb{M}_{\mathcal{E}}\mathbb{G}$ into Eq. (34) yields:

K. Lipnikov et al./Journal of Computational Physics xxx (2010) xxx-xxx

$$\boldsymbol{v}^{T}\mathbb{C}^{T}\mathbb{M}_{\mathcal{F}}\mathbb{C}\boldsymbol{u}_{h}+\boldsymbol{v}^{T}\mathbb{M}_{\mathcal{E}}\mathbb{G}p_{h}=\boldsymbol{v}^{T}\mathbb{M}_{\mathcal{E}}\mathbb{G}p_{h}=0,$$

17

which implies that $\mathbb{G}p_h = 0$ because $\mathbb{M}_{\mathcal{E}}$ is a non-singular matrix and \boldsymbol{v} is arbitrary. Condition $\mathbb{G}p_h = 0$ implies that p_h is a *constant node function*, i.e., it has the same value in every mesh vertex. From the homogeneous Dirichlet conditions it immediately follows that $p_h = 0$. Then, we observe that condition $\mathbb{C}\boldsymbol{u}_h = 0$ and the result of Lemma 4.1 imply that there exists a node function $q_h \in \mathcal{N}$ such that $\boldsymbol{u}_h = \mathbb{G}q_h$. From Eq. (35) with $q = q_h$ and $\mathbb{B}^T = \mathbb{M}_{\mathcal{E}}\mathbb{G}$ we have

$$\boldsymbol{q}_{h}^{T} \mathbb{B} \boldsymbol{u}_{h} = \boldsymbol{q}_{h}^{T} \mathbb{G}^{T} \mathbb{M}_{\mathcal{E}} \boldsymbol{u}_{h} = \left(\mathbb{G} \boldsymbol{q}_{h}\right)^{T} \mathbb{M}_{\mathcal{E}} \mathbb{G} \boldsymbol{q}_{h} = \boldsymbol{0}$$

$$(111)$$

and, thus, that $\mathbb{G}q_h = 0$ because $\mathbb{M}_{\mathcal{E}}$ is a positive definite matrix. Thus, we get $u_h = \mathbb{G}q_h = 0$. This proves the assertion of the theorem. \Box

Remark 4.1. The result could obviously have been obtained also by the usual theory of linear saddle-point problems. Indeed, we should only show that in (34) and (35) the matrix $\mathbb{B}^T \equiv \mathbb{M}_{\mathcal{E}} \mathbb{G}$ is injective and that the matrix $\mathbb{A} \equiv \mathbb{C}^T \mathbb{M}_{\mathcal{F}} \mathbb{C}$ is positive definite on the kernel of \mathbb{B} , that amounts to say: $\boldsymbol{u}_h^T \mathbb{C}^T \mathbb{M}_{\mathcal{F}} \mathbb{C} \boldsymbol{u}_h > 0$ whenever $\mathbb{G}^T \mathbb{M}_{\mathcal{E}} \boldsymbol{u}_h = 0$ and $\boldsymbol{u}_h \neq 0$. The amount of work, however, would have been only slightly smaller.

5. Numerical experiments

5.1. Accuracy tests

We investigate the accuracy and the convergence behavior of the MFD method (25) and (26) by numerically solving problem (13) and (14) on the domain $\Omega =]0, 1[\times]0, 1[\times]0, 1[$ for the magnetic permeabilities

$$\mu_1^{-1}(x, y, z) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix},$$
(112)

$$\mu_2^{-1}(\mathbf{x}, \mathbf{y}, \mathbf{z}) = \begin{pmatrix} 1 + y^2 + z^2 & -xy & -xz \\ -xy & 1 + x^2 + z^2 & -yz \\ -xz & -yz & 1 + x^2 + y^2 \end{pmatrix}.$$
(113)

The boundary function **g** and the right-hand side source term **J** are set in accordance with the exact solution p(x, y, z) = 0 and

Table 1

Parameters of mesh families M_1-M_5 that are used for accuracy test cases. The symbol k is the refinement level, n is the refinement parameter (where appropriate), m_P is the number of polyhedric cells of the mesh, m_F is the number of faces, m_E is the number of edges, m_N is the number of vertices, and h is the mesh size parameter.

Mesh	k	n	$m_{ m P}$	$m_{\mathcal{F}}$	$m_{\mathcal{E}}$	m_N	h
M1	0	4	64	240	300	125	$2.500 \ 10^{-1}$
	1	8	512	1728	1944	729	$1.250 \ 10^{-1}$
	2	16	4096	13056	13872	4913	6.25010^{-2}
	3	32	32768	101376	104544	35937	$3.125 \ 10^{-2}$
M2	0	4	120	444	546	233	$2.500 \ 10^{-1}$
	1	8	960	3216	3588	1333	$1.250 \ 10^{-1}$
	2	16	7680	24384	25800	9097	6.250 10 ⁻²
	3	32	61440	189696	195216	66961	$3.125 \ 10^{-2}$
Мз	0	5	216	1002	1415	630	$2.158 \ 10^{-1}$
2	1	10	1210	5331	7200	3080	$1.348 \ 10^{-1}$
	2	15	4096	17552	23145	9690	$1.023 \ 10^{-1}$
	3	20	8820	37261	48600	20160	8.186 10 ⁻²
	4	25	17576	73502	95075	39150	6.297 10 ⁻²
	5	30	28830	119791	154200	63240	5.457 10 ⁻²
	6	35	46656	192852	247205	101010	$4.548 \ 10^{-2}$
Ma	1		888	2865	3153	1177	$2.706 \ 10^{-1}$
-	2		11444	35451	37495	13489	$1.277 \ 10^{-1}$
	3		61440	189696	195216	66961	$6.487 \ 10^{-2}$
M5	0	5	264	1647	2768	1386	$2.798 \ 10^{-1}$
	1	10	1516	10253	17476	8740	$1.599 \ 10^{-1}$
	2	15	4490	31464	53950	26977	$1.092 \ 10^{-1}$
	3	20	9934	71139	122412	61208	7.934 10 ⁻²
	4	25	18609	134955	232694	116349	$6.427 \ 10^{-2}$

K. Lipnikov et al./Journal of Computational Physics xxx (2010) xxx-xxx

$$\mathbf{u}(x,y,z) = \begin{pmatrix} 2\pi \ r(x)\sin(2\pi y)\cos(2\pi z) \\ -r'(x)\cos(2\pi y)\cos(2\pi z) \\ -2r'(x)\sin(2\pi y)\sin(2\pi z) \end{pmatrix}, \quad r(x) = x^3.$$
(114)

To measure the quality of the numerical solution we use the two mesh-dependent norms:

$$\forall \boldsymbol{v} \in \mathcal{E} : \quad |||\boldsymbol{v}|||_{\mathcal{E}}^{2} = [\boldsymbol{v}, \boldsymbol{v}]_{\mathcal{E}} = \boldsymbol{v}^{\mathsf{T}} \mathbb{M}_{\mathcal{E}} \boldsymbol{v},$$

$$\forall \boldsymbol{v} \in \mathcal{F} : \quad |||\boldsymbol{v}|||_{\mathcal{F}}^{2} = [\boldsymbol{v}, \boldsymbol{v}]_{\mathcal{F}} = \boldsymbol{v}^{\mathsf{T}} \mathbb{M}_{\mathcal{F}} \boldsymbol{v}.$$

$$(115)$$

The relative error for the approximation to vector fields **u** and curl(**u**) provided by u_h and $CURL(u_h)$ are given by:

$$\operatorname{Error}(\boldsymbol{u}) = \frac{\||\boldsymbol{u}_h - \boldsymbol{u}^{\varepsilon}\|\|_{\mathcal{E}}}{\||\boldsymbol{u}^{\varepsilon}\|\|_{\mathcal{E}}},$$

$$\operatorname{Error}(\operatorname{curl}(\boldsymbol{u})) = \frac{\||\mathcal{CURL}(\boldsymbol{u}_h - \boldsymbol{u}^{\varepsilon})\|\|_{\mathcal{F}}}{\||\boldsymbol{u}^{\varepsilon}|\|_{\mathcal{F}}}.$$
(117)

$$\operatorname{rror}(\operatorname{Curl}(\boldsymbol{u})) = \frac{||\mathcal{CURL}(\boldsymbol{u}_h - \boldsymbol{u}^{\varepsilon})|||_{\mathcal{F}}}{|||\mathcal{CURL}(\boldsymbol{u}^{\varepsilon})|||_{\mathcal{F}}}.$$
(118)



Fig. 3. Accuracy tests on regular meshes. Plots (a) and (b) display the first mesh (left) and the second mesh (right) of the mesh sequence M1; mesh parameters are reported in Table 1. Plots (c) and (d) show the approximation errors for M1 using the constant magnetic permeability given by (112) and the variable magnetic permeability given by (113). In each plot, we report $E_{rror}(u)$ (circles), see Eq. (117), and $E_{rror}(curl(u))$ (squares), see Eq. (118), and two straight lines showing the theoretical slopes $\mathcal{O}(h)$ (labeled by **1**) and $\mathcal{O}(h^2)$ (labeled by **2**).

Please cite this article in press as: K. Lipnikov et al., The mimetic finite difference method for the 3D magnetostatic field problems on polyhedral meshes, J. Comput. Phys. (2010), doi:10.1016/j.jcp.2010.09.007

K. Lipnikov et al./Journal of Computational Physics xxx (2010) xxx-xxx

Relative approximation errors are measured for a sequence of refined meshes that belong to the five mesh families M_1-M_5 , which are characterized in Table 1. The numerical results for these mesh sequences are shown in the five Figs. 3–7. In each figure, plots (a) and (b) show the first two meshes used for the calculations. Plots (c) and (d) show the approximation errors measured through formulas (117) and (118) for calculations using, respectively, the constant magnetic permeability (112) and the variable magnetic permeability (113). For the sake of comparison, plots (c) and (d) also show the "theoretical" linear and quadratic slopes, respectively labeled by 1 and 2, in the bottom-left corner.

Each mesh in M_1 is formed by a regular $n \times n \times n$ decomposition of Ω into cubic cells. The first mesh of this sequence corresponds to n = 4 and the parameter n is doubled at each mesh refinement; thus, the first two meshes displayed in plots (a) and (b) of Fig. 3 correspond to n = 4 and n = 8. We consider the mesh family M_1 to investigate the presence of a superconvergence effect that seems to be characteristic of mimetic approximations when using regular mesh partitionings. Such an effect has been already observed in the mimetic approximation of the flux in diffusion problems [14] and convection–diffusion problems [17]. In the present calculations, the superconvergence rate is clearly visible and is reflected by the quadratic slopes of the error curves.

Each mesh in M_2 is still formed by regular cubic cells, but now it also features a refinement in one corner of domain Ω , which is obtained by locally doubling the parameter n. We use mesh family M_2 to investigate the accuracy of the mimetic



Fig. 4. Accuracy tests on locally refined meshes. Plots (a) and (b) display the first mesh (left) and the second mesh (right) of the mesh sequence M_2 ; mesh parameters are reported in Table 1. Plots (c) and (d) show the approximation errors for M_2 using the constant magnetic permeability given by (112) and the variable magnetic permeability given by (113). In each plot, we report Error(\mathbf{u}) (circles), see Eq. (117), and Error($\mathbf{curl}(\mathbf{u})$) (squares), see Eq. (118), and two straight lines showing the theoretical slopes $\mathcal{O}(h)$ (labeled by 1) and $\mathcal{O}(h^2)$ (labeled by 2).

K. Lipnikov et al./Journal of Computational Physics xxx (2010) xxx-xxx



Fig. 5. Accuracy tests on prismatic meshes. Each mesh is generated by orthogonally extruding a mainly-hexagonal 2D base mesh and cutting the extrusion in the vertical direction by using a set of almost equidistant and randomly tilted planes. Plots (a) and (b) display the first mesh (left) and the second mesh (right) of the mesh sequence M_3 . In both plots a part of the mesh around the point (1,1,1) has been removed to show the interior; mesh parameters are reported in Table 1. Note that the prism faces at the domain boundaries orthogonal to the X - Y reference plane are degenerate, i.e., are formed by two parallel sub-faces. Plots (c) and (d) show the approximation errors for M_3 using the constant magnetic permeability given by (112) and the variable magnetic permeability given by (113). In each plot, we report Error(u) (circles), see Eq. (117), and Error(curl(u)) (squares), see Eq. (118), and two straight lines showing the theoretical slopes $\mathcal{O}(h)$ (labeled by 1) and $\mathcal{O}(h^2)$ (labeled by 2).

formulation when applied to a sequence of non-conforming mesh partitions. It is worth mentioning that one of the most remarkable advantages offered by mimetic formulations is the capability of treating non-conforming meshes without the introduction of hanging nodes that would require a special treatment in the scheme. A direct comparison of the numerical results in plots (c) and (d) of Fig. 4 to the "theoretical" slopes in the bottom-left corners reveals a quadratic convergence rate. Nonetheless, since a thorough theoretical comprehension of superconvergence of mimetic schemes still misses, we cannot guarantee that this quadratic rate will persist if further refinements are considered.

Each mesh in M_3 is composed by orthogonal prisms with a polygonal base. These prisms are obtained by extruding a 2D polygonal base mesh in the *xy* reference plane along direction *z* onto a set of almost equispaced horizontal layers. The 2D base mesh is built in two steps. First, we generate the Voronoi cells for the $(n + 1) \times (n + 1)$ set of 2D points (x_{ij}, y_{ij}) given by

$$\begin{aligned} x_{ij} &= \xi_i + (1/10) \, \sin(2\pi\xi_i) \sin(2\pi\eta_j), \quad i = 0, \dots, n, \\ y_{ij} &= \eta_i + (1/10) \, \sin(2\pi\xi_j) \sin(2\pi\eta_j), \quad j = 0, \dots, n, \end{aligned}$$
(119)



Fig. 6. Accuracy tests on unstructured hexahedral meshes. Each mesh is obtained by partitioning into four hexahedrons the tetrahedral cells of an underlying Delaunay mesh provided by the mesh generator tetgen. Plots (a) and (b) display the first mesh (left) and the second mesh (right) of the mesh sequence M_4 . In both plots a part of the mesh around the point (11,1) has been removed to show the interior; mesh parameters are reported in Table 1. Plots (c) and (d) show the approximation errors for M_4 using the constant magnetic permeability given by (112) and the variable magnetic permeability given by (113). In each plot, we report $Error(\mathbf{u})$ (circles), see Eq. (117), and $Error(curl(\mathbf{u}))$ (squares), see Eq. (118), and two straight lines showing the theoretical slopes $\mathcal{O}(h)$ (labeled by **1**) and $\mathcal{O}(h^2)$ (labeled by **2**).

where $\xi_i = ih$, $\eta_j = jh$ and h = 1/n. Second, we move each interior mesh node v to the center of mass of a triangle formed by the centers of the three Voronoi cells sharing v. To generate 3D prisms, we consider n horizontal layers for even n and n + 1 horizontal planes for odd n, which are equidistant and parallel to the reference *xy* plane. Then, to get a set of oblique layers, the slope of each plane is randomly modified by an amount that is small enough to ensure that the layers do not intersect inside Ω . The first two meshes, displayed in plots (a) and (b) in Fig. 5, correspond to n = 5 and n = 10. The convergence rate that is visible in plots (c) and (d) is initially close to 2 and, then, approaches 1, thus confirming the fact that this mimetic scheme is linearly convergent on a sequence of very general meshes.

Each mesh in M_4 is composed by irregularly shaped hexahedral cells that are not located on a *logically cubic* network. Each mesh is provided by decomposing the tetrahedral cells of an underlying Delaunay tetrahedralization of Ω into four hexahedrons. The Delaunay meshes are provided by the mesh generator tetgen. As shown by plots (a) and (b) in Fig. 6, a refined



Fig. 7. Accuracy tests on randomly generated Voronoi meshes. Each mesh is obtained by taking the Voronoi diagrams of a set of points that are initially regularly distributed over the domain and then randomly displaced. The Voronoi cells that overlap the domain boundary are clipped using the Sutherland–Hodgman algorithm. Plots (a) and (b) display the first mesh (left) and the second mesh (right) of the mesh sequence M_5 . In both plots a part of the mesh around the point (1, 1, 1) has been removed to show the interior; mesh parameters are reported in Table 1. Plots (c) and (d) show the approximation errors for M_5 using the constant magnetic permeability given by (112) and the variable magnetic permeability given by (113). In each plot, we report Error(\boldsymbol{u}) (circles), see Eq. (117), and Error(curl(\boldsymbol{u})) (squares), see Eq. (118), and two straight lines showing the theoretical slopes $\mathcal{O}(h)$ (labeled by **1**) and $\mathcal{O}(h^2)$ (labeled by **2**).

mesh is not nested into one of the coarser meshes. A linear convergence rate is clearly shown by the numerical errors (117) and (118) reported in plots (c) and (d) of Fig. 6, i.e., for both constant and variable magnetic permeabilities.

Each mesh in M_5 is composed by irregularly shaped polyhedral cells. These polyhedral cells are the Voronoi diagrams of a set of points that are almost regularly distributed over the domain Ω . The set of points is generated through the following two steps. First, we take the barycenters of the cubic cells of an $n \times n \times n$ regular decomposition of domain Ω ; second, each point is moved to a new position inside the corresponding cell by a random displacement. The amount of the displacement is small with respect to 1/n, the size of the underlying cubic cells, in order to avoid pathological situations like too small cells, or too small faces, or too small edges, after the Voronoi mesh construction is performed. The Voronoi cells that extend over the domain boundary are clipped at the boundary surfaces by using the Sutherland–Hodgman algorithm [45]. The first two meshes of the mesh sequence M_5 , which are displayed in plots (a) and (b) of Fig. 7, are generated by taking n = 5 and n = 10, while the two remaining meshes are generated by taking n = 15 and n = 20. As for mesh sequences M_3 and M_4 , it is evident that the refined meshes of M_5 are not nested inside coarser meshes. Approximation errors, measured through formulas (117)

K. Lipnikov et al./Journal of Computational Physics xxx (2010) xxx-xxx



Fig. 8. Left picture shows trace of the computational mesh for the C-shaped magnet. Right picture shows orientation of the magnetic induction **B** inside the magnet.

and (118), are displayed in plots (c) and (d) and the error curves display a linear convergence rate for calculations using both constant and variable magnetic permeabilities.

5.2. C-shape magnet test

In this section we consider a C-shaped electromagnet. The model of the magnet consists of a copper slab wrapped around the core of a ferromagnetic material (see Fig. 8). The core is a cylinder of electric steel bent to form a C-shape. The core enhances the magnetic field produced by the circular current **J** running in the copper. We use $\mathbf{J} = (-z_0 - z, 0, \mathbf{x} - \mathbf{x}_0)^T$ where \mathbf{x}_0 and z_0 are the coordinates of a y-line which is the center of the steel cylinder. The relative permeabilities of copper and electrical steel are $\mu_c = 1 - 6.4 \cdot 10^{-6}$ and $\mu_s = 4000$, respectively.

The radius of the cylindrical core is 0.5 and thickness of the copper is 0.5. The magnet is embedded into a box filled with air and the homogeneous Dirichlet boundary condition u = 0 is imposed on the box boundary. The distance between the core and the box boundary is about 1. The model is meshed with a quasi-uniform hexahedral mesh (using package CUBIT) with about 50,000 elements and points. The trace of the computational mesh is shown in Fig. 8. The size of the algebraic problem is 206,813. Note that a tetrahedrization of this mesh will approximately double the number of unknowns.

The right picture in Fig. 8 shows the magnetic induction $\boldsymbol{B} = \operatorname{curl}(\boldsymbol{u})$. The arrows plotted at mesh points indicates the expected alignment of the magnetic field with the ferromagnetic core.

6. Conclusion

In this paper, we proposed an MFD method that extends the mimetic formulation to the numerical treatment of magnetostatic field problems. In particular, we developed an MFD method for calculating the magnetic vector potential \boldsymbol{u} that satisfies the solenoidal condition div(\boldsymbol{u}) = 0. Our mimetic formulation uses degrees of freedom attached to the vertices and edges, and employs natural discrete operators that mimic the curl and the gradient operator of the differential setting. Using the discrete curl and gradient operators and two suitable quadrature rules for the numerical discretization of volume integrals on the computational domain, we provide a numerical discretization of the *div–curl* variational formulation of magnetostatics. These quadrature rules use the edge and face degrees of freedom, and the resulting inner product takes the form of a vector–matrix–vector multiplication where the matrix is derived from an *algebraic consistency condition* that generalizes the usual L^2 inner product construction of the MFD method. We proved the existence and uniqueness of the numerical solution by means of an argument that generalizes the concept of *logically rectangular or cubic meshes* by Hyman and Shashkov to the case of unstructured polyhedral meshes. Finally, the accuracy of the method is shown by numerically solving a set of academic problems and then applying it to a real engineering problem.

Acknowledgments

This work was carried out under the auspices of the National Nuclear Security Administration of the US Department of Energy at Los Alamos National Laboratory under Contract No. DE-AC52-06NA25396 and the DOE Office of Science Advanced

24

K. Lipnikov et al./Journal of Computational Physics xxx (2010) xxx-xxx

Scientific Computing Research (ASCR) Program in Applied Mathematics Research. The second author thanks the 2009 Short Term Mobility Program of Italian Consiglio Nazionale delle Ricerche (CNR). The third author thanks the PRIN-2008 research program of Italian MIUR.

References

- J.E. Aarnes, S. Krogstad, K.-A. Lie, Multiscale mixed/mimetic methods on corner-point grids, Comput. Geosci. (Special Issue on Multiscale Methods) 12 (2008) 297–315.
- [2] L. Beirão da Veiga, A residual based error estimator for the mimetic finite difference method, Numer. Math. 108 (3) (2008) 387-406.
- [3] L. Beirão da Veiga, A mimetic finite difference method for linear elasticity, M2AN: Math. Model. Numer. Anal. 44 (2) (2010) 231–250.
- [4] L. Beirão da Veiga, K. Lipnikov, G. Manzini, Convergence analysis of the high-order mimetic finite difference method, Numer. Math. 113 (3) (2009) 325– 356.
- [5] L. Beirão da Veiga, G. Manzini, A higher-order formulation of the mimetic finite difference method, SIAM, J. Sci. Comput. 31 (1) (2008) 732–760.
- [6] L. Beirão da Veiga, G. Manzini, An a-posteriori error estimator for the mimetic finite difference approximation of elliptic problems, Int. J. Numer. Meth. Eng. 76 (11) (2008) 1696–1723.
- [7] P. Bochev, J.M. Hyman, Principle of mimetic discretizations of differential operators, in: D. Arnold, P. Bochev, R. Lehoucq, R. Nicolaides, M. Shashkov (Eds.), Compatible discretizations, Proceedings of IMA hot topics workshop on compatible discretizations, IMA, vol. 142, Springer-Verlag, 2006.
- [8] A. Bossavit, Whitney forms: a class of finite elements for three dimensional computation in electromagnetism, IEE Proc. 135 (1988) 493–500.
- [9] F. Brezzi, A. Buffa, Innovative mimetic discretizations for electromagnetic problems, J. Comput. Appl. Math. 234 (6) (2010) 1980–1987.
- [10] F. Brezzi, A. Buffa, K. Lipnikov, Mimetic finite differences for elliptic problems, M2AN: Math. Model. Numer. Anal. 43 (2009) 277-295.
- [11] F. Brezzi, M. Fortin, Mixed and Hybrid Finite Element Methods, Springer-Verlag, New York, 1991.
- [12] F. Brezzi, K. Lipnikov, M. Shashkov, Convergence of mimetic finite difference method for diffusion problems on polyhedral meshes, SIAM J. Numer. Anal. 43 (2005) 1872–1896.
- [13] F. Brezzi, K. Lipnikov, M. Shashkov, V. Simoncini, A new discretization methodology for diffusion problems on generalized polyhedral meshes, Comput. Methods Appl. Mech. Eng. 196 (2007) 3682–3692.
- [14] F. Brezzi, K. Lipnikov, V. Simoncini, A family of mimetic finite difference methods on polygonal and polyhedral meshes, Math. Models Methods Appl. Sci. 15 (2005) 1533–1553.
- [15] J.C. Campbell, M.J. Shashkov, A tensor artificial viscosity using a mimetic finite difference algorithm, J. Comput. Phys. 172 (2) (2001) 739-765.
- [16] A. Cangiani, G. Manzini, Flux reconstruction and pressure post-processing in mimetic finite difference methods, Comput. Methods Appl. Mech. Eng. 197 (9–12) (2008) 933–945.
- [17] A. Cangiani, G. Manzini, A. Russo, Convergence analysis of the mimetic finite difference method for elliptic problems, SIAM J. Numer. Anal. 47 (4) (2009) 2612–2637.
- [18] P.G. Ciarlet, The Finite Element Method for Elliptic Problems, North-Holland, 1978.
- [19] Y. Coudière, F. Hubert, A 3D discrete duality finite volume method for non linear elliptic problem, in: A. Handlovicov, P. Frolkovic, K. Mikula, D. Sevcovic (Eds.), Proceedings of Algoritmy 2009, 2009, pp. 51–60.
- [20] Y. Coudière, G. Manzini, The discrete duality finite volume method for convection-diffusion problems, SIAM J. Numer. Anal. 47 (6) (2010) 4163-4192.
- [21] S. Delcourte, K. Domelevo, P. Omnes, A discrete duality finite volume approach to Hodge decomposition and div-curl problems on almost arbitrary two-dimensional meshes, SIAM J. Numer. Anal. 45 (3) (2007) 1142-1174.
- [22] J. Droniou, R. Eymard, A mixed finite volume scheme for anisotropic diffusion problems on any grid, Numer. Math. 105 (1) (2006) 35–71.
- [23] J. Droniou, R. Eymard, T. Gallouët, R. Herbin, A unified approach to mimetic finite difference, hybrid finite volume and mixed finite volume methods, Math. Models Methods Appl. Sci. (M3AS) 20 (2) (2010) 265–295.
- [24] R. Eymard, T. Gallouët, R. Herbin, Discretization of heterogeneous and anisotropic diffusion problems on general non-conforming meshes, SUSHI: a scheme using stabilisation and hybrid interfaces, 2009. doi:10.1093/imanum/drn084.
- [25] V. Gradinaru, R. Hiptmair, Whitney elements on pyramids, Electron. Trans. Numer. Anal. 8 (1999) 154-168.
- [26] V. Gyrya, K. Lipnikov, High-order mimetic finite difference method for diffusion problems on polygonal meshes, J. Comput. Phys. 227 (2008) 8841-8854.
- [27] F. Hermeline, A finite volume method for approximating 3d diffusion operators on general meshes, J. Comput. Phys. 228 (16) (2009) 5763–5786.
- [28] R. Hiptmair, Discrete Hodge operators, Numer. Math. 90 (2) (2001) 265-289.
- [29] J. Hyman, M. Shashkov, Mimetic discretisations of Maxwell's equations and the equations of magnetic diffusion, Prog. Electromagnet. Res. 32 (2001) 89-121.
- [30] J.M. Hyman, M. Shashkov, Adjoint operators for the natural discretizations of the divergence gradient and curl on logically rectangular grids, Appl. Numer. Math. 25 (4) (1997) 413–442.
- [31] J.M. Hyman, M. Shashkov, Natural discretizations of the divergence gradient and curl on logically rectangular grids, Comput. Math. Appl. 33 (4) (1997) 81-104.
- [32] J.M. Hyman, M. Shashkov, The orthogonal decomposition theorems for mimetic finite difference methods, SIAM J. Numer. Anal. 36 (3) (1999) 788-818.
- [33] Yu. Kuznetsov, S. Repin, New mixed finite element method on polygonal and polyhedral meshes, Russ. J. Numer. Anal. Math. Model. 18 (3) (2003) 261– 278
- [34] L.D. Landau, E.M. Lifshitz, L.P. Pitaevskii, Electrodynamics of Continuous Media, second ed., Elsevier Butterworth-Heinemann, Burlington, MA, 2004.
- [35] S.L. Lyons, R.R. Parashkevov, X.H. Wu, A family of H¹-conforming finite element spaces for calculations on 3D grids with pinch-outs, Numer. Linear Algebra Appl. 13 (9) (2006) 789–799.
- [36] L. Margolin, M. Shashkov, P. Smolarkiewicz, A discrete operator calculus for finite difference approximations, Comput. Methods Appl. Mech. Eng. 187 (2000) 365–383.
- [37] C. Matiussi, An analysis of finite volume, finite element, and finite difference methods using some concepts of algebraic topology, J. Comput. Phys. 133 (1997) 289–309.
- [38] P. Monk, Finite Element Methods for Maxwell equations, Oxford University Press, New York, 2003.
- [39] J.-C. Nédélec, A new family of mixed finite elements in \mathbb{R}^3 , Numer. Math. 50 (1) (1986) 57–81.
- [40] R.A. Nicolaides, D.-Q. Wang, Convergence analysis of a covolume scheme for Maxwell's equations in three dimensions, Math. Comput. 67 (223) (1998) 947–963.
- [41] R.A. Nicolaides, Direct discretization of planar div-curl problems, SIAM J. Numer. Anal. 29 (1) (1992) 32–56.
- [42] R.A. Nicolaides, X. Wu, Covolume solutions of three-dimensional div-curl equations, SIAM J. Numer. Anal. 34 (6) (1997) 2195-2203.
- [43] J.B. Perot, V. Subramanian, Higher-order mimetic methods for unstructured meshes, J. Comput. Phys. 219 (1) (2006) 68-85.
- [44] M. Shashkov, S. Steinberg, Solving diffusion equations with rough coefficients in rough grids, J. Comput. Phys. 129 (2) (1996) 383–405.
- [45] I.E. Sutherland, G.W. Hodgman, Reentrant polygon clipping, Commun. ACM 17 (1) (1974) 32-42.
- [46] A. Tabarraei, N. Sukumar, Application of polygonal finite elements in linear elasticity, Int. J. Comput. Methods 3 (4) (2006) 503–520.
- [47] A. Tabarraei, N. Sukumar, Extended finite element method on polygonal and quadtree meshes, Comput. Methods Appl. Mech. Eng. 197 (5) (2007) 425– 438.
- [48] K.A. Trapp, Inner products in covolume and mimetic methods, M2AN: Math. Model. Numer. Anal. 42 (2008) 941-959.
- [49] K.S. Yee, Numerical solution of initial boundary value problems involving Maxwell's equations in isotropic media, IEEE T. Antenn. Propag. 14 (1966) 302–307.