

Immersed Virtual Element Methods for Electromagnetic Interface Problems in Three Dimensions

Shuhao Cao

*Division of Computing, Analytics, and Mathematics,
School of Science and Engineering, University of Missouri-Kansas City
Kansas City, MO 64110
scao@umkc.edu*

Long Chen

*Department of Mathematics, University of California Irvine
Irvine, CA 92697
chenlong@math.uci.edu*

Ruchi Guo

*Department of Mathematics, University of California Irvine
Irvine, CA 92697
ruchig@uci.edu*

Finite element methods for Maxwell's equations are highly sensitive to the conformity of approximation spaces, and non-conforming methods may cause loss of convergence. This fact leads to an essential obstacle for almost all the interface-unfitted mesh methods in the literature regarding the application to Maxwell interface problems, as they are based on non-conforming spaces. In this work, a novel immersed virtual element method for solving a 3D Maxwell interface problems is developed, and the motivation is to combine the conformity of virtual element spaces and robust approximation capabilities of immersed finite element spaces. The proposed method is able to achieve optimal convergence for a 3D Maxwell interface problem. To develop a systematic framework, the H^1 , $\mathbf{H}(\text{curl})$ and $\mathbf{H}(\text{div})$ interface problems and their corresponding problem-orientated immersed virtual element spaces are considered all together. In addition, the de Rham complex will be established based on which the Hiptmair-Xu (HX) preconditioner can be used to develop a fast solver for the $\mathbf{H}(\text{curl})$ interface problem.

Keywords: Maxwell's equations; interface problems; virtual element methods; immersed finite element methods; maximum angle conditions; de Rham complex; fast solvers

AMS Subject Classification: 65N12, 65N15, 65N30, 46E35

1. Introduction

In this article, we shall develop a systematic framework to construct three-dimensional (3D) H^1 , $\mathbf{H}(\text{curl})$, and $\mathbf{H}(\text{div})$ virtual element spaces involving discontinuous coefficients, referred to as the immersed virtual element (IVE) spaces, that can be used to solve the corresponding interface problems described in Section

1.1 on unfitted meshes. The proposed method is particularly important for electromagnetic interface problems as the current unfitted-mesh methods in the literature have essential difficulty in handling $\mathbf{H}(\text{curl})$ problems, see the detailed discussion in Section **1.2**.

1.1. Model problems

Let $\Omega \subseteq \mathbb{R}^3$ denote an open and bounded modeling domain, and a subdomain $\Omega^- \subseteq \Omega$ contains the medium which has the physical property distinguished from the background medium occupying the subdomain $\Omega^+ = \Omega \setminus \overline{\Omega^-}$. The surface $\Gamma = \partial\Omega^-$ is called interface and assumed to be sufficiently smooth with the normal vector \mathbf{n} pointing from Ω^+ to Ω^- . We introduce two discontinuous piecewise constant parameters representing the medium properties:

$$\alpha = \begin{cases} \alpha^- & \text{in } \Omega^-, \\ \alpha^+ & \text{in } \Omega^+, \end{cases} \quad \beta = \begin{cases} \beta^- & \text{in } \Omega^-, \\ \beta^+ & \text{in } \Omega^+, \end{cases} \quad (1.1)$$

where α^\pm and β^\pm are assumed to be positive constants.

The classic H^1 -elliptic interface problem reads as

$$-\nabla \cdot (\beta \nabla u) = f \quad \text{in } \Omega^- \cup \Omega^+, \quad (1.2)$$

with $f \in L^2(\Omega)$, subject to certain boundary conditions on $\partial\Omega$ and jump conditions

$$[u]_\Gamma := u^+ - u^- = 0, \quad (1.3a)$$

$$[\beta \nabla u \cdot \mathbf{n}]_\Gamma := \beta^+ \nabla u^+ \cdot \mathbf{n} - \beta^- \nabla u^- \cdot \mathbf{n} = 0, \quad (1.3b)$$

where the parameter β may represent, for example, the conductivity in electrical applications [61,84](#), or the dielectric constant in Poisson-Boltzmann equations [35,88](#).

For electromagnetic interface problems, we consider the following curl-curl-elliptic model

$$\text{curl}(\alpha \text{curl } \mathbf{u}) + \beta \mathbf{u} = \mathbf{f} \quad \text{in } \Omega = \Omega^- \cup \Omega^+, \quad (1.4)$$

which is derived from discretizing a time-dependent Maxwell system in which the magnetic field is eliminated. Here for simplicity we assume $\mathbf{f} \in \mathbf{H}(\text{div}; \Omega)$. If positive piecewise constant parameters ϵ , σ and μ represent the electric permeability, conductivity and magnetic permeability of the medium respectively, then $\alpha = \mu^{-1}$ and $\beta = \epsilon \Delta t^{-2} + \sigma \Delta t^{-1}$. Due to the interface, the following jump conditions are imposed for the electrical field \mathbf{u}^\pm at the interface:

$$[\mathbf{u} \times \mathbf{n}]_\Gamma := \mathbf{u}^+ \times \mathbf{n} - \mathbf{u}^- \times \mathbf{n} = 0, \quad (1.5a)$$

$$[\alpha \text{curl } \mathbf{u} \times \mathbf{n}]_\Gamma := \alpha^+ \text{curl } \mathbf{u}^+ \times \mathbf{n} - \alpha^- \text{curl } \mathbf{u}^- \times \mathbf{n} = 0, \quad (1.5b)$$

$$[\beta \mathbf{u} \cdot \mathbf{n}]_\Gamma := \beta^+ \mathbf{u}^+ \cdot \mathbf{n} - \beta^- \mathbf{u}^- \cdot \mathbf{n} = 0. \quad (1.5c)$$

In the problem above, β has a similar physical meaning to that in the H^1 interface problem, for example, to represent the electric conductivity. In addition, the condition (1.5a) comes from the tangential continuity of electrical fields. As $\alpha \operatorname{curl} \mathbf{u}$ describes the temporal change of magnetic fields, the condition (1.5b) is related to the tangential continuity of magnetic fields. In fact, such physical relations are naturally encoded in a de Rham complex, see the discussion in Section 3.

Electromagnetic interface problems are of great importance due to a large variety of science and engineering applications. Typical examples include electromagnetic motors and actuators involving metal-air or metal-metal interface ^{22,48} and electromagnetic inverse scattering ^{44,75} that use electromagnetic waves to detect objection. Solving the $\mathbf{H}(\operatorname{curl})$ interface problem with optimal convergence is a challenging goal that conventional unfitted-mesh methods fail to meet (e.g., see the discussion in Section 1.2), and trying to overcome this difficulty is the main motivation for the present research.

For only the purpose of completeness, the $\mathbf{H}(\operatorname{div})$ interface problem is given by

$$-\nabla \operatorname{div}(\mathbf{u}) + \alpha \mathbf{u} = \mathbf{f} \quad \text{in } \Omega^- \cup \Omega^+, \quad (1.6)$$

with $\mathbf{f} \in \mathbf{H}(\operatorname{curl}; \Omega)$, subject to a certain boundary condition on $\partial\Omega$ and the jump conditions

$$[\mathbf{u} \cdot \mathbf{n}]_\Gamma := \mathbf{u}^+ \cdot \mathbf{n} - \mathbf{u}^- \cdot \mathbf{n} = 0, \quad (1.7a)$$

$$[\alpha \mathbf{u} \times \mathbf{n}]_\Gamma := \alpha^+ \mathbf{u}^+ \times \mathbf{n} - \alpha^- \mathbf{u}^- \times \mathbf{n} = \mathbf{0}, \quad (1.7b)$$

$$[\operatorname{div}(\mathbf{u})]_\Gamma := \operatorname{div}(\mathbf{u}^+) - \operatorname{div}(\mathbf{u}^-) = 0. \quad (1.7c)$$

The system comes from a mixed finite element method with a gradient formulation ⁶. The related $\mathbf{H}(\operatorname{div})$ interface problem and $\mathbf{H}(\operatorname{div})$ -immersed element have been discussed in Ref. 59, 68 and thus will not be the focus of this work. The parameter α here is inherited from the $\mathbf{H}(\operatorname{curl})$ case.

1.2. Challenges of electromagnetic interface problems on unfitted meshes

For conforming finite element methods (FEMs) to perform optimally, the mesh has to fit or approximate the interface geometry “well enough”. However, an efficient high-quality 3D mesh generation itself remains a challenging problem, which is particularly expensive for complicated geometries (see e.g., Chapter 5.6 in Ref. 73). A promising solution, to alleviate the difficulty in mesh generation, is to generate a cheap background unfitted mesh, and then to further triangulate those elements cut by the interface ⁴¹. The modification is highly efficient since this extra procedure only needs to be done locally around the interface. However, this approach in general cannot yield shape-regular elements near the interface; instead the shape regularity of triangulation is relaxed to the maximum angle condition. The interpolation estimates based on the maximum angle condition have been widely studied for Lagrange elements ⁹, Raviart-Thomas elements ^{1,74}, and 3D Nédélec elements ²⁵.

This approach is very successful in the two-dimensional (2D) case ⁴⁰ as an admissible local triangulation satisfying the maximum angle condition always exists for a shape-regular background mesh, e.g., see Lemma 3.1 in Ref. 29 and Proposition 2.4 in Ref. 37. Nevertheless, in the 3D case, these locally re-meshed triangulations may not form a globally admissible mesh, as it may not necessarily satisfy the Delaunay property and/or the maximum angle condition due to the existence of slivers ⁷⁹.

To overcome the difficulty of 3D mesh generation, Ref. 37 proposed a novel method, which uses polyhedra as interface-fitted elements cut from a background Cartesian mesh, rather than to further triangulate to obtain a tetrahedral mesh. To handle the discretization on polyhedra, a virtual element method (VEM) ¹⁴ is used. In fact, $\mathbf{H}(\text{curl})$ virtual element spaces have been constructed and applied to Maxwell's equations in Ref. 17, 15, 16, 19. However, the analysis for $\mathbf{H}(\text{curl})$ problems is quite a different story. Some more recent error analysis for VEM, e.g., the ones developed in Ref. 18, 23, 27, cannot be directly used to obtain even optimal error estimates, and some more delicate techniques are needed on an ad hoc basis, e.g., Ref. 17, 30, 15. For interface problems, an extra layer of difficulty is to make error bounds robust with respect to potential anisotropic element shapes. In Ref. 27, a more rigorous analysis is given on anisotropic elements generated from Cartesian meshes cut by the interface for the 2D H^1 case.

Meanwhile, on unfitted meshes, another direction to circumvent the mesh generation issue is to modify finite element (FE) spaces such that the new spaces can capture the jump behaviors in an optimal sense. There have been extensive works in this direction including immersed finite element (IFE) methods ^{3,71,52}, CutFEM or Nitsche's penalty methods ^{21,26,77,67,72}, multiscale FEMs ⁴³ and so on, which are widely applied to various interface problems. We also refer readers to FDTD methods ⁸⁹ based on finite difference formulation for Maxwell's equations with material interfaces. For almost all the unfitted-mesh methods in the literature, the FE space modification is usually applied element-wise or piecewise relative to the interface. Thus, this practice results in discontinuities across interface elements' non-interface boundaries or at the interface. Such non-conformity can be handled by penalties on element boundaries to impose continuity such as in the IFE methods ^{51,55,71}, or on the interface itself to impose jump conditions such as the Nitsche's methods ^{21,26,77,67,72}. With suitable penalties, robust optimal convergence rates can be indeed obtained for the H^1 -type interface problems (1.2), but to the authors' best knowledge, not the considered electromagnetic interface problem (1.4).

Compared with the analysis for H^1 problems, the most drastic difference stems from the underlying Sobolev space $\mathbf{H}^s(\text{curl}; \Omega)$. In particular, for many non-conforming and discontinuous Galerkin (dG)-type methods, one needs to estimate the penalty term which, by the standard techniques (e.g., see Lemma 5.52 in Ref. 75), leads to estimates as follows

$$h^{-1/2} \|\mathbf{u} - \pi_F \mathbf{u}\|_{\mathbf{L}^2(F)} \lesssim h^{s-1} \|\mathbf{u}\|_{\mathbf{H}^s(\text{curl}; K)}, \quad (1.8)$$

where π_F is a certain projection operator on a face F of an element K . The order in

(1.8) implies that even a moderate regularity $s = 1$ yields no approximation order due to the presence of a penalty/stabilization term in the form of $h^{-1} \int_F [\mathbf{u}_h \times \mathbf{n}] \cdot [\mathbf{v}_h \times \mathbf{n}] \, ds$. For standard dG methods, the work in Ref. 63, 62 can circumvent the suboptimality, and the analysis relies on a $\mathbf{H}(\text{curl})$ -conforming subspace of the broken dG space on tetrahedral meshes. However, for unfitted-mesh methods for interface problems in the literature, this problem becomes more severe, since such a conforming subspace may not exist. Numerically, the loss of convergence has been observed and reported in a series of work ^{31,32,57} for the $\mathbf{H}(\text{curl})$ interface problem. In Section 7.2, we also present one numerical example to show that a penalty-type IFE method cannot achieve optimal convergence. So we believe that this difficulty is essential rather than caused by the limitation of analysis techniques.

The scaling factor h^{-1} in this essential issue is commonly used for stabilization in dG methods, but shows to be too “strong” for the space $\mathbf{H}^1(\text{curl}; \Omega)$. In fact, for a Lipschitz domain D , it is well-known that the trace of $H^1(D)$ is in $H^{1/2}(\partial D)$. While for $\mathbf{H}(\text{curl}; D)$, the tangential trace is merely in $\mathbf{H}^{-1/2}(\text{div}; \partial D)$, which should lead to different scaling factors for the stabilization terms on faces. Here we refer readers to Ref. 24 for the analysis of the relation between the scaling factor of a non-conforming method and function’s regularity, and more recently a weighted Sobolev space treatment ¹². In summary, the scaling factor, which is traditionally viewed to be strong enough to ensure stability for H^1 problems, leads to suboptimal convergence in non-conforming methods for $\mathbf{H}(\text{curl})$ problems. On the contrary, various conforming VEMs ^{13,17,29,19}, can use a correct scaling h in (1.8) to achieve optimal convergence, but they are not easy to adapt to 3D anisotropic meshes. More recently in Ref. 54, the virtual element method for the 3D H^1 -interface problem is analyzed under the setting of anisotropic meshes near the interface.

In conclusion, developing unfitted-mesh methods for the $\mathbf{H}(\text{curl})$ interface problem is much more challenging than its H^1 counterpart. For non-matching mesh methods, some work can obtain optimal convergence under the usual $\mathbf{H}^1(\text{curl})$ -regularity by making a certain assumption of meshes being coupled at the interface, see Ref. 64, 39. For many unfitted-mesh methods, the meshes or spaces are generally completely broken, then at least the H^2 -regularity has to be assumed to achieve optimal convergence, e.g., see Ref. 20, 72. In Ref. 57 for the 2D case and Ref. 34 for the 3D case, Petrov-Galerkin methods are developed that can achieve optimal convergence, but results in a non-symmetric scheme. A robust optimal convergence for VEM is established in Ref. 29, but it relies on a “virtual” triangulation satisfying the maximum angle condition, which may not be available in 3D. Therefore, to our best knowledge, currently there seems no satisfactory methodology for the 3D $\mathbf{H}(\text{curl})$ interface problem considered.

1.3. A Novel Method

To develop unfitted-mesh methods for the $\mathbf{H}(\text{curl})$ interface problem, based on the discussion above, it is preferable to use a conforming space. In the meantime,

this space must admit sufficient approximation capabilities robust with respect to the anisotropy of subelements. This consideration motivates us to combine the conformity of virtual element spaces and the approximation capabilities of IFE spaces. In our recent work ³⁰, we have successfully realized this idea for the 2D case, which is referred to as immersed virtual element (IVE) methods.

The fundamental idea is to impose local PDEs on interface elements to enforce both conformity and jump condition of which the solutions are used as the spaces for discretization. The IVE spaces can be understood as a special family of H^1 , $\mathbf{H}(\text{curl})$ and $\mathbf{H}(\text{div})$ virtual element spaces ^{14,15,16,17} with discontinuous coefficients. For the H^1 case, it is also exactly the space of special FEMs by Babuška et al. ^{10,11} for a simple 1D case, and it becomes the multiscale FE space ⁴³ for higher dimensional cases where the local PDEs are solved on sub-grids. The similar idea was also employed in the enriched IFE method ². The proposed IVE discretization follows the meta-framework of VEM: the local PDEs need not be solved exactly, certain projections are computed instead with sufficient approximation capability to capture the jump conditions. It can successfully yield optimal convergence rates for the $\mathbf{H}(\text{curl})$ interface problem, which has been rigorously proved in the 2D case ³⁰. In this work, we focus on the development of the IVE spaces, the scheme, and the implementation in the 3D case. We leave the theoretical part to another upcoming work as a rigorous error analysis involves much more technicalities in 3D and is not a trivial generalization of that in 2D.

Developing 3D IVE spaces is significantly more complicated than the 2D case, especially for the $\mathbf{H}(\text{curl})$ space. An immediate question is how to design appropriate div-curl systems as local problems with discontinuous coefficients that have a rigorous well-posedness. Here, special attention must also be paid to designing the local problems such that their solutions have computable projections to IFE spaces. The key is to modify the source terms for the local problems and to construct certain weighted projections with regard to the weights as Hodge star operators. The second issue is to design suitable trace spaces on element boundaries, in which the functions serve as the boundary conditions for those local problems. The trace spaces need to provide sufficient and robust approximation properties. In the 2D case, the boundary space consists of piecewise constants or linear functions on each edge, the simpleness of which attributes to the trivial geometry of the element boundaries, see Ref. 14, 27, 30. However, in the 3D case, it becomes much more obscure. For the classical virtual spaces ^{14,15,16,17}, the trace spaces are generally formulated by solutions of some extra 2D local problems defined on polygonal faces. In this work, we propose a rather different yet simpler approach: to use the standard FE spaces defined on a 2D triangulation satisfying the maximum angle condition on each element face. Such a triangulation not only benefits the robust approximation property due to the maximum angle condition, but also facilitates the code development leading to an efficient implementation with suitable data structures. In summary, on the boundary faces, we opt for an interface-fitted 2D triangulation and use local problems to extend the shape functions to the interior of

each interface element. Hence, the present research is, in fact, a combination of the three classical methodologies: VEM, IFE and FEM, towards solving the challenging electromagnetic interface problem efficiently.

Although our focus is on electromagnetic interface problems, we shall develop a systematic framework for all the H^1 , $\mathbf{H}(\text{curl})$ and $\mathbf{H}(\text{div})$ interface problems contributing to a solid mathematical foundation. They are connected by the following de Rham complex, and are shown to have the usual nodal, edge, and face degrees of freedom (DoFs), through which both the exact sequence and the commutative property can be established.

$$\begin{array}{ccccccccccc}
 \mathbb{R} & \xhookrightarrow{\quad} & H^2(\beta; \mathcal{T}_h) & \xrightarrow{\text{grad}} & \mathbf{H}^1(\text{curl}, \alpha, \beta; \mathcal{T}_h) & \xrightarrow{\text{curl}} & \mathbf{H}^1(\text{div}, \alpha; \mathcal{T}_h) & \xrightarrow{\text{div}} & H^1(\mathcal{T}_h) & \longrightarrow 0 \\
 & & \downarrow I_h^n & & \downarrow I_h^e & & \downarrow I_h^f & & \downarrow \Pi_h^0 & \\
 \mathbb{R} & \xhookrightarrow{\quad} & V_h^n & \xrightarrow{\text{grad}} & \mathbf{V}_h^e & \xrightarrow{\text{curl}} & \mathbf{V}_h^f & \xrightarrow{\text{div}} & Q_h & \longrightarrow 0
 \end{array} \tag{1.9}$$

Another major challenge for 3D interface problems is an appropriate fast solver. Multigrid methods are widely used, and we refer readers to Ref. 86 for FEM and Ref. 37 for VEM, both of which study the H^1 -interface problem. For $H(\text{curl})$ equations, the fast solvers are even more challenging ^{87,60} due to the non-trivial kernel space of the curl operator, which is another motivation to lay out the de Rham complex (1.9) for the proposed VEM spaces. In this work, we generalize multigrid-based Hiptmair-Xu (HX) preconditioner ^{60,38} for regular $\mathbf{H}(\text{curl})$ problem to the interface case. Moreover, for fitted mesh methods, the condition numbers may still suffer from the possible anisotropic element shapes, even though the error bounds are robust. In Ref. 86, the DoFs near the interface and in the background mesh are split to form “fine-coarse” block matrices, thus an optimal two-level solver is developed. In this paper, a block diagonal smoother is proposed to handle the anisotropic element shape near the interface, similar to the practice in Ref. 86. To our best knowledge, this is the first research towards applying the HX preconditioner to VEM, and also the first fast solver for unfitted-mesh methods for solving electromagnetic interface problems. Numerical results demonstrate that the solver is robust with respect to both the mesh size and the shape of small-cutting elements.

This article has additional 6 sections. In the next section, we introduce the meshes, focusing especially on the element boundary triangulation. In Section 3, we describe the desired Sobolev spaces encoding jump conditions that are to be approximated. In Sections 4 and 5, we introduce IFE spaces and IVE spaces, respectively. In Section 6, we describe the computation scheme, fast solvers, and implementation aspects. In the last section, we present a group of numerical experiments.

2. Meshes

In this article, we focus on a given interface-independent and shape-regular tetrahedral mesh of Ω , but the proposed method can be also adapted to any Cartesian

cubic meshes. This tetrahedral mesh is referred to as the background mesh, and is denoted by \mathcal{T}_h . If an element in \mathcal{T}_h intersects the interface, then it is called an interface element, or a non-interface element otherwise. The collection of interface elements is denoted as \mathcal{T}_h^i . For each $K \in \mathcal{T}_h^i$, we denote $\Gamma^K = \Gamma \cap K$. Let \mathcal{T}_h^n be the set of non-interface elements. All elements are considered as open sets.

One of the critical ingredients to formulate the H^1 , $\mathbf{H}(\text{curl})$ and $\mathbf{H}(\text{div})$ local interface problems is to impose appropriate boundary conditions on element boundary. Different from the prevailing approach in the literature, the authors in Ref. 37 proposes a novel approach by using exclusively square and triangular faces in interface polyhedra. In Ref. 37, a Delaunay triangulation routine is called for the nodes in the background mesh, interface nodes, and some added vertices near the interface, then the triangular faces are extracted for their corresponding polyhedra. In this work, similar to the practice in Ref. 37, standard FE functions on a 2D triangulation of any given element face are used as the boundary conditions. Hence, we make a fundamental assumption called *interface fitted boundary triangulation*:

(A) For each interface element K , each of its face admits a triangulation satisfying the maximum angle condition. The triangles are formed by only the vertices of K and/or the cutting points of the original interface, i.e., there are no newly-added interior vertices to form the edges. If a face is cut by the interface, then this triangulation must be fitted to the interface in the sense that the curve (the intersection of the face and interface) is approximated by an edge of this triangulation with error in the order of $\mathcal{O}(h_K^2)$.

We illustrate the Assumption A in Figure 2.1: each face of an interface element is partitioned into multiple triangles by the newly added edges including the one connecting the cutting points (red points in the figure). It can be understood that a local 2D fitted mesh satisfying the maximum angle condition is generated around the interface but only on faces. Although, as aforementioned in the introduction, generating a 3D interface-fitted mesh may be difficult or even impossible in certain situations, it is much easier to generate a 2D interface-fitted mesh. In particular, since the considered original background tetrahedral meshes only have triangular faces, the boundary triangulation with the maximum angle condition is always guaranteed by Lemma 3.1 in Ref. 29. See, for example, the left plot in Figure 2.1 if the element is cut by the interface only once. If cubic meshes are used, then Proposition 2.4 in Ref. 37 guarantees an admissible *boundary triangulation*. But unlike Ref. 37 where the element is divided into two polyhedrons, here the cut tetrahedron is still treated as one element with more than 4 triangular faces.

We highlight that the *interface fitted boundary triangulation* is able to link the fitted 2D and unfitted 3D meshes, and also bridges the standard 2D FE spaces and 3D virtual element spaces. Our previous work in the 2D case³⁰ suggests that it is also one of the keys to overcome suboptimal convergence caused by non-conforming spaces for Maxwell's equations, as well as help in anisotropic analysis for the virtual spaces. In addition, the proposed *boundary triangulation*, in fact, greatly benefits the

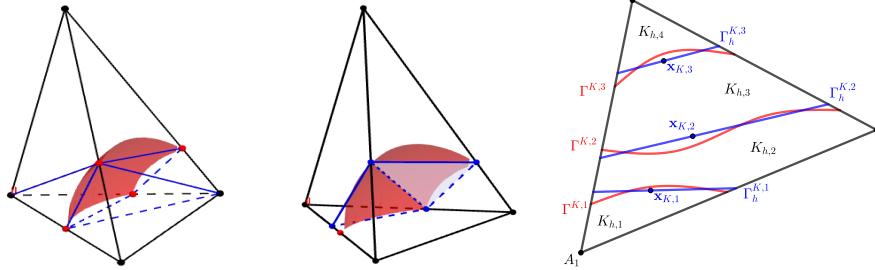


Fig. 2.1: Left: illustration of boundary triangulation of Assumption A. Middle: an approximate plane Γ_h^K to Γ^K . Right: 2D illustration of an element cut by the interface with multiple times.

computation. One of the difficult aspects of implementing polytopal finite element approximation is the ever-changing number of DoFs in an element. In our approach, since only triangular faces are present in every interface or non-interface element, the assembling can be uniformly handled fixed-width matrices in the face-oriented data structure, please refer to Section 6.4 for details, see also Ref. 37, 13 for a face-based approach.

Another advantage of the proposed method is the flexibility to handle complex interface element geometry. Specifically, on elements that are cut by interface with multiple times, e.g., see the right plot in Figure 2.1, the proposed IVE spaces can be easily constructed as long as an admissible *boundary triangulation* can be constructed.

3. Some Sobolev Spaces and Well-posedness

In this section, we describe a group of modified Sobolev spaces that incorporate the interface conditions. Let us first recall some standard spaces. Given an open subdomain $D \subseteq \Omega$, for $s \geq 0$, we let $H^s(D)$ be the standard scalar Sobolev space and $\mathbf{H}^s(D) := (H^s(D))^3$. Now introduce

$$\mathbf{H}^s(\text{curl}; D) = \{\mathbf{u} \in \mathbf{H}^s(D) : \text{curl } \mathbf{u} \in \mathbf{H}^s(D)\}, \quad (3.1a)$$

$$\mathbf{H}^s(\text{div}; D) = \{\mathbf{u} \in \mathbf{H}^s(D) : \text{div } \mathbf{u} \in H^s(D)\}. \quad (3.1b)$$

If $D \cap \Gamma \neq \emptyset$, we let $D^\pm = \Omega^\pm \cap D$, and further let $H^s(\cup D^\pm)$, $\mathbf{H}^s(\text{curl}; \cup D^\pm)$ and $\mathbf{H}^s(\text{div}; \cup D^\pm)$ consist of functions that belong to the corresponding spaces on each D^\pm but without any conditions on ∂D^\pm .

Now, with the mesh \mathcal{T}_h we are ready to define the interface-encoded Sobolev spaces:

$$H^2(\beta; \mathcal{T}_h) = H^1(\Omega) \cap \{u \in H^2(\cup K^\pm) : \beta \nabla u \in \mathbf{H}(\text{div}; K), \forall K \in \mathcal{T}_h\}, \quad (3.2a)$$

$$\mathbf{H}^1(\text{curl}, \alpha, \beta; \mathcal{T}_h) = \mathbf{H}(\text{curl}; \Omega) \cap \{\mathbf{u} \in \mathbf{H}^1(\text{curl}; \cup K^\pm) : \beta \mathbf{u} \in \mathbf{H}(\text{div}; K),$$

$$\alpha \operatorname{curl} \mathbf{u} \in \mathbf{H}(\operatorname{curl}; K), \forall K \in \mathcal{T}_h \}, \quad (3.2b)$$

$$\begin{aligned} \mathbf{H}^1(\operatorname{div}, \alpha; \mathcal{T}_h) = \mathbf{H}(\operatorname{div}; \Omega) \cap \{ \mathbf{u} \in \mathbf{H}^1(\operatorname{div}; \cup K^\pm) : \alpha \mathbf{u} \in \mathbf{H}(\operatorname{curl}; K), \\ \operatorname{div} \mathbf{u} \in H^1(K), \forall K \in \mathcal{T}_h \}, \end{aligned} \quad (3.2c)$$

$$H^1(\mathcal{T}_h) = L^2(\Omega) \cap \{ u \in H^1(K), \forall K \in \mathcal{T}_h \}. \quad (3.2d)$$

Note that, on a non-interface element K , the conditions are trivial since they are just consequences of $H^2(K)$, $\mathbf{H}^1(\operatorname{curl}; K)$ and $\mathbf{H}^1(\operatorname{div}; K)$. On an interface element K those conditions exactly encode both the conformity and interface information. To see the relation more clearly, we let $H^2(\beta; K)$, $\mathbf{H}^1(\operatorname{curl}, \alpha, \beta; K)$, $\mathbf{H}^1(\operatorname{div}, \alpha; K)$ and $H^1(K)$ be the local spaces on K of their global counterparts in (3.2). The spaces above are just constructed so that the following diagram is well-defined:

$$\begin{array}{ccccccccccc} \mathbb{R} & \xrightarrow{\hookrightarrow} & H^2(\beta; K) & \xrightarrow{\nabla} & \mathbf{H}^1(\operatorname{curl}, \alpha, \beta; K) & \xrightarrow{\operatorname{curl}} & \mathbf{H}^1(\operatorname{div}, \alpha; K) & \xrightarrow{\operatorname{div}} & H^1(K) & \longrightarrow & 0 \\ & & \downarrow I & & \downarrow \beta & & \downarrow \alpha & & \downarrow I & & \\ 0 & \longleftarrow & L^2(K) & \xleftarrow{\operatorname{div}} & \mathbf{H}(\operatorname{div}; K) & \xleftarrow{\operatorname{curl}} & \mathbf{H}(\operatorname{curl}; K) & \xleftarrow{\nabla} & H^1(K) & \xleftarrow{\hookleftarrow} & \mathbb{R} \end{array} \quad (3.3)$$

In Diagram (3.3), α and β could be understood as Hodge star operators mapping k -forms to $(3 - k)$ -forms for $k = 2, 1$, respectively. Take $\beta : \mathbf{H}^1(\operatorname{curl}, \alpha, \beta; K) \rightarrow \mathbf{H}(\operatorname{div}; K)$ as an example. A function \mathbf{u} in $\mathbf{H}^1(\operatorname{curl}, \alpha, \beta; K)$ can be thought of as a vector proxy of a 1-form. Then $\beta \mathbf{u} \in \mathbf{H}(\operatorname{div}; K)$ is a 2-form. The jump conditions (1.5) on the interface are from the continuity of the mapped forms. Construction of the desired virtual spaces is to mimic this diagram in the discretized level.

In the following discussion, given any face F in the mesh, we shall denote the tangential component of \mathbf{u} by $\mathbf{u}^\tau|_F$ for admissible \mathbf{u} defined in the bulk ω such that $F \subseteq \partial\omega$, and we will drop $|_F$ if there is no danger of causing confusion. In addition, we will also frequently use the 2D rotation operator denoted by rot_F on F . Let ∇_F denote the surface gradient. Then, for a function φ defined on F , rot_F is defined in the distributional sense such that

$$\langle \operatorname{rot}_F \varphi, v \rangle_F := (\varphi, \nabla_F v \times \mathbf{n})_F, \quad \forall v \in H_0^1(F). \quad (3.4)$$

For each subdomain $\omega \subseteq \Omega$, and φ defined on $\partial\omega$, $\operatorname{rot}_{\partial\omega} \varphi$ can be defined similarly,

$$\langle \operatorname{rot}_{\partial\omega} \varphi, v \rangle_{\partial\omega} := (\varphi, \nabla_F v \times \mathbf{n})_{\partial\omega}, \quad \forall v \in H^1(\omega), \quad (3.5)$$

while it can be verified that $\operatorname{rot}_{\partial\omega} \varphi|_F = \operatorname{rot}_F \varphi$ when $\operatorname{rot}_{\partial\omega} \varphi \in L^2(\omega)$. Here $\langle \cdot, \cdot \rangle_{\partial\omega}$ denotes the usual pairing between $H^{-1/2}(\partial\omega)$ - $H^{1/2}(\partial\omega)$, and $\langle \cdot, \cdot \rangle_F$ is defined similarly for $F \subset \partial\omega$. In particular, the well-known formula states

$$\operatorname{curl} \mathbf{u} \cdot \mathbf{n}_F = \operatorname{rot}_F \mathbf{u}, \quad \text{for } \mathbf{u} \in \mathbf{H}(\operatorname{curl}; \omega), \quad \text{on } F \subset \partial\omega, \quad (3.6)$$

where \mathbf{n}_F is the exterior unit normal vector of F with respect to ω .

Note that the proposed global problems as well as the local problems all involve discontinuous coefficients. In order to pursue a rigorous definition of the IVE spaces, we discuss the well-posedness of some div-curl systems with discontinuous

coefficients. The systems with constant coefficients are discussed in Ref. 5, and the results with general coefficients can be found in Ref. 81, 82, 49. But here we present more detailed analysis to show that the constants in the *a priori* estimates are independent of interface location.

Lemma 3.1. *Let ω be a Lipschitz polyhedral domain which is simply-connected, let an interface γ separate ω into ω^\pm , and define a piecewise constant function $b = b^\pm > 0$ in ω^\pm . For the data functions $\mathbf{f} \in \mathbf{H}(\text{div}; \omega) \cap \ker(\text{div})$, $h \in L^2(\omega)$ and $g \in H^{-1/2}(\partial\omega)$ such that the compatibility condition holds:*

$$\int_\omega h \, dx = \langle g, 1 \rangle_{\partial\omega}, \quad (3.7)$$

then the following problem admits a unique solution $\boldsymbol{\varphi} \in \mathbf{H}(\text{div}; \omega)$ and $b\boldsymbol{\varphi} \in \mathbf{H}(\text{curl}; \omega)$

$$\text{curl}(b\boldsymbol{\varphi}) = \mathbf{f}, \quad \text{div}(\boldsymbol{\varphi}) = h \quad \text{in } \omega, \quad \boldsymbol{\varphi} \cdot \mathbf{n} = g \quad \text{on } \partial\omega. \quad (3.8)$$

If additionally $g \in L^2(\partial\omega)$, the stability result holds:

$$b_{\min}(1 + b_{\max})^{-1} \|\boldsymbol{\varphi}\|_{L^2(\omega)} \leq C_\omega (\|\mathbf{f}\|_{L^2(\omega)} + \|h\|_{L^2(\omega)} + \|g\|_{L^2(\partial\omega)}), \quad (3.9)$$

where $b_{\min} = \min\{b^-, b^+\}$, $b_{\max} = \max\{b^-, b^+\}$ and the constant C_ω only depends on the geometry of ω . If ω is star-convex with respect to a ball of the radius ρ_ω , then $C_\omega = C(h_\omega/\rho_\omega)$ where h_ω is the diameter of ω . Furthermore, if $g \in H^{1/2}(\partial\omega)$, ω is convex, and γ is a closed surface that is sufficient smooth and does not intersect the boundary, then $\boldsymbol{\varphi} \in \mathbf{H}^1(\cup\omega^\pm)$.

Proof. Since ω is assumed to be simply-connected, by Theorem 1.1 in Ref. 82, we know the solution $\boldsymbol{\varphi}$ to (3.8) admits the following decomposition

$$\boldsymbol{\varphi} = b^{-1} \nabla v + \text{curl } \mathbf{w} \quad (3.10)$$

where v is the solution to the equation

$$\text{div}(b^{-1} \nabla v) = h \quad \text{in } \omega, \quad b^{-1} \nabla v \cdot \mathbf{n} = g \quad \text{on } \partial\omega, \quad \int_\omega v \, dx = 0, \quad (3.11)$$

and \mathbf{w} is the solution to the equation

$$\text{curl}(b \text{curl } \mathbf{w}) = \mathbf{f} \quad \text{in } \omega, \quad \text{div}(\mathbf{w}) = 0 \quad \text{in } \omega, \quad \mathbf{w} \times \mathbf{n} = \mathbf{0} \quad \text{on } \partial\omega. \quad (3.12)$$

Here (3.11) is a standard well-posed elliptic interface problem and the well-posedness of (3.12) can be found in Ref. 81.

To show (3.9), based on (3.10), we show the *a-priori* estimates for both v and \mathbf{w} in terms of data. Testing (3.11) by v and using integration by parts we have

$$\int_\omega b^{-1} \nabla v \cdot \nabla v \, dx = - \int_\omega h v \, dx + \int_{\partial\omega} g v \, ds. \quad (3.13)$$

It implies, with Poincaré inequality and the trace inequality for v , that

$$\begin{aligned} \|b^{-1/2}\nabla v\|_{L^2(\omega)}^2 &\leq \|h\|_{L^2(\omega)}\|v\|_{L^2(\omega)} + \|g\|_{L^2(\partial\omega)}\|v\|_{L^2(\partial\omega)} \\ &\lesssim (\|h\|_{L^2(\omega)} + \|g\|_{L^2(\partial\omega)})\|\nabla v\|_{L^2(\omega)}. \end{aligned} \quad (3.14)$$

Cancelling one $\|\nabla v\|_{L^2(\omega)}$ in (3.14) yields the estimate: $b_{\max}^{-1}\|\nabla v\|_{L^2(\omega)} \lesssim \|h\|_{L^2(\omega)} + \|g\|_{L^2(\partial\omega)}$. As for \mathbf{w} , we note that $\mathbf{w} \in \mathbf{H}(\text{curl}) \cap \mathbf{H}(\text{div})$ with $\text{div}(\mathbf{w}) = 0$, and thus we can apply Corollary 3.51 in Ref. 75 to obtain

$$\|\mathbf{w}\|_{L^2(\omega)} \lesssim \|\text{curl } \mathbf{w}\|_{L^2(\omega)} + \|\mathbf{w} \times \mathbf{n}\|_{L^2(\partial\omega)} = \|\text{curl } \mathbf{w}\|_{L^2(\omega)}. \quad (3.15)$$

Then, testing (3.12) with \mathbf{w} , applying the integration by parts and using (3.15), we have

$$b_{\min}\|\text{curl } \mathbf{w}\|_{L^2(\omega)} \lesssim \|\mathbf{f}\|_{L^2(\omega)}. \quad (3.16)$$

Combining the estimates above, we have (3.9), and the dependence of the generic constants follows from the constants in the trace and Poincaré inequalities used above, see Ref. 23.

For $g \in H^{1/2}(\partial\omega)$, convex ω , and closed smooth γ not intersecting the boundary, there certainly holds that $v \in H^2(\cup\omega^\pm)$ and thus $\nabla v \in \mathbf{H}^1(\cup\omega^\pm)$ 65,66. As for \mathbf{w} , following the argument of Theorem 5.2 in Ref. 66, we construct $\tilde{\mathbf{w}} \in \mathbf{H}^1(\omega)$ such that $\text{curl}(\tilde{\mathbf{w}}) = \mathbf{f}$, $\text{div}(\tilde{\mathbf{w}}) = 0$ and $\tilde{\mathbf{w}} \cdot \mathbf{n} = 0$ due to $\text{div}(\mathbf{f}) = 0$, see Theorem 3.8 in Ref. 50. Then, we have $\text{curl}(b \text{curl } \mathbf{w} - \tilde{\mathbf{w}}) = \mathbf{0}$ in ω and $(b \text{curl } \mathbf{w} - \tilde{\mathbf{w}}) \cdot \mathbf{n} = 0$ and $\partial\omega$, and conclude by exact sequence that $b \text{curl } \mathbf{w} - \tilde{\mathbf{w}} = \nabla\phi$ for some $\phi \in H^1(\omega)$ such that

$$\begin{cases} \text{div}(b^{-1}\nabla\phi) = -\text{div}(b^{-1}\tilde{\mathbf{w}}) & \text{in } \omega^\pm, \\ [\phi]_\gamma = 0 & \text{on } \gamma, \\ [b^{-1}\nabla\phi \cdot \mathbf{n}]_\gamma = -[b^{-1}\tilde{\mathbf{w}} \cdot \mathbf{n}]_\gamma & \text{on } \gamma, \\ \nabla\phi \cdot \mathbf{n} = 0, & \text{on } \partial\omega. \end{cases}$$

Note that this is an interface problem with the non-homogeneous flux jump condition. As $\text{div}(b^{-1}\tilde{\mathbf{w}}|_{\omega^\pm}) \in L^2(\omega^\pm)$ and $\tilde{\mathbf{w}} \cdot \mathbf{n}|_\gamma \in H^{1/2}(\gamma)$, further by the assumption that γ does not intersect $\partial\omega$, we have $\phi \in H^2(\cup\omega^\pm)$ and thus $\nabla\phi \in \mathbf{H}^1(\cup\omega^\pm)$ by Ref. 42. Therefore, we conclude $\text{curl } \mathbf{w} \in \mathbf{H}^1(\cup\omega^\pm)$, and $\varphi \in \mathbf{H}^1(\cup\omega^\pm)$ by the decomposition (3.10). \square

Lemma 3.2. *Given a simple-connected domain ω with Lipschitz boundary, let an interface γ separates ω into ω^\pm and define a piecewise constant function $a = a^\pm > 0$ in ω^\pm . For the data functions $\mathbf{f} \in \mathbf{H}(\text{div}; \omega) \cap \ker(\text{div})$, $h \in L^2(\omega)$ and $\mathbf{g} \in \mathbf{H}^{-1/2}(\partial\omega)$ such that the compatibility condition holds:*

$$\langle \mathbf{f} \cdot \mathbf{n}, v \rangle_{\partial\omega} = \langle \text{rot}_{\partial\omega}(\mathbf{n} \times \mathbf{g}), v \rangle_{\partial\omega} \quad \forall v \in H^1(\omega), \quad (3.17)$$

then the following problem admits a solution $\varphi \in \mathbf{H}(\text{curl}; \omega)$ and $a\varphi \in \mathbf{H}(\text{div}; \omega)$

$$\text{curl } (\varphi) = \mathbf{f}, \quad \text{div}(a\varphi) = h \quad \text{in } \omega, \quad \varphi \times \mathbf{n} = \mathbf{g}, \quad \text{on } \partial\omega. \quad (3.18)$$

Furthermore, if $\mathbf{g} \in \mathbf{L}^2(\partial\omega)$, the following a-priori estimate holds:

$$a_{\min}(1 + a_{\max})^{-1} \|\varphi\|_{L^2(\omega)} \lesssim \|\mathbf{f}\|_{L^2(\omega)} + \|h\|_{L^2(\omega)} + \|\mathbf{g}\|_{L^2(\partial\omega)}, \quad (3.19)$$

where $a_{\min} = \min\{a^-, a^+\}$, $a_{\max} = \max\{a^-, a^+\}$ and the constant C_ω only depends the geometry of ω . If ω is star-convex, then $C_\omega = C(h_\omega/\rho_\omega)$.

Proof. As ω is assumed to be simply-connected, by Theorem 1.2 in Ref. 82, the solution φ to (3.18) has the following decomposition:

$$\varphi = \nabla v + a^{-1} \operatorname{curl}(\mathbf{w}), \quad (3.20)$$

where v is the solution of

$$\operatorname{div}(a \nabla v) = h \quad \text{in } \omega, \quad v = 0 \quad \text{on } \partial\omega, \quad (3.21)$$

and \mathbf{w} is the solution of

$$\begin{aligned} \operatorname{curl}(a^{-1} \operatorname{curl} \mathbf{w}) &= \mathbf{f}, & \operatorname{div}(\mathbf{w}) &= 0 & \text{in } \omega, \\ a^{-1} \operatorname{curl}(\mathbf{w}) \times \mathbf{n} &= \mathbf{g}, & \mathbf{w} \cdot \mathbf{n} &= 0, & \text{on } \partial\omega. \end{aligned} \quad (3.22)$$

By the similar argument to Lemma 3.1 with the last remark in Ref. 45, we have

$$\begin{aligned} a_{\min} \|\nabla v\|_{L^2(\omega)} &\lesssim \|h\|_{L^2(\omega)} \\ a_{\max}^{-1} \|\operatorname{curl}(\mathbf{w})\|_{L^2(\omega)} &\lesssim \|\mathbf{f}\|_{L^2(\omega)} + \|\mathbf{g}\|_{L^2(\partial\omega)}, \end{aligned} \quad (3.23)$$

which leads to the desired estimate by (3.20). \square

Remark 3.1. The key of the a priori estimates of (3.9) and (3.19) is the independence with respect to the interface location. The result of Corollary 3.51 in Ref. 75 employs a compactness argument for curl-div systems on homogeneous media. One may indeed use this technique to obtain the similar estimates for interface problems, which, however, may lead to constants depending on the interface location.

Lemma 3.3. *Given a simple-connected domain ω with Lipschitz boundary, let an interface γ separates ω into ω^\pm and define two piecewise constant functions $a = a^\pm > 0$ and $b = b^\pm > 0$ in ω^\pm . For the data functions $\mathbf{f} \in \mathbf{H}(\operatorname{div}; \omega) \cap \ker(\operatorname{div})$ and $\mathbf{g} \in \mathbf{H}(\operatorname{rot}; \partial\omega)$ such that*

$$\int_{\partial\omega} \operatorname{rot}_{\partial\omega} \mathbf{g} \, ds = 0. \quad (3.24)$$

Then, the equation

$$\operatorname{curl}(a \operatorname{curl} \varphi) = \mathbf{f}, \quad \operatorname{div}(b \varphi) = 0, \quad \varphi^\tau = \mathbf{g}, \quad (3.25)$$

admits a unique solution φ satisfying $\varphi \in \mathbf{H}(\operatorname{curl}; \omega)$, $b\varphi \in \mathbf{H}(\operatorname{div}; \omega)$, $a \operatorname{curl} \varphi \in \mathbf{H}(\operatorname{curl}; \omega)$.

Proof. We first consider a potential ψ such that

$$\operatorname{curl}(a\psi) = \mathbf{f}, \quad \operatorname{div}(\psi) = 0, \quad \text{in } \omega, \quad \psi \cdot \mathbf{n} = \operatorname{rot}_{\partial\omega} \mathbf{g}, \quad \text{on } \partial\omega. \quad (3.26)$$

The condition (3.24) together with Lemma 3.1 shows the unique existence of ψ . Then, it is easy to see that (3.25) can be equivalently written as

$$\operatorname{curl}(\varphi) = \psi, \quad \operatorname{div}(b\varphi) = 0 \quad \text{in } \omega, \quad \varphi^\tau = \mathbf{g} \quad \text{on } \partial\omega. \quad (3.27)$$

Note that the boundary condition in (3.27) is equivalent to $\varphi \times \mathbf{n} = \mathbf{g} \times \mathbf{n}$. As $\mathbf{n} \times (\mathbf{g} \times \mathbf{n}) = \mathbf{g}$, with integration by parts, we have for any $v \in H^1(\omega)$,

$$\int_{\partial\omega} (\mathbf{n} \times (\mathbf{g} \times \mathbf{n}))(\nabla v \times \mathbf{n}) \, ds = \int_{\partial\omega} \mathbf{g}(\nabla v \times \mathbf{n}) \, ds = \int_{\partial\omega} (\operatorname{rot}_{\partial\omega} \mathbf{g})v \, ds. \quad (3.28)$$

Then, the boundary condition in (3.26) shows that the compatibility condition in (3.17) indeed holds. Thus, the well-posedness follows from Lemma 3.2. \square

At last, we present the complex formed by the new globally-defined spaces of (3.2).

Lemma 3.4. *The following sequence is a complex:*

$$\mathbb{R} \xrightarrow{\quad} H^2(\beta; \mathcal{T}_h) \xrightarrow{\operatorname{grad}} \mathbf{H}^1(\operatorname{curl}, \alpha, \beta; \mathcal{T}_h) \xrightarrow{\operatorname{curl}} \mathbf{H}^1(\operatorname{div}, \alpha; \mathcal{T}_h) \xrightarrow{\operatorname{div}} H^1(\mathcal{T}_h) \longrightarrow 0. \quad (3.29)$$

When Ω is a convex polyhedron, and the interface is also sufficiently smooth not intersecting $\partial\Omega$, it is also exact.

Proof.

We first verify that $\nabla H^2(\beta; \mathcal{T}_h) \subseteq \operatorname{ker}(\operatorname{curl}) \cap \mathbf{H}^1(\operatorname{curl}, \alpha, \beta; \mathcal{T}_h)$. This is true due to the jump conditions associated with $\nabla H^2(\beta; \mathcal{T}_h)$, and $\operatorname{curl} \nabla H^2(\beta; \mathcal{T}_h) = 0$. Similarly, $\operatorname{curl} \mathbf{H}^1(\operatorname{curl}, \alpha, \beta; \mathcal{T}_h) \subseteq \operatorname{ker}(\operatorname{div}) \cap \mathbf{H}(\operatorname{div}, \alpha; \mathcal{T}_h)$ due to the jump condition associated with $\mathbf{H}^1(\operatorname{curl}, \alpha, \beta; \mathcal{T}_h)$, and the fact $\operatorname{div} \operatorname{curl} = 0$. Finally, it is trivial that $\operatorname{div} \mathbf{H}^1(\operatorname{div}, \alpha; \mathcal{T}_h) \subseteq H^1(\mathcal{T}_h)$. These results together finish the proof.

We then verify the exactness. We first show $\nabla H^2(\beta; \mathcal{T}_h) = \operatorname{ker}(\operatorname{curl}) \cap \mathbf{H}^1(\operatorname{curl}, \alpha, \beta; \mathcal{T}_h)$. Given each $\mathbf{u} \in \operatorname{ker}(\operatorname{curl}) \cap \mathbf{H}^1(\operatorname{curl}, \alpha, \beta; \mathcal{T}_h)$, by the classic exact sequence, there exists $u \in H^1(\Omega)$ such that $\nabla u = \mathbf{u}$, and by the jump conditions associated with $\mathbf{H}^1(\operatorname{curl}, \alpha, \beta; \mathcal{T}_h)$ we have u also satisfies those of $H^2(\beta; \mathcal{T}_h)$. In addition, on each element K , $\nabla u = \mathbf{u} \in \mathbf{H}^1(\cup K^\pm)$ implies $u \in H^2(\cup K^\pm)$. Therefore, $u \in H^2(\beta; \mathcal{T}_h)$.

We then verify $\operatorname{curl} \mathbf{H}^1(\operatorname{curl}, \alpha, \beta; \mathcal{T}_h) = \operatorname{ker}(\operatorname{div}) \cap \mathbf{H}^1(\operatorname{div}, \alpha; \mathcal{T}_h)$. Given a function $\mathbf{u} \in \operatorname{ker}(\operatorname{div}) \cap \mathbf{H}^1(\operatorname{div}, \alpha; \mathcal{T}_h)$, we consider a function $\varphi \in \mathbf{H}(\operatorname{div}; \Omega)$ satisfying

$$\operatorname{curl}(\beta^{-1}\varphi) = \mathbf{u}, \quad \operatorname{div}(\varphi) = 0 \quad \text{in } \Omega, \quad \varphi \cdot \mathbf{n} = 0 \quad \text{on } \partial\Omega. \quad (3.30)$$

By Lemma 3.1 with $\omega = \Omega$, $\gamma = \Gamma$ and $b = \beta^{-1}$, we have this system being well-defined with $\beta^{-1}\varphi \in \mathbf{H}(\operatorname{curl}; \Omega)$ and $\varphi \in \mathbf{H}_0(\operatorname{div}; \Omega)$. Using Lemma 3.1 again, by the geometric condition of Ω and Γ , we also have $\varphi \in \mathbf{H}^1(\cup \Omega^\pm)$. Thus, we obtain $\mathbf{v} :=$

$\beta^{-1}\varphi \in \mathbf{H}(\text{curl}; \Omega) \cap \mathbf{H}^1(\cup\Omega^\pm)$ and $\beta\mathbf{v} \in \mathbf{H}(\text{div}; \Omega)$. Furthermore, on each element K , $\mathbf{u} \in \mathbf{H}^1(\text{div}; K^\pm)$ implies $\mathbf{v} \in \mathbf{H}^1(\text{curl}; K^\pm)$. In addition, $[\alpha \text{curl } \mathbf{v} \times \mathbf{n}]_{\Gamma^K} = 0$ is trivial by the property of \mathbf{u} .

We next show $\text{div } \mathbf{H}^1(\text{div}, \alpha; \mathcal{T}_h) = H^1(\mathcal{T}_h)$. Given each $f \in H^1(\mathcal{T}_h)$. We consider a ϕ satisfying

$$\left\{ \begin{array}{l} \text{div}(\alpha^{-1}\nabla\phi) = f, \text{ in } \Omega, \\ [\phi]_\Gamma = 0, \quad [\alpha^{-1}\nabla\phi \cdot \mathbf{n}]_\Gamma = 0, \text{ on } \Gamma \end{array} \right. \quad (3.31a)$$

$$\left\{ \begin{array}{l} \phi = 0 \text{ on } \partial\Omega. \end{array} \right. \quad (3.31b)$$

$$\left\{ \begin{array}{l} \phi = 0 \text{ on } \partial\Omega. \end{array} \right. \quad (3.31c)$$

By the elliptic regularity ⁸, we have $\phi \in H^2(\cup\Omega^\pm)$, and let $\mathbf{w} = \alpha^{-1}\nabla\phi \in \mathbf{H}^1(\cup\Omega^\pm)$. On each element K , as $f \in H^1(\cup K^\pm)$, we have $\mathbf{w} \in \mathbf{H}^1(\text{div}; K^\pm)$. At last, (3.31b) leads to $[\alpha\mathbf{w} \times \mathbf{n}]_{\Gamma^K} = \mathbf{0}$ and $[\mathbf{w} \cdot \mathbf{n}]_{\Gamma^K} = 0$. \square

Remark 3.2. The classic de Rham complex with higher smoothness is given by Ref. 50, 83:

$$\mathbb{R} \xrightarrow{\hookrightarrow} H^2(\Omega) \xrightarrow{\text{grad}} \mathbf{H}^1(\text{curl}; \Omega) \xrightarrow{\text{curl}} \mathbf{H}^1(\Omega) \xrightarrow{\text{div}} L^2(\Omega) \longrightarrow 0. \quad (3.32)$$

We note that this sequence can be simply revised to be

$$\mathbb{R} \xrightarrow{\hookrightarrow} H^2(\Omega) \xrightarrow{\text{grad}} \mathbf{H}^1(\text{curl}; \Omega) \xrightarrow{\text{curl}} \mathbf{H}^1(\text{div}; \Omega) \xrightarrow{\text{div}} H^1(\Omega) \longrightarrow 0. \quad (3.33)$$

The revision can be understood immediately from $\ker(\text{div}) \cap \mathbf{H}^1(\Omega) = \ker(\text{div}) \cap \mathbf{H}^1(\text{div}; \Omega)$. The proposed new sequence (3.29) is a further generalization of (3.33) in which the jump information is incorporated. Finite element counterparts of (3.32) and (3.33) can be found in Ref. 36. Virtual element discretization of (3.29) will be discussed in Section 5.5.

4. Immersed Finite Element Spaces

In this section, we present H^1 , $\mathbf{H}(\text{curl})$ and $\mathbf{H}(\text{div})$ IFE functions. The basis functions are some piecewise polynomials satisfying the jump conditions in certain sense to ensure the local approximation property. Particularly, the H^1 IFE space has been developed in Ref. 69, but this is the first time that $\mathbf{H}(\text{curl})$ and $\mathbf{H}(\text{div})$ IFE spaces are systematically developed. Different from all the IFE spaces in literature, the spaces constructed here serve the purpose for approximation under the VEM framework, thus are not limited by the constraint that DoFs need to be imposed on their associated geometric objects. Instead, the DoFs are handled by the IVE spaces discussed in Section 5. To facilitate a simple presentation, we shall focus on the case that elements are only cut by the interface once, i.e., each edge has at most one cutting point, which is a reasonable assumption employed by many works in the literature ^{56,58,69}. In fact, the interface elements may generally satisfy this assumption if the background mesh is sufficiently fine, namely the interface is

locally flat enough. We also remark that IFE spaces can be constructed for more complicated interface element geometry violating this assumption, which we leave to Appendix A for this general case.

We need a linear approximation to the interface portion Γ^K , denoted by Γ_h^K . For example, in Ref. 56 Γ_h^K is constructed as a plane passing through the three cutting points forming a triangle satisfying the maximum angle condition, see the middle plot in Figure 2.1 for an illustration. The following lemma essentially acknowledges this setting. Another widely-used linear approximation approach is to use $\Gamma_h : \phi_h(\mathbf{x}) = 0$ with ϕ_h being the linearization of the sign-distance function ϕ of Γ on the same mesh. These choices indicate that the interface can be well-resolved by a mesh that is sufficiently fine.

Lemma 4.1. *Suppose the mesh is sufficiently fine such that $h < h_0$ for a fixed threshold $h_0 > 0$, then on each interface element $K \in \mathcal{T}_h^i$, there exist constants C_Γ independent of the interface location and mesh size h_K such that for every point $X \in \Gamma^K$ with its orthogonal projection X^\perp onto Γ_h^K ,*

$$\|X - X^\perp\| \leq C_\Gamma h_K^2. \quad (4.1)$$

As the Maxwell equations generally have low regularity near the interface, we only consider the lowest order methods, and thus the $O(h^2)$ approximation geometrical accuracy in (4.1) is sufficient. If high-order methods are desired, one needs to either resolve the interface exactly by using the blending element techniques 53 or approximate the interface by polynomials of order at least $2p - 1$ 70. It is also worthwhile to mention a recent work of VEM 47 for 2D elements with curved edges.

Let Γ_h^K partition K into K_h^\pm , and let α_h and β_h be the piecewise constant functions whose jumps are now across Γ_h^K instead of Γ^K ; namely

$$\alpha_h = \begin{cases} \alpha^- & \text{in } K_h^-, \\ \alpha^+ & \text{in } K_h^+, \end{cases} \quad \beta_h = \begin{cases} \beta^- & \text{in } K_h^-, \\ \beta^+ & \text{in } K_h^+. \end{cases} \quad (4.2)$$

But, here we shall postpone the specific parameters α_h and β_h in the PDEs to a later discussion, and focus on a generic piecewise constant function denoted as c_h to present the IFE functions. In the following discussion, $\mathcal{P}_k(K)$ denotes the polynomial space with degree k on K . Let $\bar{\mathbf{n}}_K$ be the normal vector to Γ_h^K that is approximately in the same direction with \mathbf{n}_K to Γ^K . Define two piecewise constant vector spaces:

$$\mathbf{P}_0^e(c_h; K) = \{\mathbf{c} : \mathbf{c}^\pm = \mathbf{c}|_{K_h^\pm} \in [\mathcal{P}_0(K_h^\pm)]^3, \mathbf{c} \in \mathbf{H}(\text{curl}; K), c_h \mathbf{c} \in \mathbf{H}(\text{div}; K)\}, \quad (4.3a)$$

$$\mathbf{P}_0^f(c_h; K) = \{\mathbf{c} : \mathbf{c}^\pm = \mathbf{c}|_{K_h^\pm} \in [\mathcal{P}_0(K_h^\pm)]^3, \mathbf{c} \in \mathbf{H}(\text{div}; K), c_h \mathbf{c} \in \mathbf{H}(\text{curl}; K)\}. \quad (4.3b)$$

The super scripts e and f emphasize that the two spaces are, respectively, in the edge and face spaces (1-form and 2-form in the language of differential forms);

namely

$$\mathbf{P}_0^f(c_h; K) \subset \mathbf{H}^1(\text{div}, c_h; K) \cap \ker(\text{div}) \quad \text{and} \quad \mathbf{P}_0^e(c_h; K) \subset \mathbf{H}^1(\text{curl}, c'_h, c_h; K) \cap \ker(\text{curl}),$$

where c'_h may be any arbitrary piecewise constant due to the curl-free property of $\mathbf{P}_0^e(c_h; K)$. Hence, for the parameters α_h and β_h , there particularly holds

$$\mathbf{P}_0^e(\beta_h; K) \subset \mathbf{H}^1(\text{curl}, \alpha_h, \beta_h; K) \quad \text{and} \quad \mathbf{P}_0^f(\alpha_h; K) \subset \mathbf{H}^1(\text{div}, \alpha_h; K),$$

which give certain reasonable approximations to these two desired Sobolev spaces. We shall see that these two spaces are the fundamental ingredients to construct all the H^1 , $\mathbf{H}(\text{curl})$ and $\mathbf{H}(\text{div})$ IFE functions, as well as to construct and project the IVE spaces.

Furthermore, from the definition, it is not hard to conclude the following relation

$$\mathbf{P}_0^e(c_h; K) \xrightleftharpoons[c_h^{-1}]{c_h} \mathbf{P}_0^f(c_h^{-1}; K). \quad (4.4)$$

Here the discontinuous coefficient c_h can be viewed as a Hodge star operator. This perspective is the key for computing the projection of the proposed IVE spaces, see Section 5.4.

In order to derive explicit formulas for the functions in the spaces (4.3), we further let $\bar{\mathbf{t}}_K^1$ and $\bar{\mathbf{t}}_K^2$ be the two orthonormal tangential unit vectors to Γ_h^K , and denote the matrix $T_K = [\bar{\mathbf{n}}_K, \bar{\mathbf{t}}_K^1, \bar{\mathbf{t}}_K^2]$. Then, we define the matrices:

$$M_K^{f,c_h} = T_K \begin{bmatrix} 1 & 0 & 0 \\ 0 & \tilde{c} & 0 \\ 0 & 0 & \tilde{c} \end{bmatrix} (T_K)^\top \quad \text{and} \quad M_K^{e,c_h} = T_K \begin{bmatrix} \tilde{c} & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} (T_K)^\top, \quad (4.5)$$

where $\tilde{c} = c_h^+ / c_h^-$. Clearly, both M_K^{f,c_h} and M_K^{e,c_h} are symmetric and positive definite. Thus, the spaces $\mathbf{P}_0^e(c_h; K)$ and $\mathbf{P}_0^f(c_h; K)$ can be rewritten as

$$\mathbf{P}_0^e(c_h; K) = \left\{ \mathbf{c} : \mathbf{c}^\pm = \mathbf{c}|_{K_h^\pm}, \mathbf{c}^- = M_K^{e,c_h} \mathbf{c}^+, \mathbf{c}^+ \in [\mathcal{P}_0(K_h^+)]^3 \right\}, \quad (4.6a)$$

$$\mathbf{P}_0^f(c_h; K) = \left\{ \mathbf{c} : \mathbf{c}^\pm = \mathbf{c}|_{K_h^\pm}, \mathbf{c}^- = M_K^{f,c_h} \mathbf{c}^+, \mathbf{c}^+ \in [\mathcal{P}_0(K_h^+)]^3 \right\}. \quad (4.6b)$$

$\mathbf{P}_0^f(c_h; K)$ and $\mathbf{P}_0^e(c_h; K)$ are subspaces of the piecewise constant vector functions (dimension 6). With the jump conditions as the constraints, it can be easily verified that the dimension of both $\mathbf{P}_0^f(c_h; K)$ and $\mathbf{P}_0^e(c_h; K)$ is 3.

Now, we proceed to present the H^1 , $\mathbf{H}(\text{curl})$ and $\mathbf{H}(\text{div})$ IFE functions. We consider $\mathbf{P}_0^f(a_h; K)$ and $\mathbf{P}_0^e(b_h; K)$, formed by two general positive piecewise constant functions a_h and b_h . Then, all the H^1 , $\mathbf{H}(\text{curl})$ and $\mathbf{H}(\text{div})$ IFE functions with the general parameters a_h and b_h have simple formulas presented in Table 1 where \mathbf{x}_K is any point on Γ^K . One can directly verify that they belong to the corresponding Sobolev spaces and satisfy the associated jump conditions in the table. Note that the normal jump condition in the $\mathbf{H}(\text{curl})$ case and the tangential jump condition in the $\mathbf{H}(\text{div})$ case only hold at the single point \mathbf{x}_K instead of the entire Γ_h^K . This does not violate the necessary continuities for these two spaces to be in

$\mathbf{H}(\operatorname{curl}; K)$ and $\mathbf{H}(\operatorname{div}; K)$. Of course, different choices of \mathbf{x}_K lead to different spaces. In addition, compared with the standard Lagrange, Nédélec, and Raviart-Thomas elements, the only difference for their IFE counterparts is to replace the constant vectors in $[\mathcal{P}_0(K)]^3$ by the vectors in $\mathbf{P}_0^f(a_h; K)$ and $\mathbf{P}_0^e(b_h; K)$, thus providing the necessary piecewise constant approximation to $\beta \nabla u$, $\alpha \operatorname{curl} \mathbf{u}$, and $\beta \mathbf{u}$ on interface elements, respectively.

IFE spaces	$S_h^n(b_h; K)$	$\mathbf{S}_h^e(a_h, b_h; K)$	$\mathbf{S}_h^f(a_h; K)$
Dimension	4	6	4
Sobolev spaces	$H^1(K)$	$\mathbf{H}(\operatorname{curl}; K)$	$\mathbf{H}(\operatorname{div}; K)$
Function format	$\mathbf{b} \cdot (\mathbf{x} - \mathbf{x}_K) + c$ $\mathbf{b} \in \mathbf{P}_0^e(b_h; K),$ $c \in \mathcal{P}_0(K)$	$\mathbf{a} \times (\mathbf{x} - \mathbf{x}_K) + \mathbf{b}$ $\mathbf{a} \in \mathbf{P}_0^f(a_h; K),$ $\mathbf{b} \in \mathbf{P}_0^e(b_h; K)$	$c(\mathbf{x} - \mathbf{x}_K) + \mathbf{a}$ $c \in \mathcal{P}_0(K),$ $\mathbf{a} \in \mathbf{P}_0^f(a_h; K)$
Jump conditions	$[v_h]_{\Gamma_h^K} = 0$ $[b_h \nabla v_h \cdot \bar{\mathbf{n}}]_{\Gamma_h^K} = 0$	$[\mathbf{v}_h \times \bar{\mathbf{n}}]_{\Gamma_h^K} = \mathbf{0}$ $[a_h \operatorname{curl} \mathbf{v}_h \times \bar{\mathbf{n}}]_{\Gamma_h^K} = \mathbf{0}$ $[b_h \mathbf{v}_h \cdot \bar{\mathbf{n}}]_{\mathbf{x}_K} = 0$	$[\mathbf{v}_h \cdot \bar{\mathbf{n}}]_{\Gamma_h^K} = 0$ $[a_h \mathbf{v}_h \times \bar{\mathbf{n}}]_{\mathbf{x}_K} = \mathbf{0}$ $[\operatorname{div} \mathbf{v}_h]_{\Gamma_h^K} = 0$

Table 1: IFE spaces, their dimensions, their function format, the corresponding jump conditions and the Sobolev spaces to which they belong, where \mathbf{x}_K is any point at Γ_h^K .

In addition, on each interface element, these spaces admit a local exact sequence established in the following lemma.

Lemma 4.2. *The following sequence is a complex and is exact:*

$$\mathbb{R} \xrightarrow{\hookrightarrow} S_h^n(b_h; K) \xrightarrow{\operatorname{grad}} \mathbf{S}_h^e(a_h, b_h; K) \xrightarrow{\operatorname{curl}} \mathbf{S}_h^f(a_h; K) \xrightarrow{\operatorname{div}} \mathcal{P}_0(K) \longrightarrow 0. \quad (4.7)$$

Furthermore, the constant vector spaces $\mathbf{P}_0^f(a_h; K)$ and $\mathbf{P}_0^e(b_h; K)$, respectively, are the curl-free and div-free subspaces of $\mathbf{S}_h^e(a_h, b_h; K)$ and $\mathbf{S}_h^f(a_h; K)$:

$$\mathbf{P}_0^e(b_h; K) = \operatorname{grad} S_h^n(b_h; K) = \mathbf{S}_h^e(a_h, b_h; K) \cap \ker(\operatorname{curl}), \quad (4.8a)$$

$$\mathbf{P}_0^f(a_h; K) = \operatorname{curl} \mathbf{S}_h^e(a_h, b_h; K) = \mathbf{S}_h^f(a_h; K) \cap \ker(\operatorname{div}). \quad (4.8b)$$

Proof. It can be verified directly. \square

Remark 4.1. In computation, the IVE functions and their curls are projected

to the constant spaces $\mathbf{P}_0^e(\beta_h; K)$ and $\mathbf{P}_0^f(\alpha_h; K)$. To ensure an optimal first order convergence, the projections need not be to the full IFE spaces $S_h^n(\beta_h; K)$, $\mathbf{S}_h^e(\alpha_h, \beta_h; K)$ and $\mathbf{S}_h^f(\beta_h; K)$. But these spaces will be useful in the computation procedure of the projections.

To end this section, we show trace inequalities for piecewise constant IFE functions. The key is the generic constant is independent of the location of interface.

Lemma 4.3. *Given each interface element K and one of its face F , for every $\mathbf{c} \in \mathbf{P}_0^e(b_h; K)$ or $\mathbf{c} \in \mathbf{P}_0^f(a_h; K)$, there holds that*

$$\|\mathbf{c}\|_{L^2(F)} \lesssim h_K^{-1/2} \|\mathbf{c}\|_{L^2(K)}, \quad (4.9)$$

where the generic constant is independent of the location of interface but depends on a_h or b_h .

Proof. By (4.5) and (4.6), we know that there is a matrix M with $\|M\|_\infty \lesssim 1$ such that

$$\|\mathbf{c}^-\| = \|M\mathbf{c}^+\| \lesssim \|\mathbf{c}^+\|. \quad (4.10)$$

Without loss of generality, we only consider the case that F intersects with the interface and assume F is cut into F_h^\pm . By geometry, it is not hard to see that either there is a pyramid $P \subseteq K_h^+$ which has the base F_h^+ and height $\mathcal{O}(h_K)$ or this is true for the “-” piece. Again, without loss of generality, we assume the former case is true. Then, the standard trace inequality on P simply implies

$$\|\mathbf{c}^+\|_{L^2(F_h^+)} \lesssim h_K^{-1/2} \|\mathbf{c}^+\|_{L^2(K_h^+)} \quad (4.11)$$

As for the “-” piece, we apply the trace inequality on the entire element K with (4.10) to obtain

$$\|\mathbf{c}^-\|_{L^2(F_h^-)} \lesssim h_K^{-1/2} \|\mathbf{c}^-\|_{L^2(K)} \lesssim h_K^{-1/2} (\|\mathbf{c}^-\|_{L^2(K_h^-)} + \|M\mathbf{c}^+\|_{L^2(K_h^+)}) \lesssim \|\mathbf{c}\|_{L^2(K)}. \quad (4.12)$$

Combining (4.11) and (4.12), we have the desired estimate. \square

5. Immersed Virtual Element Spaces

It is generally not possible to construct conforming piecewise polynomial spaces to the Sobolev spaces in (3.2). Traditionally, Lagrange, Nédélec and Raviart-Thomas elements are conforming to H^1 , $\mathbf{H}(\text{curl})$ and $\mathbf{H}(\text{div})$ spaces, yet they cannot provide sufficient approximation when a mesh-cutting interface is present. The IFE spaces introduced above can capture the jump information, but at the cost of losing the conformity. In this section, we construct immersed virtual element (IVE) spaces based on solutions to local interface problems. IVE spaces can be both conforming and satisfy interface conditions perfectly. For a non-interface element K , the local finite element space is simply defined as the linear polynomial space $\mathcal{P}_1(K)$, the

lowest order Nédélec space $\mathcal{ND}_0(K)$ ⁷⁶, and the lowest order Raviart-Thomas space $\mathcal{RT}_0(K)$ ⁸⁰.

Given an interface element $K \in \mathcal{T}_h^i$, we let \mathcal{N}_K and \mathcal{E}_K be the collection of all the nodes and edges in the triangulation of ∂K . Note that the nodes include the vertices in the background mesh and cutting points, and the edges include all the sub-edges cut by the interface and newly-added edges. Then \mathcal{F}_K denotes the resulting triangular faces. Given a T which may be a cube, square, tetrahedron or triangle, we let $\mathcal{ND}_0(T)$ and $\mathcal{RT}_0(T)$ be the first family of Nédélec polynomial space and the Raviart-Thomas polynomial space of the lowest degree on T . The Lagrange space is simply the first-degree polynomial space $\mathcal{P}_1(T)$.

Next, we also need two weighted projections onto the piecewise constant vector spaces $\mathbf{P}_0^f(c_h; K)$ and $\mathbf{P}_0^e(c_h; K)$ which will be used in the definition of the IVE spaces as well as the computation:

$$\boldsymbol{\Pi}_K^{e,c_h} : \mathbf{H}(\text{curl}; K) \rightarrow \mathbf{P}_0^e(c_h; K), \text{ satisfying} \quad (5.1a)$$

$$\int_K c_h \boldsymbol{\Pi}_K^{e,c_h} \mathbf{v}_h \cdot \mathbf{p}_h \, d\mathbf{x} = \int_K c_h \mathbf{v}_h \cdot \mathbf{p}_h \, d\mathbf{x}, \quad \forall \mathbf{p}_h \in \mathbf{P}_0^e(c_h; K),$$

$$\boldsymbol{\Pi}_K^{f,c_h} : \mathbf{H}(\text{div}; K) \rightarrow \mathbf{P}_0^f(c_h; K), \text{ satisfying} \quad (5.1b)$$

$$\int_K c_h \boldsymbol{\Pi}_K^{f,c_h} \mathbf{v}_h \cdot \mathbf{p}_h \, d\mathbf{x} = \int_K c_h \mathbf{v}_h \cdot \mathbf{p}_h \, d\mathbf{x}, \quad \forall \mathbf{p}_h \in \mathbf{P}_0^f(c_h; K).$$

The super-scripts, e and f , still emphasize the distinct Sobolev spaces, i.e., the images of $\boldsymbol{\Pi}_K^{e,c_h}$ and $\boldsymbol{\Pi}_K^{f,c_h}$ belong to $\mathbf{H}(\text{curl}; K)$ and $\mathbf{H}(\text{div}; K)$, respectively.

5.1. The H^1 IVE Space

We first consider the H^1 case. Given the boundary triangulation, we define the boundary function space:

$$\mathcal{B}_h^n(\partial K) = \{v_h \in C(\partial K) : v_h|_T \in \mathcal{P}_1(T), \forall T \in \mathcal{F}_K\}. \quad (5.2)$$

Then, on an interface element K , the H^1 IVE space involving the discontinuous coefficient β is defined as

$$V_h^n(K) = \{v_h : \beta \nabla v_h \in \mathbf{H}(\text{div}; K), \nabla \cdot (\beta \nabla v_h) = 0, v_h|_{\partial K} \in \mathcal{B}_h^n(\partial K)\}. \quad (5.3)$$

Clearly $V_h^n(K) \subseteq H^1(K)$. On the boundary ∂K we use the continuous \mathcal{P}_1 finite element space on the body-fitted surface triangulation. In the interior we use β -harmonic extension so that the shape functions satisfy the jump conditions on the interface.

The property of the nodal DoFs is given by the following lemma.

Lemma 5.1. *The space $V_h^n(K)$ has nodal DoFs $\{v_h(\mathbf{z}), \mathbf{z} \in \mathcal{N}_K\}$.*

Proof. First, $v_h \in V_h^n(K)$ is uniquely determined by the boundary condition in $\mathcal{B}_h^n(\partial K)$. The space $\mathcal{B}_h^n(\partial K)$ has the nodal DoFs associated with the nodes in \mathcal{N}_K . So functions in $V_h^n(K)$ are uniquely determined by their nodal values. \square

On any non-interface element, the standard Lagrange FE space, i.e., $\mathcal{P}_1(K)$, is used. Thus, with the nodal DoFs, we are able to define the global H^1 -conforming IVE space as

$$V_h^n = \{v_h \in H_0^1(\Omega) : v_h|_K \in V_h^n(K), \forall K \in \mathcal{T}_h^i, \text{ and } v_h|_K \in \mathcal{P}_1(K), \forall K \in \mathcal{T}_h^n\}. \quad (5.4)$$

Functions in V_h^n are piecewise linear on the element boundary triangulation and in general non-polynomial inside interface element, which is the key to capture both the jump conditions and conformity.

Similar to the standard VEMs, the function values in the interior are not needed, projections to certain spaces with approximation properties are computed instead. In the following paragraph, we show how to compute $\boldsymbol{\Pi}_K^{e,\beta_h} \nabla v_h$ for $v_h \in V_h^n(K)$. For every $\mathbf{p}_h \in \mathbf{P}_0^e(\beta_h; K)$, by (4.4) there holds $\beta_h \mathbf{p}_h \in \mathbf{P}_0^f(\beta_h^{-1}; K) \subset \mathbf{H}(\text{div}; K) \cap \ker(\text{div})$. Then, the integration by parts shows

$$\int_K \beta_h \boldsymbol{\Pi}_K^{e,\beta_h} \nabla v_h \cdot \mathbf{p}_h \, d\mathbf{x} = \int_K \nabla v_h \cdot (\beta_h \mathbf{p}_h) \, d\mathbf{x} = \int_{\partial K} v_h (\beta_h \mathbf{p}_h \cdot \mathbf{n}) \, ds, \quad (5.5)$$

of which the right-hand side is computable. The L^2 projection of v_h to $\mathcal{P}_0(K)$ is not computable by the definition of the current space. However, this is not needed as v_h itself can be approximated by the formula in Table 1 using the gradient obtained from (5.5). Denote this weighted H^1 projection by \bar{v}_h , the constant c in Table 1 can be chosen such that $\int_{\partial K} \bar{v}_h = \int_{\partial K} v_h$. This constraint gives compactness, thus a sufficient approximation for computing the right-hand side term to guarantee the first order optimal convergence.

5.2. The $\mathbf{H}(\text{curl})$ IVE Space

For the $\mathbf{H}(\text{curl})$ case, the boundary space is defined as

$$\mathcal{B}_h^e(\partial K) = \{\mathbf{v}_h : \mathbf{v}_h|_T \in \mathcal{ND}_0(T), \forall T \in \mathcal{F}_K, (\mathbf{v}_h \cdot \mathbf{t})|_e \text{ is continuous on each } e \in \mathcal{E}_K\}. \quad (5.6)$$

Each $\mathcal{ND}_0(T)$ contains the 2D vector polynomials tangentially defined on the planar triangle T . By formulations from trace finite elements on triangulated surfaces ⁷⁸, $\mathcal{B}_h^e(\partial K)$ is a well-defined finite element space on its own, and has the DoFs of $\int_e \mathbf{v}_h \cdot \mathbf{t} \, ds$, $e \in \mathcal{E}_K$. With this boundary space, we first introduce an auxiliary $\mathbf{H}(\text{curl})$ IVE space:

$$\begin{aligned} \tilde{\mathbf{V}}_h^e(K) = \{\mathbf{v}_h \in \mathbf{H}(\text{curl}; K) : \beta \mathbf{v}_h \in \mathbf{H}(\text{div}; K), \text{ div}(\beta \mathbf{v}_h) = 0, \\ \alpha \text{ curl } \mathbf{v}_h \in \mathbf{H}(\text{curl}; K), \text{ curl } \alpha \text{ curl } \mathbf{v}_h \in \mathbf{P}_0^f(\beta_h^{-1}; K), \mathbf{v}_h^\tau \in \mathcal{B}_h^e(\partial K)\}. \end{aligned} \quad (5.7)$$

Here, α and β can be also understood as Hodge star operators exactly mimicking the second and the third vertical mappings in the desired diagram (3.3) between different Sobolev spaces.

The following lemma gives the well-posedness and DoFs of $\tilde{\mathbf{V}}_h^e(K)$.

Lemma 5.2. $\tilde{\mathbf{V}}_h^e(K)$ is unisolvant with respect to the DoFs $\{\int_e \mathbf{v}_h \cdot \mathbf{t} \, ds, e \in \mathcal{E}_K\}$ and $\{\int_K \beta_h \mathbf{v}_h \cdot \mathbf{p}_h \, dx, \mathbf{p}_h \in \mathbf{P}_0^e(\beta_h; K)\}$.

Proof. Let $\mathbf{q}_h \in \mathbf{P}_0^f(\beta_h^{-1}; K)$ and $\mathbf{r}_h \in \mathcal{B}_h^e(\partial K)$ be some arbitrary data functions. Let us formulate the following local interface problem arising from the definition in (5.7):

$$\operatorname{curl} \alpha \operatorname{curl} \mathbf{v}_h = \mathbf{q}_h, \quad \operatorname{div}(\beta \mathbf{v}_h) = 0, \quad \text{in } K, \quad \mathbf{v}_h^\tau = \mathbf{r}_h \quad \text{on } \partial K. \quad (5.8)$$

Note that the solutions \mathbf{v}_h of (5.8) form the space $\tilde{\mathbf{V}}_h^e(K)$. The well-posedness of (5.8) is given by Lemma 3.3. It implies that the dimension of the solution space is equal to the (finite) dimension of the space of possible data functions: the boundary data function \mathbf{r}_h which is uniquely determined by $\{\int_e \mathbf{r}_h \cdot \mathbf{t} \, ds, e \in \mathcal{E}_K\}$ and the right-hand side $\mathbf{q}_h \in \mathbf{P}_0^f(\beta_h^{-1}; K)$ which has dimension 3. Therefore, the dimension of solution space matches the DoFs count $3 + |\mathcal{E}_K|$ for this virtual space.

In the rest of the proof, it needs to be established that the given moments on edges and element interior are indeed DoFs. To this end, it suffices to show that a function with vanishing DoFs is trivial in this space. Noticing $\operatorname{curl} \alpha \operatorname{curl} \mathbf{v}_h \in \mathbf{P}_0^f(\beta_h^{-1}; K) = \beta_h \mathbf{P}_0^e(\beta_h; K)$ by (4.4), thus from the interior DoFs in Lemma 5.2 we have

$$0 = \int_K \mathbf{v}_h \cdot \operatorname{curl} \alpha \operatorname{curl} \mathbf{v}_h \, dx = \int_K \alpha \operatorname{curl} \mathbf{v}_h \cdot \operatorname{curl} \mathbf{v}_h \, dx - \int_{\partial K} \mathbf{v}_h^\tau \cdot (\alpha \operatorname{curl} \mathbf{v}_h \times \mathbf{n}) \, ds, \quad (5.9)$$

where we have used integration by parts in the second equality. As the edge moments are zero, we know $\mathbf{v}_h^\tau = 0$. So, we have $\int_K \alpha \operatorname{curl} \mathbf{v}_h \cdot \operatorname{curl} \mathbf{v}_h \, dx = 0$ which implies $\operatorname{curl} \mathbf{v}_h = \mathbf{0}$ as α is positive. In addition, by $\operatorname{div}(\beta \mathbf{v}_h) = 0$ and the vanishing trace, we derive from Lemma 3.2 that $\mathbf{v}_h = \mathbf{0}$. \square

Similar to the classical VEM, both \mathbf{v}_h and $\operatorname{curl} \mathbf{v}_h$ are not computable. But we shall see that their weighted projections to the IFE spaces are computable. We first address the projection of $\operatorname{curl} \mathbf{v}_h$ which will be then used to develop a new IVE space as a subspace of $\tilde{\mathbf{V}}_h^e$ that only has the edge DoFs.

By the Hodge star property (4.4), we argue that $\boldsymbol{\Pi}_K^{f,a_h} \operatorname{curl} \mathbf{v}_h$ defined in (5.1b) is computable for any positive a_h . In particular, for each $\mathbf{p}_h \in \mathbf{P}_0^f(a_h; K)$ we know $a_h \mathbf{p}_h \in \mathbf{H}(\operatorname{curl}; K) \cap \ker(\operatorname{curl})$. Then, applying the projection $\boldsymbol{\Pi}_K^{f,a_h}$ to $\operatorname{curl} \mathbf{v}_h$ with the integration by parts, we obtain

$$\int_K a_h \boldsymbol{\Pi}_K^{f,a_h} \operatorname{curl} \mathbf{v}_h \cdot \mathbf{p}_h \, dx = \int_K \operatorname{curl} \mathbf{v}_h \cdot (a_h \mathbf{p}_h) \, dx = \int_{\partial K} (\mathbf{v}_h \times \mathbf{n})(a_h \mathbf{p}_h)^\tau \, ds. \quad (5.10)$$

The right-hand side above is computable through only the edge DoFs $\int_e \mathbf{v}_h \cdot \mathbf{t} \, ds, e \in \mathcal{E}_K$, since the boundary triangulation is known. In computation, $\boldsymbol{\Pi}_K^{f,a_h} \operatorname{curl} \mathbf{v}_h$ provides a sufficient approximation order on elements intersecting the interface.

Now, we let $\mathbf{S}_h^e(a_h, 0; K)$ be the subspace of $\mathbf{S}_h^e(a_h, b_h; K)$ that has the fixed $\mathbf{b} = \mathbf{0}$ in Table 1. Namely, the functions in $\mathbf{S}_h^e(a_h, 0; K)$ are just $\mathbf{w}_h = \mathbf{a} \times (\mathbf{x} - \mathbf{x}_K)$, $\mathbf{a} \in \mathbf{P}_0^f(a_h; K)$, and thus the space only has the dimension 3. Then, based on the auxiliary space $\tilde{\mathbf{V}}_h^e(K)$, we introduce its subspace:

$$\mathbf{V}_h^e(K) = \{\mathbf{v}_h \in \tilde{\mathbf{V}}_h^e(K) : \int_K \operatorname{curl} \mathbf{v}_h \cdot \mathbf{w}_h \, d\mathbf{x} = 0, \forall \mathbf{w}_h \in \mathbf{S}_h^e(\beta_h^{-1}, 0; K)\}. \quad (5.11)$$

Clearly, $\mathbf{V}_h^e(K) \subset \tilde{\mathbf{V}}_h^e(K)$, and the following lemma ensures that only the edge DoFs are needed for the unisolvency of this subspace due to the extra constraint.

Lemma 5.3. $\mathbf{V}_h^e(K)$ is unisolvant with respect to the edge DoFs $\{\int_e \mathbf{v}_h \cdot \mathbf{t} \, ds, e \in \mathcal{E}_K\}$.

Proof. It suffices to show that this extra condition (5.11), in fact, makes the interior DoFs $\int_K \beta_h \mathbf{v}_h \cdot \mathbf{p}_h \, d\mathbf{x}$, $\mathbf{p}_h \in \mathbf{P}_0^e(\beta_h; K)$ fixed thus not degrees of freedom anymore. To see this, for each $\mathbf{p}_h \in \mathbf{P}_0^e(\beta_h; K)$, we have $\beta_h \mathbf{p}_h \in \mathbf{P}_0^f(\beta_h^{-1}; K)$ and $\operatorname{div} \beta_h \mathbf{p}_h = 0$. Therefore, by the local exact sequence (4.7) and (4.4), there exists $\mathbf{w}_h \in \mathbf{S}_h^e(\beta_h^{-1}, 0; K)$ such that $\operatorname{curl} \mathbf{w}_h = \beta_h \mathbf{p}_h$. Then, the integration by parts shows

$$\int_K \beta_h \mathbf{v}_h \cdot \mathbf{p}_h \, d\mathbf{x} = \int_K \mathbf{v}_h \cdot \operatorname{curl} \mathbf{w}_h \, d\mathbf{x} = \underbrace{\int_K \operatorname{curl} \mathbf{v}_h \cdot \mathbf{w}_h \, d\mathbf{x}}_{(I)} - \underbrace{\int_{\partial K} \mathbf{v}_h^\tau \cdot (\mathbf{w}_h \times \mathbf{n}) \, ds}_{(II)}. \quad (5.12)$$

Note that (I) = 0 by the extra condition in the definition. For (II), $\mathbf{v}_h^\tau \in \mathcal{B}_h^e(\partial K)$ is also solely determined by the edge DoFs. Therefore, under the extra constraint, the right-hand side of (5.12) is computable for every pair of \mathbf{p}_h and \mathbf{w}_h , as long as the edge DoFs $\{\int_e \mathbf{v}_h \cdot \mathbf{t} \, ds, e \in \mathcal{E}_K\}$, are given. \square

For the new space $\mathbf{V}_h^e(K)$, the identity (5.12) also gives a simple formula for computing $\Pi_K^{e, \beta_h} \mathbf{v}_h$:

$$\int_K \beta_h \Pi_K^{e, \beta_h} \mathbf{v}_h \cdot \mathbf{p}_h \, d\mathbf{x} = \int_K \beta_h \mathbf{v}_h \cdot \mathbf{p}_h \, d\mathbf{x} = - \int_{\partial K} \mathbf{v}_h^\tau \cdot (\mathbf{w}_h \times \mathbf{n}) \, ds, \quad (5.13)$$

with $\mathbf{w}_h = (\beta_h \mathbf{p}_h / 2) \times (\mathbf{x} - \mathbf{x}_K)$ given by Table 1.

With the edge DoFs, we are able to define a global $\mathbf{H}(\operatorname{curl})$ -conforming space that uses (5.23) on interface elements and standard Nédélec elements on non-interface elements:

$$\mathbf{V}_h^e = \{\mathbf{v}_h \in \mathbf{H}(\operatorname{curl}) : \mathbf{v}_h|_K \in \mathbf{V}_h^e(K), K \in \mathcal{T}_h^i \text{ and } \mathbf{v}_h|_K \in \mathcal{ND}_0(K), K \in \mathcal{T}_h^n\}. \quad (5.14)$$

5.3. The $\mathbf{H}(\operatorname{div})$ IVE Space

For the $\mathbf{H}(\operatorname{div})$ case, the boundary function space is defined as

$$\mathcal{B}_h^f(\partial K) = \{v_h : v_h|_T \in \mathcal{P}_0(T), \forall T \in \mathcal{F}_K\}. \quad (5.15)$$

Similar to the $\mathbf{H}(\text{curl})$ space, let us define the IVE space in the following:

$$\begin{aligned} \mathbf{V}_h^f(K) = \{ \mathbf{v}_h \in H(\text{div}; K) : \text{div}(\mathbf{v}_h) \in \mathcal{P}_0(K), \quad \mathbf{v}_h \cdot \mathbf{n}|_{\partial K} \in \mathcal{B}_h^f(\partial K), \\ \alpha \mathbf{v}_h \in \mathbf{H}(\text{curl}; K), \quad \text{curl}(\alpha \mathbf{v}_h) = \mathbf{0} \}. \end{aligned} \quad (5.16)$$

Again, the discontinuous coefficient α serves as a Hodge star operator that is to mimic the third vertical mapping in Diagram 3.3. The well-posedness and DoFs of this space is given by the following lemma.

Lemma 5.4. $\mathbf{V}_h^f(K)$ is unisolvent with respect to the DoFs $\{\mathbf{v}_h|_F \cdot \mathbf{n}_F, F \in \mathcal{F}_K\}$.

Proof. We let $c_h \in \mathcal{P}_0(K)$ and $r_h \in \mathcal{B}_h^f(\partial K)$ be some arbitrary data functions for the definition (5.16) satisfying the compatibility condition

$$\int_K c_h \, dx = \int_{\partial K} r_h \, ds. \quad (5.17)$$

Then, we consider the following div-curl system:

$$\text{div}(\mathbf{v}_h) = c_h, \quad \text{curl} \alpha \mathbf{v}_h = \mathbf{0} \text{ in } K, \quad \text{and} \quad \mathbf{v}_h \cdot \mathbf{n} = r_h \text{ on } \partial K \quad (5.18)$$

whose solutions \mathbf{v}_h form the space $\mathbf{V}_h^f(K)$. Due to the compatibility condition (5.17), by Lemma 3.1 the system in (5.18) is well-posed and admits a unique solution $\mathbf{v}_h \in \mathbf{H}(\text{div}; K)$ with $\alpha \mathbf{v}_h \in \mathbf{H}(\text{curl}; K)$. The dimension of $\mathbf{V}_h^f(K)$ is just the dimension of space of the independent data functions, $|\mathcal{F}_K|$, where we note that c_h does not count as it is determined by r_h from (5.17).

Next, to show that the given moments are indeed DoFs, we suppose they all vanish, and thus $c_h = r_h = 0$ in (5.18) which immediately implies $\mathbf{v}_h = \mathbf{0}$ by Lemma 3.1. \square

Now, we discuss how to compute projections of the proposed $\mathbf{H}(\text{div})$ IVE space. The identity (5.17), in fact, yields a formula for computing $\text{div}(\mathbf{v}_h)$:

$$\text{div}(\mathbf{v}_h) = |K|^{-1} \int_{\partial K} \mathbf{v}_h \cdot \mathbf{n} \, ds \quad (5.19)$$

which is computable. As for \mathbf{v}_h itself, we then argue that $\mathbf{\Pi}_K^{f,a_h} \mathbf{v}_h$ defined in (5.1b) is always computable for any given positive a_h . Similar to (5.10), given each $\mathbf{p}_h \in \mathbf{P}_0^f(a_h; K)$, we have $a_h \mathbf{p}_h \in \mathbf{P}_0^e(a_h^{-1}; K)$. Then, there exists $\psi_h = (a_h \mathbf{p}_h) \cdot (\mathbf{x} - \mathbf{x}_K) \in S_h^n(a_h^{-1}; K)$ such that $\nabla \psi_h = a_h \mathbf{p}_h$, and thus we can derive

$$\begin{aligned} \int_K a_h \mathbf{\Pi}_K^{f,a_h} \mathbf{v}_h \cdot \mathbf{p}_h \, dx &= \int_K a_h \mathbf{v}_h \cdot \mathbf{p}_h \, dx = \int_K \mathbf{v}_h \cdot \nabla \psi_h \, dx \\ &= - \int_K \text{div}(\mathbf{v}_h) \psi_h \, dx + \int_{\partial K} \mathbf{v}_h \cdot \mathbf{n} \psi_h \, ds \end{aligned} \quad (5.20)$$

which is computable. In computation, a_h in this space is set to α_h as $\alpha_h \mathbf{p}_h \in \mathbf{H}(\text{curl})$.

Thanks to the face DoFs, we can define a global $\mathbf{H}(\text{div})$ -conforming space:

$$\mathbf{V}_h^f = \{ \mathbf{v}_h \in \mathbf{H}(\text{div}; \Omega) : \mathbf{v}_h \in \mathbf{V}_h^f(K), \quad K \in \mathcal{T}_h^i \text{ and } \mathbf{v}_h|_K \in \mathcal{RT}_0(K), \quad K \in \mathcal{T}_h^n \}. \quad (5.21)$$

5.4. Some Comments and Alternative Definitions

The proposed $\mathbf{H}(\text{curl})$ and $\mathbf{H}(\text{div})$ IVE spaces above are exactly the extension of the classical virtual spaces in literature ^{14,15,16,13,17} to the case of discontinuous coefficients. The modification also includes the source terms and the boundary conditions of the local interface problems, by which the weighted projections are computable.

The face triangulation and the associated standard FE space is critical for the appropriate definition of the IVE spaces. Note that the 2D IVE spaces in Ref. 30 may not be developed on faces intersection with the interface, as jump conditions on faces are quite obscure. Take the H^1 interface problem as example. Given a face F , the desired jump information $[\beta \nabla_F v \cdot \hat{\mathbf{n}}_{\Gamma,F}]_{\Gamma}$ cannot be derived from $[v]_{\Gamma}$ and $[\beta \nabla v \cdot \mathbf{n}]_{\Gamma}$, where ∇_F is the surface gradient on F and $\hat{\mathbf{n}}_{\Gamma,F}$ is the normal vector to $\Gamma \cap F$ but parallel to F . We refer readers to the derivation of the jump conditions on interface edges in the 2D case ⁴³ that has to introduce the derivative along the normal direction of element boundary and thus adds much more complexity. Instead, we use well-defined finite element spaces for an interface-fitted triangulation on the boundary faces, which actually makes the theory and computation much simpler. In addition, this approach can also provide sufficient approximation capabilities and keep the DoFs. Furthermore, in the next section, we shall see that it can also benefit implementation through the proposed data structure.

One may note that the definition of the IVE spaces above do not rely on the assumption that the interface only cuts elements once. In fact, all the local problems are automatically well-posed for almost arbitrary interface element configuration, as long as the face triangulation exists. For example, they can be used on elements shown in the right plot of Figure 2.1 (a 2D illustration). This very feature together with the IFE spaces in Appendix A makes the proposed method much more flexible than the traditional IFE methods in the literature.

Next, let us summarize the relation between the involved spaces and weights in the computation of projections, which may be unified as

$$\int_K c_h \mathbf{\Pi}_K^{s,c_h} \mathbf{v}_h \cdot \mathbf{p}_h \, dx = \int_K \mathbf{v}_h \cdot \underbrace{(c_h \mathbf{p}_h)}_{\in \mathbf{P}_0^{s'}(c_h^{-1}; K)} \, dx, \quad (5.22)$$

$$\text{where } \begin{cases} (s, s', c_h) = (e, f, \beta_h), \text{ if } (\mathbf{v}_h, \mathbf{p}_h) \in \mathbf{H}(\text{curl}; K) \times \mathbf{P}_0^e(c_h; K), \\ (s, s', c_h) = (f, e, \alpha_h), \text{ if } (\mathbf{v}_h, \mathbf{p}_h) \in \mathbf{H}(\text{div}; K) \times \mathbf{P}_0^f(c_h; K). \end{cases}$$

By the language of differential forms, in order for the wedge product of a k -form and l -form to be scalar, there needs $k + l = 3$ in the 3D case. Note that \mathbf{v}_h and \mathbf{p}_h both belong to the k -form, $k = 1, 2$, so c_h acts as a Hodge star operator (4.4) mapping \mathbf{p}_h to the $(3 - k)$ -form for the desired wedge product. Here, the value of $c_h = \alpha_h$ or β_h depends on the $\mathbf{H}(\text{curl})$ or $\mathbf{H}(\text{div})$ spaces matching the underlying Maxwell's equations.

At last, we provide an alternative definition of the $\mathbf{H}(\text{curl})$ IVE spaces being

a different subspace of $\tilde{\mathbf{V}}_h^e(K)$, which has some nice mathematical properties. The key is also to impose suitable conditions to assign the interior DoFs.

For the $\mathbf{H}(\text{curl})$ space, we may consider

$$\begin{aligned} \mathbf{V}_h^e(K) = \{ \mathbf{v}_h \in \tilde{\mathbf{V}}_h^e(K) : & \int_K \text{curl } \mathbf{v}_h \cdot \mathbf{w}_h \, d\mathbf{x} = \int_K \boldsymbol{\Pi}_K^{f,\beta_h} \text{curl } \mathbf{v}_h \cdot \mathbf{w}_h \, d\mathbf{x}, \\ & \forall \mathbf{w}_h \in \mathbf{S}_h^e(\beta_h^{-1}, 0; K) \}. \end{aligned} \quad (5.23)$$

Then, the interior DoFs can be determined also through integration by parts:

$$\begin{aligned} \int_K \beta_h \mathbf{v}_h \cdot \mathbf{p}_h \, d\mathbf{x} &= \int_K \text{curl } \mathbf{v}_h \cdot \mathbf{w}_h \, d\mathbf{x} - \int_{\partial K} \mathbf{v}_h^\tau \cdot (\mathbf{w}_h \times \mathbf{n}) \, ds \\ &= \int_K \boldsymbol{\Pi}_K^{f,\beta_h} \text{curl } \mathbf{v}_h \cdot \mathbf{w}_h \, d\mathbf{x} - \int_{\partial K} \mathbf{v}_h^\tau \cdot (\mathbf{w}_h \times \mathbf{n}) \, ds, \quad \forall \mathbf{p}_h \in \mathbf{P}_0^e(\beta_h; K), \end{aligned} \quad (5.24)$$

where $\mathbf{w}_h = (\beta_h \mathbf{p}_h / 2) \times (\mathbf{x} - \mathbf{x}_K)$ from Table 1 makes the space $\mathbf{V}_h^e(K)$ only have the edge DoFs. The identity (5.24) also gives the formula for computing $\boldsymbol{\Pi}_K^{e,\beta_h} \mathbf{v}_h$. But, compared with (5.13), (5.24) needs to compute the extra term $\int_K \boldsymbol{\Pi}_K^{f,\beta_h} \text{curl } \mathbf{v}_h \cdot \mathbf{w}_h \, d\mathbf{x}$, which is slightly more expensive.

This approach to determine the subspaces is similar to the one in Ref. 4, 23 for the classical H^1 virtual spaces. Here, the benefit is to have the new spaces free of the choice of \mathbf{x}_K . Note that the spaces in (5.11) depends on the choice of the point $\mathbf{x}_K \in \Gamma_h^K$ which can be arbitrary on the plane Γ_h^K with a distance $\mathcal{O}(h_K)$ to the element K . However, the new space in (5.23) is invariant with respect to the various $\mathbf{x}_K \in \Gamma_h^K$, even though the underlying IFE spaces $\mathbf{S}_h^e(\beta_h^{-1}, 0; K)$ are not.

5.5. A discrete de Rham Complex

The proposed IVE spaces inherit the de Rham complex properties of standard finite element spaces including the exact sequence and commutativity.

Thanks to the nodal, edge and face DoFs of the proposed IVE spaces, let us first define the corresponding interpolations:

$$I_h^n : H^2(\beta; \mathcal{T}_h) \rightarrow V_h^n \quad \text{satisfying} \quad I_h^n u(\mathbf{x}) = u(\mathbf{x}), \quad \forall \mathbf{x} \in \mathcal{N}_h, \quad (5.25a)$$

$$I_h^e : \mathbf{H}^1(\text{curl}, \alpha, \beta; \mathcal{T}_h) \rightarrow \mathbf{V}_h^e \quad \text{satisfying} \quad \int_e I_h^e \mathbf{u} \cdot \mathbf{t} \, ds = \int_e \mathbf{u} \cdot \mathbf{t} \, ds, \quad \forall e \in \mathcal{E}_h, \quad (5.25b)$$

$$I_h^f : \mathbf{H}^1(\text{div}, \alpha; \mathcal{T}_h) \rightarrow \mathbf{V}_h^f \quad \text{satisfying} \quad \int_F I_h^f \mathbf{u} \cdot \mathbf{n} \, ds = \int_F \mathbf{u} \cdot \mathbf{n} \, ds, \quad \forall F \in \mathcal{F}_h. \quad (5.25c)$$

We further need the standard L^2 projection denoted by $\Pi_K^0 : L^2(K) \rightarrow \mathcal{P}_0(K)$, and define the global one as Π_h^0 such that $\Pi_h^0|_T = \Pi_T^0$, i.e., $\Pi_h^0 : L^2(\Omega) \rightarrow Q_h$ where

$$Q_h = \{v_h : v_h \in \mathcal{P}_0(K), \forall K \in \mathcal{T}_h(K)\}. \quad (5.26)$$

These operators together with the IVE spaces will be used to formulate the continuous and discrete de Rham complex in (1.9). In fact, Lemma 3.4 already shows that exactness in the continuous level. So our focus will be on the discrete one in the lower part of (1.9).

Lemma 5.5. *When Ω is topologically trivial, the following complex is exact*

$$\mathbb{R} \xrightarrow{\hookrightarrow} V_h^n \xrightarrow{\text{grad}} \mathbf{V}_h^e \xrightarrow{\text{curl}} \mathbf{V}_h^f \xrightarrow{\text{div}} Q_h \longrightarrow 0. \quad (5.27)$$

Proof. The argument for showing the sequence being a complex is basically the same as Lemma 3.4. To show the exactness, we can look at the DoFs which form a co-chain exact complex on the cell-complex defined by the mesh. For the completeness, we include a detailed proof below.

First verify $\nabla V_h^n = \ker(\text{curl}) \cap \mathbf{V}_h^e$. By the classic exact sequence, given each $\mathbf{v}_h \in \ker(\text{curl}) \cap \mathbf{V}_h^e$, there exists $v_h \in H^1(\Omega)$ such that $\nabla v_h = \mathbf{v}_h$. Given an interface element K , the jump conditions associated with $\mathbf{V}_h^e(K)$ imply that v_h also satisfies those of $V_h^n(K)$. In addition, let F be one of its face in the boundary triangulation, since $\text{rot}_F \mathbf{v}_h = 0$ and $\mathbf{v}_h^\tau|_F \in \mathcal{ND}_0(F)$, we have $\nabla_F v_h = \mathbf{v}_h^\tau|_F \in [\mathcal{P}_0(F)]^2$, which implies $v_h \in \mathcal{P}_1(F)$. Hence, $v_h \in V_h^n(K)$. On each non-interface element K , \mathbf{v}_h is just a constant vector, so $v_h \in \mathcal{P}_1(K)$. Therefore, we conclude $v_h \in V_h^n$.

Second, we just, to the end, prove $\text{div}(\mathbf{V}_h^f) = Q_h$. Given each $q \in Q_h$, there exists a regular potential $\mathbf{u} \in \mathbf{H}^1(\Omega)$ s.t. $\text{div } \mathbf{u} = q$. Then, we define $\mathbf{u}_h = I_h^f \mathbf{u}$ and

$$|K| q|_K = \int_K \text{div}(\mathbf{u}) \, dx = \int_{\partial K} \mathbf{u} \cdot \mathbf{n} \, ds = \int_{\partial K} I_h^f \mathbf{u} \cdot \mathbf{n} \, ds = \int_K \text{div}(I_h^f \mathbf{u}) \, dx$$

which implies $\text{div}(I_h^f \mathbf{u}|_K) = q|_K$ on each element K finishing the proof.

To verify $\text{curl } \mathbf{V}_h^e = \mathbf{V}_h^f \cap \ker(\text{div})$, we can use a dimension count. Denote by $\#\mathcal{V}_h, \#\mathcal{E}_h, \#\mathcal{F}_h, \#\mathcal{T}_h$ the number of vertices, edges, faces, and elements, respectively. From the surjectivity, i.e., $\text{div}(\mathbf{V}_h^f) = Q_h$, we know $\dim(\mathbf{V}_h^f \cap \ker(\text{div})) = \dim \mathbf{V}_h^f - \dim Q_h = \#\mathcal{F}_h - \#\mathcal{T}_h$. On the other hand, $\dim \text{curl } \mathbf{V}_h^e = \dim \mathbf{V}_h^e - \dim(\ker(\text{curl}) \cap \mathbf{V}_h^e) = \dim \mathbf{V}_h^e - \dim(\nabla V_h^n) = \#\mathcal{E}_h - \#\mathcal{V}_h + 1$. Then by Euler's formula, we get $\dim \text{curl } \mathbf{V}_h^e = \dim(\mathbf{V}_h^f \cap \ker(\text{div}))$. As $\text{curl } \mathbf{V}_h^e \subseteq \mathbf{V}_h^f \cap \ker(\text{div})$, we conclude that they are equal. \square

6. The Immersed Virtual Element Schemes

Based on the previously established spaces and projections, in this section we are ready to present the IVE schemes. With the exact sequence, we also develop fast solvers for the $\mathbf{H}(\text{curl})$ interface problem. At last, we present a data structure that can facilitate an efficient and vectorized implementation of the proposed method.

We shall focus on the H^1 and $\mathbf{H}(\text{curl})$ interface problems due to their vast applications. For simplicity, we let $(\cdot, \cdot)_D$ be the standard L^2 inner product on D .

6.1. The IVE Scheme for H^1 interface problem

For the H^1 case, we define a local bilinear form as

$$b_K(u_h, v_h) = (\beta_h \mathbf{\Pi}_K^{e, \beta_h} \nabla u_h, \mathbf{\Pi}_K^{e, \beta_h} \nabla v_h)_K + S_K((I - \mathbf{\Pi}_K^{e, \beta_h}) \nabla u_h, (I - \mathbf{\Pi}_K^{e, \beta_h}) \nabla v_h), \quad (6.1)$$

where the projection $\mathbf{\Pi}_K^{e, \beta_h}$ is given in (5.1a) on interface elements K and simply assumed to be the identity operator on non-interface elements as the standard FE spaces are used and computable. The first term in (6.1) is a reasonable and computable approximation to $(\beta \nabla u_h, \nabla v_h)$, as $\mathbf{\Pi}_K^{e, \beta_h}$ will preserve the piecewise constant space $\mathbf{P}_0^e(\beta_h; K)$, but it alone does not lead to a stable method as $\mathbf{\Pi}_K^{e, \beta_h} \nabla(\cdot)$ contains a non-trivial kernel. Namely, there exists a non-constant function $v_h \in V_h^n$ s.t. $\mathbf{\Pi}_K^{e, \beta_h} \nabla v_h = 0$. In the VEM literature, two requirements are imposed for the stabilization S_K . One is the k -consistency, i.e., the stabilization vanishes for polynomial spaces of degree k . As $\mathbf{\Pi}_K^{e, \beta_h}$ can preserve the piecewise constant space $\mathbf{P}_0^e(\beta_h; K)$ and the slice operator $I - \mathbf{\Pi}_K^{e, \beta_h}$ is used, S_K is 0-consistent. Another consideration is the norm equivalence $b_K(v_h, v_h) \approx \|\nabla v_h\|^2$. But we really need is the coercivity; see Lemma 6.2 and Section 6.3 below for detailed discussion.

Various choices of the stabilization have been proposed in the literature^{14,23,28} based on different norms on the boundary. In this work, we will employ the following surface H^1 stabilization:

$$S_K(\mathbf{w}_h, \mathbf{z}_h) := \gamma h_K \sum_{F \in \mathcal{F}_K} (\mathbf{w}_h^\tau, \mathbf{z}_h^\tau)_F, \quad (6.2)$$

where $\mathbf{w}_h^\tau|_F$ and $\mathbf{z}_h^\tau|_F$ are the tangential components on the face F . In particular, we note that $(\nabla v_h)^\tau|_F = \nabla v_h - (\nabla v_h \cdot \mathbf{n}_F) \mathbf{n}_F = \nabla_F v_h$ is the surface gradient of v_h , and it is computable since the trace of v_h on ∂K belongs to the standard FE space and is known. As the standard FE spaces are defined on the boundary triangulation, the stabilization in (6.2) must be piecewisely computed. Then, the global bilinear form is defined as

$$b_h(u_h, v_h) = \sum_{K \in \mathcal{T}_h} b_K(u_h, v_h). \quad (6.3)$$

The proposed IVE scheme is to find $u_h \in V_h^n$ such that

$$b_h(u_h, v_h) = \sum_{K \in \mathcal{T}_h} \int_K f \tilde{\mathbf{\Pi}}_K^{e, \beta_h} v_h \, d\mathbf{x}, \quad \forall v_h \in V_h^n. \quad (6.4)$$

Note that the projection of v_h itself is not computable for the current space, and thus we simply employ the approximated gradient $\tilde{\mathbf{\Pi}}_K^{e, \beta_h} \nabla v_h$ and the formula in Table 1 to form

$$\tilde{\mathbf{\Pi}}_K^{e, \beta_h} v_h = (\mathbf{\Pi}_K^{e, \beta_h} \nabla v_h) \cdot (\mathbf{x} - \mathbf{x}_K) + c \quad (6.5)$$

with the constant c chosen such that $\int_{\partial K} \tilde{\mathbf{\Pi}}_K^{e, \beta_h} v_h \, ds = \int_{\partial K} v_h \, ds$. Clearly, there holds

$$\nabla \tilde{\mathbf{\Pi}}_K^{e, \beta_h} v_h = \mathbf{\Pi}_K^{e, \beta_h} \nabla v_h. \quad (6.6)$$

In fact, the H^1 interface problem is not our focus. Only the stiffness matrix for the H^1 interface problem is needed for the auxiliary space preconditioner in the fast solver for the $\mathbf{H}(\text{curl})$ interface problems.

The stabilization in (6.2) indeed leads to a stable method which is given by the following results.

Lemma 6.1. *For each interface element K , there exists a constant depending only on the shape regularity of K , the coefficient β and the parameter γ s.t.*

$$\|\sqrt{\beta} \nabla v_h\|_{L^2(K)} \lesssim |v_h|_{H^{1/2}(\partial K)} \lesssim h_K^{1/2} |v_h|_{H^1(\partial K)} \quad v_h \in V_h^n(K). \quad (6.7)$$

Proof. We use the energy minimization argument. Given each $v_h \in V_h^n(K)$, consider an arbitrary function $w_h \in H^1(K)$ such that $w_h - v_h = 0$ on ∂K . Then, integration by parts on K^\pm with the flux jump condition $[\beta \nabla v_h \cdot \mathbf{n}]|_{\Gamma_K} = 0$ and $\nabla \cdot (\beta \nabla v_h) = 0$ yields

$$\int_K \beta \nabla v_h \cdot \nabla (v_h - w_h) \, d\mathbf{x} = \int_{\partial K} \beta \nabla v_h \cdot \mathbf{n} (v_h - w_h) \, ds = 0.$$

On one hand, with the Hölder's inequality, it implies that v_h minimizes the $\|\sqrt{\beta} \nabla \cdot\|_{L^2(K)}$ energy norm, i.e.,

$$\|\sqrt{\beta} \nabla v_h\|_{L^2(K)} \leq \|\sqrt{\beta} \nabla w_h\|_{L^2(K)}. \quad (6.8)$$

On the other hand, by the inverse trace theorem given in Section 27 in Ref. 23, we have a function z_h such that $z_h = v_h$ on ∂K and $|z_h|_{H^1(K)} \lesssim |v_h|_{H^{1/2}(\partial K)}$. Hence, using (6.8) with $w_h = z_h$ we arrive at

$$\|\sqrt{\beta} \nabla v_h\|_{L^2(K)} \lesssim \|\nabla z_h\|_{L^2(K)} \lesssim |v_h|_{H^{1/2}(\partial K)}, \quad (6.9)$$

which gives the first inequality in (6.7). The second inequality in (6.7) simply follows from (2.16) in Ref. 23. \square

Lemma 6.2. *There holds that*

$$\|\sqrt{\beta} \nabla v_h\|_{L^2(K)}^2 \lesssim b_K(v_h, v_h), \quad \forall v_h \in V_h^n(K), \quad (6.10)$$

where the constant depends only on the shape regularity of K , β , and γ .

Proof. We will use the projection $\tilde{\Pi}_K^{e, \beta_h}$ and the relation (6.6). By Lemma 6.1, the triangular inequality and the trace inequality by Lemma 4.3, we have

$$\begin{aligned} \|\sqrt{\beta} \nabla v_h\|_{L^2(K)} &\lesssim h_K^{1/2} |v_h|_{H^1(\partial K)} \lesssim h_K^{1/2} |\tilde{\Pi}_K^{e, \beta_h} v_h|_{H^1(\partial K)} + h_K^{1/2} |v_h - \tilde{\Pi}_K^{e, \beta_h} v_h|_{H^1(\partial K)} \\ &\lesssim \|\tilde{\Pi}_K^{e, \beta_h} \nabla v_h\|_{L^2(K)} + h_K^{1/2} \|\nabla_F (v_h - \tilde{\Pi}_K^{e, \beta_h} v_h)\|_{L^2(\partial K)} \lesssim b_K(v_h, v_h), \end{aligned} \quad (6.11)$$

which finishes the proof. \square

6.2. The IVE scheme for the $\mathbf{H}(\mathbf{curl})$ interface problem

In this case, we need to deal with the terms of $\mathbf{curl} \mathbf{curl} \mathbf{u}$ and \mathbf{u} separately. For the same reason discussed above, we need to project both $\mathbf{curl} \mathbf{u}_h$ and \mathbf{u}_h and then add their associated stabilization terms to enforce coercivity. For the $\mathbf{curl} \mathbf{curl}$ term, we introduce

$$\begin{aligned} a_K^1(\mathbf{u}_h, \mathbf{v}_h) = & (\alpha_h \boldsymbol{\Pi}_K^{f,\alpha_h} \mathbf{curl} \mathbf{u}_h, \boldsymbol{\Pi}_K^{f,\alpha_h} \mathbf{curl} \mathbf{v}_h)_K \\ & + S_K^1((I - \boldsymbol{\Pi}_K^{f,\alpha_h}) \mathbf{curl} \mathbf{u}_h, (I - \boldsymbol{\Pi}_K^{f,\alpha_h}) \mathbf{curl} \mathbf{v}_h), \end{aligned} \quad (6.12)$$

where, similarly, $\boldsymbol{\Pi}_K^{f,\alpha_h}$ is chosen as (5.1b) on interface elements but just the identity operator on non-interface elements. As $\mathbf{curl} \mathbf{V}_h^e \subset \mathbf{V}_h^f$, the stabilization term is defined as

$$S_K^1(\mathbf{w}_h, \mathbf{z}_h) = \gamma_1 h \sum_{F \in \mathcal{F}_K} (\mathbf{w}_h \cdot \mathbf{n}_F, \mathbf{z}_h \cdot \mathbf{n}_F)_F, \quad (6.13)$$

where we note that $\mathbf{curl} \mathbf{u}_h \cdot \mathbf{n} = \text{rot}_F \mathbf{u}_h$ for $\mathbf{u}_h \in \mathbf{V}_h^e$ can be computed through the formula in (3.6) with $\text{rot}_F \mathbf{u}_h = |F|^{-1} \int_{\partial F} \mathbf{u}_h \cdot \mathbf{t} \, ds$ on each triangular face F .

The bilinear form for the weighted L^2 inner product is defined as

$$a_K^0(\mathbf{u}_h, \mathbf{v}_h) = (\beta_h \boldsymbol{\Pi}_K^{e,\beta_h} \mathbf{u}_h, \boldsymbol{\Pi}_K^{e,\beta_h} \mathbf{v}_h)_K + S_K^0((I - \boldsymbol{\Pi}_K^{e,\beta_h}) \mathbf{u}_h, (I - \boldsymbol{\Pi}_K^{e,\beta_h}) \mathbf{v}_h), \quad (6.14)$$

where $\boldsymbol{\Pi}_K^{e,\beta_h}$ is defined in (5.13) on interface elements and the identity on non-interface elements. The stabilization is given by

$$S_K^0(\mathbf{w}_h, \mathbf{z}_h) = \gamma_0 \sum_{F \in \mathcal{F}_K} (\mathbf{w}_h^\tau, \mathbf{z}_h^\tau)_F, \quad (6.15)$$

where \mathbf{w}_h^τ and \mathbf{z}_h^τ still denote the tangential components onto each face F . With the triangulation on faces, $\mathbf{w}_h^\tau \in \mathbf{V}_h^e$ is computable through the edge DoFs. We highlight that the scaling $h^0 = 1$ in the stabilization $S_K^0(\mathbf{w}_h, \mathbf{z}_h)$ is different from the usual h in classical VEM in Ref. 17, 15, 16, 19, and this is also the key for the proposed method to produce optimal convergent solutions. Changing the scaling from $\mathcal{O}(h)$ to $\mathcal{O}(1)$ may increase the consistency error locally. More precisely $S_K^0(\mathbf{v}_h, \mathbf{v}_h) = \mathcal{O}(h_K^2)$ while $(\beta_h \boldsymbol{\Pi}_K^{e,\beta_h} \mathbf{v}_h, \boldsymbol{\Pi}_K^{e,\beta_h} \mathbf{v}_h)_K = \mathcal{O}(h_K^3)$. But such a loss of order h is restricted to the interface elements only whose number is $\mathcal{O}(h)$ fraction of the total number of elements. So overall the L^2 -norm is still possible of optimal order. The theoretical justification has been given for the 2D case in Ref. 30 and will be explored in a forthcoming paper for the 3D case.

Then, we can define the global bilinear form

$$a_h(\mathbf{u}_h, \mathbf{v}_h) = \sum_{K \in \mathcal{T}_h} a_K^1(\mathbf{u}_h, \mathbf{v}_h) + a_K^0(\mathbf{u}_h, \mathbf{v}_h). \quad (6.16)$$

The proposed IVE scheme for the $\mathbf{H}(\mathbf{curl})$ interface problem is to find $\mathbf{u}_h \in \mathbf{V}_h^e$ such that

$$a_h(\mathbf{u}_h, \mathbf{v}_h) = \sum_{K \in \mathcal{T}_h} \int_K \mathbf{f} \cdot \boldsymbol{\Pi}_K^{e,\beta_h} \mathbf{v}_h \, dx, \quad \forall \mathbf{v}_h \in \mathbf{V}_h^e. \quad (6.17)$$

Next, we show that the proposed stabilization can indeed make the bilinear form coercive.

Lemma 6.3. *There exists a constant depending only on the shape regularity of K s.t.*

$$\|\mathbf{v}_h\|_{L^2(K)} \lesssim h_K^{1/2} \|\mathbf{v}_h \cdot \mathbf{n}\|_{L^2(\partial K)} \quad \mathbf{v}_h \in \mathbf{V}_h^f(K). \quad (6.18)$$

Proof. As $\operatorname{curl}(\alpha \mathbf{v}_h) = \mathbf{0}$, by (3.9), we only need to estimate $\operatorname{div}(\mathbf{v}_h)$. Noticing $\operatorname{div}(\mathbf{v}_h)$ is a constant, we can write down

$$\begin{aligned} \|\operatorname{div}(\mathbf{v}_h)\|_{L^2(K)} &= |K|^{-1/2} \left| \int_K \operatorname{div}(\mathbf{v}_h) \, dx \right| \\ &= |K|^{-1/2} \left| \int_{\partial K} \mathbf{v}_h \cdot \mathbf{n} \, ds \right| \lesssim h_K^{-1/2} \|\mathbf{v}_h \cdot \mathbf{n}\|_{L^2(\partial K)}, \end{aligned} \quad (6.19)$$

where we have used that $|K|/|\partial K| \approx h_K$. \square

Lemma 6.4. *For every function $\mathbf{v}_h \in \mathbf{V}_h^e$, there holds*

$$\|\operatorname{curl} \mathbf{v}_h\|_{L^2(K)}^2 \lesssim a_K^1(\mathbf{v}_h, \mathbf{v}_h), \quad (6.20a)$$

$$\|\mathbf{v}_h\|_{L^2(K)}^2 \lesssim a_K^0(\mathbf{v}_h, \mathbf{v}_h), \quad (6.20b)$$

where the constants depend only on the shape regularity of K , β , and γ_1, γ_2 .

Proof. Let us first show (6.20a). By Lemma 6.3 and the de Rham complex, we have

$$\|\operatorname{curl} \mathbf{v}_h\|_{L^2(K)} \lesssim h_K^{1/2} \|\operatorname{curl} \mathbf{v}_h \cdot \mathbf{n}\|_{L^2(\partial K)}. \quad (6.21)$$

Then, we apply the trace inequality for the $\mathbf{H}(\operatorname{div})$ functions by Lemma 4.3 to obtain

$$\begin{aligned} \|\operatorname{curl} \mathbf{v}_h \cdot \mathbf{n}\|_{L^2(\partial K)} &\lesssim \|\boldsymbol{\Pi}_K^{f,\alpha_h} \operatorname{curl} \mathbf{v}_h \cdot \mathbf{n}\|_{L^2(\partial K)} \\ &\quad + \|\operatorname{curl} \mathbf{v}_h \cdot \mathbf{n} - \boldsymbol{\Pi}_K^{f,\alpha_h} \operatorname{curl} \mathbf{v}_h \cdot \mathbf{n}\|_{L^2(\partial K)} \\ &\lesssim h_K^{-1/2} \|\boldsymbol{\Pi}_K^{f,\alpha_h} \operatorname{curl} \mathbf{v}_h\|_{L^2(K)} \\ &\quad + \|\operatorname{curl} \mathbf{v}_h \cdot \mathbf{n} - \boldsymbol{\Pi}_K^{f,\alpha_h} \operatorname{curl} \mathbf{v}_h \cdot \mathbf{n}\|_{L^2(\partial K)}. \end{aligned} \quad (6.22)$$

Substituting (6.22) into (6.21) yields (6.20a). As for (6.20b), applying (3.19) with the appropriate scaling, we have

$$\|\mathbf{v}_h\|_{L^2(K)} \lesssim h_K \|\operatorname{curl} \mathbf{v}_h\|_{L^2(K)} + h_K^{1/2} \|\mathbf{v}_h \times \mathbf{n}\|_{L^2(\partial K)}. \quad (6.23)$$

For the second term, we apply the trace inequality for the $\mathbf{H}(\operatorname{curl})$ functions by Lemma 4.3 to obtain

$$\begin{aligned} \|\mathbf{v}_h \times \mathbf{n}\|_{L^2(\partial K)} &\lesssim \|\boldsymbol{\Pi}_K^{e,\beta_h} \mathbf{v}_h \times \mathbf{n}\|_{L^2(\partial K)} + \|(\mathbf{v}_h - \boldsymbol{\Pi}_K^{e,\beta_h} \mathbf{v}_h) \times \mathbf{n}\|_{L^2(\partial K)} \\ &\lesssim h_K^{-1/2} \|\boldsymbol{\Pi}_K^{e,\beta_h} \mathbf{v}_h\|_{L^2(K)} + \|(\mathbf{v}_h - \boldsymbol{\Pi}_K^{e,\beta_h} \mathbf{v}_h)^\tau\|_{L^2(\partial K)}. \end{aligned} \quad (6.24)$$

Putting (6.22) and (6.24) into (6.23), we have the desired estimate. \square

We emphasize that the coercivity constants depend only on the shape regularity of the underlying triangulation, the coefficient β , and the parameter γ , but most importantly, not on the location of the intersection points, i.e., robust to the cut of the interface.

Remark 6.1. Lemmas 6.2 and 6.4 immediately imply that $b_h(\cdot, \cdot)$ and $a_h(\cdot, \cdot)$ are norms on V_h^n and \mathbf{V}_h^e , respectively. These two lemmas hold regardless of the choice of $\gamma > 0$, $\gamma_1 > 0$ and $\gamma_2 > 0$, i.e., the method does not need those parameters to be large enough required by many traditional unfitted-mesh methods ^{26,71}, and thus the resulting linear systems are always positive-definite. Roughly speaking, it can be understood that the proposed IVE scheme is “more conforming” such that weaker weights are needed in the stabilization. This is particularly important for the $\mathbf{H}(\text{curl})$ problem, as we do not need to use h^{-1} scaling in the stabilization, which can avoid the suboptimal convergence in (1.8). Instead, $\mathcal{O}(1)$ and $\mathcal{O}(h)$ scaling are used for the stabilization associated with \mathbf{u}_h and $\text{curl } \mathbf{u}_h$ terms, which is key to achieve optimal convergence by our numerical experiments. Nevertheless, the rigorous analysis is still very involved, and in the next subsection we shall briefly describe the challenges.

6.3. *Comments on the norm equivalence and error analysis*

In the vast VEM literature ^{14,19,23}, the norm equivalence results are desired for error and stability analysis:

$$\|\nabla v_h\|_{L^2(K)}^2 \lesssim b_K(v_h, v_h) \lesssim \|\nabla v_h\|_{L^2(K)}^2 \quad \forall v_h \in V_h^n(K), \quad (6.25a)$$

$$\|\mathbf{v}_h\|_{\mathbf{H}(\text{curl}; K)}^2 \lesssim a_K(\mathbf{v}_h, \mathbf{v}_h) \lesssim \|\mathbf{v}_h\|_{\mathbf{H}(\text{curl}; K)}^2 \quad \forall \mathbf{v}_h \in \mathbf{V}_h^e(K). \quad (6.25b)$$

The left inequalities in (6.25), i.e., the coercivity, are given by Lemmas 6.2 and 6.4, respectively, in which the constants are independent of interface location. Although the right two inequalities in (6.25) indeed hold, their constants may depend on the interface location, as the inverse inequalities on the boundary triangulation are needed in the analysis.

Let us take the H^1 case as an illustration example. By the boundedness property of the projection $\boldsymbol{\Pi}_K^{e, \beta_h}$, we trivially have

$$\|\boldsymbol{\Pi}_K^{e, \beta_h} \nabla v_h\|_{L^2(K)} \leq C_{Pr} \|\nabla v_h\|_{L^2(K)},$$

where the constant C_{Pr} only depends on the geometry of K . The problem is on the

stabilization term. We may prove

$$\begin{aligned}
& h_K^{1/2} \|\nabla_F(I - \tilde{\Pi}_K^{e,\beta_h})v_h\|_{L^2(\partial K)} \\
& \leq C_{\text{inv}} \|(I - \tilde{\Pi}_K^{e,\beta_h})v_h\|_{L^2(\partial K)} \\
& \leq C_{\text{inv}} C_t \left(h_K^{-1/2} \|(I - \tilde{\Pi}_K^{e,\beta_h})v_h\|_{L^2(K)} + h_K^{1/2} |(I - \tilde{\Pi}_K^{e,\beta_h})v_h|_{H^1(K)} \right) \\
& \leq C_{\text{inv}} C_t (C_{Pc} + 1) h_K^{1/2} |(I - \tilde{\Pi}_K^{e,\beta_h})v_h|_{H^1(K)} \\
& \leq C_{\text{inv}} C_t (C_{Pc} + 1) (C_{Pr} + 1) h_K^{1/2} \|\Pi_K^{e,\beta_h} \nabla v_h\|_{L^2(K)},
\end{aligned}$$

where the first inequality is an inverse inequality on the surface triangulation with the constant C_{inv} , and the second and the third ones are the trace and Poincaré inequalities with the constants C_t and C_{Pc} . Note that C_t , C_{Pr} and C_{Pc} only depend on the geometry of K ; but C_{inv} depends on the element boundary triangulation which contain anisotropic triangles, and shrinking elements may make C_{inv} blow up. Indeed, restricting to the boundary $(I - \tilde{\Pi}_K^{e,\beta_h})v_h|_F$ is linear, and its surface gradient can be computed exactly using the cot formulae. The existence of small angles in the boundary triangulation will make the corresponding entry large, and thus robust norm equivalence may not hold. The similar issue applies to the $\mathbf{H}(\text{curl})$ case.

In fact, for 3D VEM, to our best knowledge, almost all the analysis in the literature require shape regularity of both the elements and faces such that the norm equivalence above can hold. In our case, however, the boundary triangulation does not satisfy the shape regularity causing essential difficulties for analysis. An alternative approach is to use the “error equation” approach [27,29,30](#) that may overcome the shape regularity issue. Careful study of the robustness to the shape of boundary triangulations is needed.

6.4. Implementation

Inherited from the classical VEM, implementation of the proposed algorithm is highly vectorized. Computing the projections from IVE spaces and assembly of matrices significantly outperform the classical IFE methods. To see this, following Ref. [37](#), we describe a `face2elem` and a `face` data structure that can greatly facilitate the implementation. `face2elem` is a vector mapping from each (local) face’s index to its mother element’s index. `face` is a matrix containing each face’s DoFs (node or edge) on its rows. Here, we use the tetrahedral interface elements in Figure [6.1](#) to illustrate the data structures. In Figure [6.1](#), the red and blue segments are, respectively, cutting edges by the interface and newly added edges for the surface triangulation. The indices are shown on the two plots for all the vertices and edges. Suppose the index of this element is 1, and then the desired data structures of `face2elem` and `face` are shown in [\(6.26\)](#).

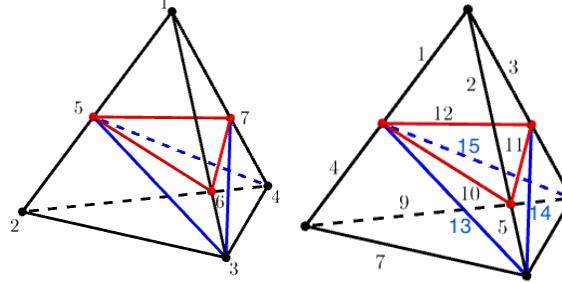


Fig. 6.1: Indices of nodes (left) and edges (right) of an interface element.

face2elem :	<table border="1"> <tr><td>1</td><td>1</td></tr> <tr><td>2</td><td>1</td></tr> <tr><td>3</td><td>1</td></tr> <tr><td>4</td><td>1</td></tr> <tr><td>5</td><td>1</td></tr> <tr><td>6</td><td>1</td></tr> <tr><td>7</td><td>1</td></tr> <tr><td>8</td><td>1</td></tr> <tr><td>9</td><td>1</td></tr> <tr><td>10</td><td>1</td></tr> </table>	1	1	2	1	3	1	4	1	5	1	6	1	7	1	8	1	9	1	10	1	<table border="1"> <tr><td>1</td><td>5</td><td>6</td></tr> <tr><td>2</td><td>1</td><td>6</td><td>7</td></tr> <tr><td>3</td><td>1</td><td>7</td><td>5</td></tr> <tr><td>4</td><td>2</td><td>3</td><td>5</td></tr> <tr><td>5</td><td>3</td><td>6</td><td>5</td></tr> <tr><td>6</td><td>3</td><td>4</td><td>7</td></tr> <tr><td>7</td><td>7</td><td>6</td><td>3</td></tr> <tr><td>8</td><td>4</td><td>5</td><td>7</td></tr> <tr><td>9</td><td>2</td><td>4</td><td>5</td></tr> <tr><td>10</td><td>2</td><td>3</td><td>4</td></tr> </table>	1	5	6	2	1	6	7	3	1	7	5	4	2	3	5	5	3	6	5	6	3	4	7	7	7	6	3	8	4	5	7	9	2	4	5	10	2	3	4	<table border="1"> <tr><td>1</td><td>10</td><td>2</td></tr> <tr><td>2</td><td>11</td><td>3</td></tr> <tr><td>3</td><td>12</td><td>1</td></tr> <tr><td>4</td><td>7</td><td>13</td><td>4</td></tr> <tr><td>5</td><td>13</td><td>5</td><td>10</td></tr> <tr><td>6</td><td>8</td><td>6</td><td>14</td></tr> <tr><td>7</td><td>5</td><td>14</td><td>11</td></tr> <tr><td>8</td><td>6</td><td>12</td><td>15</td></tr> <tr><td>9</td><td>4</td><td>9</td><td>15</td></tr> <tr><td>10</td><td>7</td><td>8</td><td>9</td></tr> </table>	1	10	2	2	11	3	3	12	1	4	7	13	4	5	13	5	10	6	8	6	14	7	5	14	11	8	6	12	15	9	4	9	15	10	7	8	9
1	1																																																																																																		
2	1																																																																																																		
3	1																																																																																																		
4	1																																																																																																		
5	1																																																																																																		
6	1																																																																																																		
7	1																																																																																																		
8	1																																																																																																		
9	1																																																																																																		
10	1																																																																																																		
1	5	6																																																																																																	
2	1	6	7																																																																																																
3	1	7	5																																																																																																
4	2	3	5																																																																																																
5	3	6	5																																																																																																
6	3	4	7																																																																																																
7	7	6	3																																																																																																
8	4	5	7																																																																																																
9	2	4	5																																																																																																
10	2	3	4																																																																																																
1	10	2																																																																																																	
2	11	3																																																																																																	
3	12	1																																																																																																	
4	7	13	4																																																																																																
5	13	5	10																																																																																																
6	8	6	14																																																																																																
7	5	14	11																																																																																																
8	6	12	15																																																																																																
9	4	9	15																																																																																																
10	7	8	9																																																																																																
face(nodes) :		face(edges) :	(6.26)																																																																																																

The key feature of VEM in implementation is to compute the projections through the DoFs. Let us use $\Pi_K^{f,\alpha_h} \operatorname{curl} \mathbf{v}_h$ and the formula in (5.10) as an example to describe the procedure. Given an element K with the global index i , $i = 1, 2, \dots, |\mathcal{T}_h|$, we need to compute $\Pi_K^{f,\alpha_h} \operatorname{curl} \varphi_{h,k}^e$ for a local edge index k , $k = 1, 2, \dots, |\mathcal{E}_K|$. Here $\Pi_K^{f,\alpha_h} \operatorname{curl} \varphi_{h,k}^e$ is a constant vector denoted as $\mathbf{c}_k \in \mathbf{P}_0^f(\alpha_h; K)$ with $\mathbf{c}_k^\pm := (\Pi_K^{f,\alpha_h} \operatorname{curl} \varphi_{h,k}^e)|_{K_h^\pm}$, where $\varphi_{h,k}^e$ is the edge shape function with respect to the k -th edge. Now, we let the test function \mathbf{p}_h in (5.10) be the three unit vectors: $\mathbf{p}_{h,l}^+ = \mathbf{e}_l$, $l = 1, 2, 3$, with $\mathbf{p}_{h,l}^- = M_K^{f,\alpha_h} \mathbf{p}_{h,l}^+$ with M_K^{f,α_h} given by (4.5). Then, we can rewrite (5.10) into a matrix-vector equation only about \mathbf{c}_k^- :

$$\begin{aligned}
 & (\alpha^- |K_h^-| + \alpha^+ |K_h^+|) (M_K^{f,\alpha_h})^\top M_K^{f,\alpha_h} \mathbf{c}_k^- \\
 &= \int_{\partial K} (\varphi_{h,k}^e \times \mathbf{n}) [(\alpha_h \mathbf{p}_{h,1})^\tau, (\alpha_h \mathbf{p}_{h,2})^\tau, (\alpha_h \mathbf{p}_{h,3})^\tau]^\top \, ds, \tag{6.27}
 \end{aligned}$$

where $\varphi_{h,k}^e \times \mathbf{n}$ is a rotation of $(\varphi_{h,k}^e)^\tau$. $(\varphi_{h,k}^e)^\tau$ is a 2D Nédélec polynomial function tangentially defined on each face, and the integration of this function associated with the k -th edge can be determined by the two nodes retrieved from the face data structure. Particularly, if the k -th edge does not belong to the boundary of a face F , then there is no contribution of this face to the right-hand side of (6.27).

With (6.27), we highlight that the geometric information needed in the computation has been automatically encoded in the data structures `face2elem` and `face`, which is the key for the efficient vectorized code. We report the CPU time for computing the projections and generate matrices in Table 2 to demonstrate the efficiency. This very feature makes the proposed method distinguished from all the classical IFE methods in the literature that have to use more detailed geometric information to compute the IFE functions.

6.5. Preconditioning

Solving the resulting linear system from Maxwell's equations is one of the central challenges in computational electromagnetism, and the interface may make it even more difficult. With a slight abuse of notation, we denote the linear system from the proposed IVE discretization of the $\mathbf{H}(\text{curl})$ interface problem as

$$A\mathbf{u}_h = \mathbf{f}_h, \quad (6.28)$$

where $\mathbf{u}_h \in \mathbb{R}^{\#\text{edge}}$ denotes the vector representation in DoFs. This system is solved by the preconditioned conjugate gradient (PCG) method. To our best knowledge, the development of fast solvers of VEM specifically for $H(\text{curl})$ -equations has not been discussed in any literature. Without suitable preconditioners, the PCG solver can be extremely slow, see the comparison in Table 4. In this work, we develop a fast solver for IVE discretization of the $\mathbf{H}(\text{curl})$ interface problem that involves two techniques. Thanks to the de Rham complex, the first one is the *auxiliary space preconditioner* for the $\mathbf{H}(\text{curl})$ equation which is developed by Hiptmair and Xu in Ref. 60 (HX preconditioner) based on the auxiliary space framework in Ref. 85. The second one is a block diagonal smoother to handle the anisotropic element shape near the interface. In the experiments, both are used the implementation in *iFEM*³³.

The resulting HX-preconditioner for the $\mathbf{H}(\text{curl})$ systems is in the form

$$B^{\text{curl}} = R^{\text{curl}} + \Pi \mathbf{B}^{\text{grad}} \Pi^\top + G \mathbf{B}^{\text{grad}} G^\top, \quad (6.29)$$

which consists of the following three components:

- a smoother R^{curl} of the $\mathbf{H}(\text{curl})$ matrix A ,
- an algebraic multigrid (AMG) solver B^{grad} for a scalar Laplacian matrix,
- an AMG solver \mathbf{B}^{grad} for a vector Laplacian matrix.

We simply employ the incidence matrix associated with the operator $\nabla : \mathbf{V}_h^n \rightarrow \mathbf{V}_h^e$ as the discrete gradient matrix G which resembles that from the lowest order Nédélec element on simplicial meshes. G maps the nodal DoFs (columns) to edge DoFs (rows). There are two nonzero entry, ± 1 , on each row. The columns of these entries correspond to the nodes of the edge. The sign is determined by the global orientation of an edge. The node-to-edge transfer matrix is denoted by $\Pi : \prod_{i=1}^3 \mathbb{R}^{\#\text{node}} \rightarrow \mathbb{R}^{\#\text{edge}}$. Note that these two matrices being well-defined are based on the node and edge DoFs of the H^1 and $\mathbf{H}(\text{curl})$ IVE spaces.

For $\mathbf{H}(\text{curl})$ problems, it is known that a multigrid solver for Poisson-type equations is not sufficient since the discrete operator corresponding to $\text{curl}(\alpha \text{curl}) + \beta I$ behaves differently for a gradient field and a solenoidal field (see e.g., Ref. 7). When sufficient piecewise regularity is assumed, we have by Ref. 46

$$\|\alpha \text{grad } \mathbf{u}\|^2 \approx \|\alpha \text{curl } \mathbf{u}\|^2 + \|\alpha \text{div } \mathbf{u}\|^2.$$

Hence, if $\mathbf{u} = \text{curl } \mathbf{w} \in (\ker(\text{curl}))^\perp$, for some suitable \mathbf{w} , such that $\text{div } \mathbf{u} = 0$, then

$$(\alpha \text{curl } \mathbf{u}, \text{curl } \mathbf{u}) + (\beta \mathbf{u}, \mathbf{u}) \approx (\alpha \text{grad } \mathbf{u}, \text{grad } \mathbf{u}) + (\beta \mathbf{u}, \mathbf{u}),$$

which corresponds to the following operator:

$$\mathbf{B}^{\text{grad}} \mathbf{u} := -\text{div}(\alpha \nabla \mathbf{u}) + \beta \mathbf{u} \quad (6.30)$$

that will be assembled as an auxiliary matrix and can be solved by an AMG solver for the vector H^1 -interface problem. On the other hand, if $\mathbf{u}, \mathbf{v} \in \ker(\text{curl})$, i.e., $\mathbf{u} = \nabla p$ and $\mathbf{v} = \nabla q$, for some suitable p, q , then

$$(\alpha \text{curl } \mathbf{u}, \text{curl } \mathbf{v}) + (\beta \mathbf{u}, \mathbf{v}) = (\beta \nabla p, \nabla q),$$

thus we can formulate the matrix problem for the gradient part of the solution by $B^{\text{grad}} = G^\top A G$, which corresponds to the following operator:

$$B^{\text{grad}} p := -\text{div}(\beta \nabla p) \quad (6.31)$$

that can be again solved efficiently by an AMG solver for the H^1 -interface problem.

Next, we present a block diagonal smoother (preconditioner). A block matrix is formed by the edge DoFs in the neighborhood expanding from the interface. We begin with the collection of the DoFs that is near the interface:

$$\mathcal{D}_1 = \{e \in \mathcal{E}_h : \exists K \in \mathcal{T}_h^i \text{ such that } e \in \mathcal{E}_K\}. \quad (6.32)$$

Then, starting from \mathcal{D}_1 we iteratively define

$$\mathcal{D}_l = \{e \in \mathcal{E}_h : e \text{ has at least one node belonging to the edges of } \mathcal{D}_{l-1}\}. \quad (6.33)$$

Let A_l be the matrix of the entries in A associated with the DoFs in \mathcal{D}_l . Then, we rewrite (6.28) into

$$A\mathbf{u} := \begin{pmatrix} A_N & A_{Nl} \\ A_{lN} & A_l \end{pmatrix} \begin{pmatrix} \mathbf{u}_N \\ \mathbf{u}_l \end{pmatrix} = \begin{pmatrix} \mathbf{f}_N \\ \mathbf{f}_l \end{pmatrix} \quad (6.34)$$

Here, the key is to solve the A_l block by a direct solver, which is indeed the price to be paid by the proposed method. However, since the size of A_l , i.e., $\#\mathcal{D}_l$, is in the order of $\mathcal{O}(\#\text{total DoF}^{2/3})$ for reasonably small l , this direct solver is generally efficient. The expanding width can reduce the number of iterations required for the resulting solver, see Table 3 for the comparison. Meanwhile, our numerical experience suggests that the increased cost is negligible for small l 's, e.g., the expanding width $l = 1$ or 2 is enough. It is almost equivalent to directly solving a 2D linear system which can be efficiently handled by “backslash” (`mldivide`) in Matlab. Furthermore, since the direct solver will be called multiple times, we opt

to store the LU -factorization of A_l in the inner iteration (preconditioning) to be more efficient. The residual equation of the A_N block can be efficiently solved using a block or point-wise Gauss-Seidel smoother. At last, we summarize the algorithm in the following for a fixed l , and denote $A_I := A_l$.

Algorithm 1 An HX preconditioned CG

Require: $\mathbf{u}^{(0)}$, \mathbf{tol} , M , l , block form of A .
Ensure: \mathbf{u}^{MG} .

```

1:  $k = 0$ .
2:  $\mathbf{r}_I \leftarrow \mathbf{f}_I - A_I \mathbf{u}^{(0)}$ 
3: while True do
4:    $\mathbf{r}_I \leftarrow \text{LUSolve}(A_I, \mathbf{r}_I)$ 
5:    $\mathbf{r}_N \leftarrow \text{Smoothen}(A_N, \mathbf{r}_N)$ 
6:    $\mathbf{r} \leftarrow [\mathbf{r}_N, \mathbf{r}_I]$ 
7:    $\mathbf{r} \leftarrow \mathbf{r} + \Pi(\text{AuxSolve}(A^{\text{aux}}, \Pi^\top \mathbf{r}))$ 
8:    $\mathbf{r}_c \leftarrow G(\text{AuxSolve}(A, G^\top \mathbf{r}))$ 
9:    $\mathbf{r} \leftarrow \mathbf{r} + \mathbf{r}_c$ 
10:   $\mathbf{u}^{\text{MG}} \leftarrow \text{CG}(A, \mathbf{r})$ 
11:  if  $k > M$  or  $\text{norm}(r) < \mathbf{tol}$  then
12:    Break
13:  end if
14:   $k \leftarrow k + 1$ 
15: end while

```

7. Numerical Examples

In this section, we present a class of numerical examples to validate the aforementioned advantages of the proposed method. The background unfitted mesh is generated by cutting Ω into N^3 cubes and each cube is then cut into several tetrahedra with the mesh size be $h = 1/N$.

7.1. The H^1 -interface problem

We first consider the H^1 -interface problem given by (1.2) for a spheric interface shown in Figure 7.1 on the domain $\Omega = (-1, 1)^3$. The exact solution is constructed as

$$u(\mathbf{x}) = \begin{cases} \exp((\|\mathbf{x}\|^2 - r^2)/\beta^-), & \text{if } \mathbf{x} \in \Omega^-, \\ \sin((\|\mathbf{x}\|^2 - r^2)/\beta^+) + 1, & \text{if } \mathbf{x} \in \Omega^+, \end{cases} \quad (7.1)$$

where the source term of (1.2) as well as the boundary conditions are computed accordingly. The numerical experiment is carried on the meshes of $N = 10, 20, 30, \dots, 160$. We first report the CPU time to compute the projections of IVE

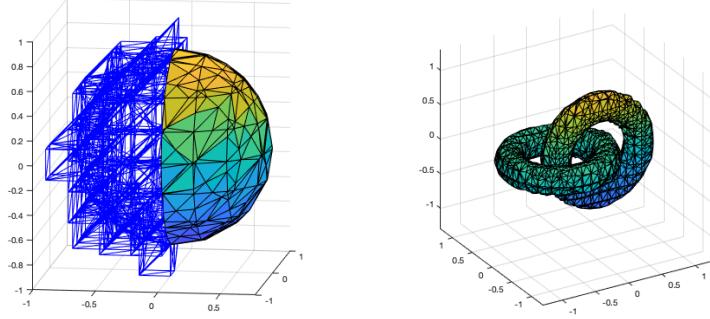


Fig. 7.1: Plots of interface and triangulation: sphere (left) and two twisted tori (right). For the spherical interface, the boundary triangulation of interface element is plotted in blue on the left half of the sphere.

functions and the related matrix assembling in Table 2 for the meshes $N = 60$ to 160. Note that at the finest level, there are approximately 153600 interface elements, and based on the proposed data structure, computing the IVE projections is highly efficient. As for the matrix assembling, we observe that the majority of time is devoted to the stabilization term, and we believe this is due to a larger number of triangular faces from the boundary triangulation. Certainly, these computations are highly parallelizable. In addition, we show the numerical errors for $\beta^- = 1$, $\beta^+ = 10$ and $\beta^- = 1$, $\beta^+ = 100$ in the left two plots of Figure 7.2. Due to the geometric errors caused by coarse meshes, the convergence orders indicated on the graph are computed by incorporating only the errors from $N = 60$ to 160, but it clearly shows the asymptotic optimal convergence. Remarkably, the optimal convergence is even achieved for the L^∞ norm which is a demanding property for interface problems.

Total # DoFs	1367631	1771561	2248091	2803221	3442951	4173281
Interface # DoFs	160926	191322	224682	259818	298866	339774
Time(s) for projection	2.96	4.00	4.88	37.92	5.65	7.13
Time(s) for matrix assembling	14.21	18.14	21.14	30.47	37.58	42.61

Table 2: CPU time for computing the projections of IFE functions and the generation of stiffness matrices including the stabilization terms.

The second example concerns a more complicated interface shape that has two tori twisted with each other, see the right plot in Figure 7.1. The level-set functions of the two tori are $\phi_1 = (((x_1 + 0.3)^2 + x_2^2)^{1/2} - 0.2)^2 + x_3^2 - (\pi/5)^2$ and $\phi_2 = (((x_1 - 0.3)^2 + x_2^2)^{1/2} - 0.2)^2 + x_3^2 - (\pi/5)^2$, and then the level-set function of this interface is given by $\phi(\mathbf{x}) = \min(\phi_1(\mathbf{x}), \phi_2(\mathbf{x}))$. The domain inside the two tori is $\Omega^- = \{\mathbf{x} : \phi(\mathbf{x}) < 0\}$ and the outside one is $\Omega^+ = \{\mathbf{x} : \phi(\mathbf{x}) > 0\}$. The exact

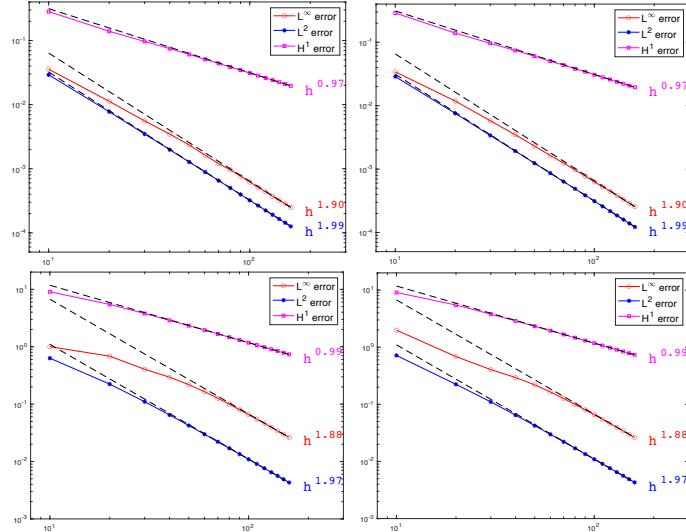


Fig. 7.2: Numerical errors and convergence order for the H^1 interface problem. The top two plots are for the spherical interface: $(\beta^-, \beta^+) = (1, 10)$ and $(\beta^-, \beta^+) = (1, 100)$, and bottom two plots are for the toroidal interface: $(\beta^-, \beta^+) = (1, 10)$ and $(\beta^-, \beta^+) = (1, 100)$. The black dashed lines indicate the expected $\mathcal{O}(h^2)$ convergence for the L^∞ and L^2 errors and $\mathcal{O}(h)$ for the H^1 errors.

solution is given by

$$u(\mathbf{x}) = \begin{cases} 1, & \text{if } \mathbf{x} \in \Omega^-, \\ \cos(\phi_1(\mathbf{x})\phi_2(\mathbf{x})), & \text{if } \mathbf{x} \in \Omega^+. \end{cases} \quad (7.2)$$

In this case, the computational domain is $\Omega = (-1.3, 1.3)^3$. The numerical solutions and errors are reported in the right two plots of Figure 7.2. As the interface has much larger curvature which requires the finer mesh to resolve. The convergence orders are estimated from the mesh size $N = 60$ to 160 which indicate the optimal convergence even for the L^∞ errors.

7.2. The $\mathbf{H}(\mathbf{curl})$ interface problem

Now, let us consider the $\mathbf{H}(\mathbf{curl})$ interface problem. It is known that solving the linear system from the Maxwell's equations is much more challenging. So here we first test the fast solver developed in Section 6.5 for an extreme case that each interface element has small-cut subelements. For this purpose, we consider the domain $\Omega = (-1, 1)^3$ with a flat interface $x_1 = 5 \times 10^{-2-r}$. Fix the mesh size as $N = 20$ and the parameters as $(\alpha^-, \alpha^+) = (1, 10)$ and $(\beta^-, \beta^+) = (1, 10)$. If $r = 0$, i.e., $x_1 = 0.05$, the interface plane cuts all the interface elements exactly through the center and thus, each subelement has regular shape. In computation, we let

$r = 1, 2, 3, 4$, i.e., the subelements on the left-side of the interface will become small accordingly. We report the condition numbers, the number of iterations and CPU time in Table 3. We can see that small-cut interface elements can indeed make the conditioning worse, which may significantly increase the iteration numbers, see the results for $l = 0$. However, the effect of small-cut interface elements can be successfully eliminated by the proposed block diagonal smoother A_l in (6.34). For this extreme case, $l = 2$ seems sufficient to make the convergence completely independent of small subelements, but our numerical experience suggests that $l = 1$ is good enough in general.

	Interface location r	0	1	2	3	4
	Condition numbers	6.4×10^5	1.7×10^7	1.7×10^9	1.7×10^{11}	1.7×10^{13}
$l = 0$	# iteration	44	53	107	327	842
	Time(s)	12	16	27	84	220
$l = 1$	# iteration	43	44	43	73	91
	Time(s)	11	11	11	21	24
$l = 2$	# iteration	43	44	43	43	42
	Time(s)	11	12	12	11	12

Table 3: Condition numbers of the $\mathbf{H}(\text{curl})$ linear system with various interface location, and the related CPU time (in seconds) and # iterations for the expanding width $l = 0, 1, 2$, where $l = 0$ means no block matrix used.

Next, we consider the spherical interface and slightly modify the benchmark example from Ref. 57 of which the analytical solution is given by

$$\mathbf{u} = \begin{cases} \frac{1}{\beta^-} \mathbf{x} + \frac{1}{\alpha^-} n_1 R_1(\mathbf{x}) [(x_2 - x_3), (x_3 - x_1), (x_1 - x_2)]^\top & \text{in } \Omega^-, \\ \frac{1}{\beta^+} \mathbf{x} + \frac{1}{\alpha^+} n_2 R_1(\mathbf{x}) R_2(\mathbf{x}) [(x_2 - x_3), (x_3 - x_1), (x_1 - x_2)]^\top & \text{in } \Omega^+, \end{cases} \quad (7.3)$$

where $\mathbf{x} = [x_1, x_2, x_3]^\top$ and $R_1(\mathbf{x}) = r_1^2 - \|\mathbf{x}\|^2$, $R_2(\mathbf{x}) = r_2^2 - \|\mathbf{x}\|^2$. The numerical experiment is carried on the meshes of $N = 10, 20, 30, \dots, 80$. In particular, the computational time and number of iterations are presented in Table 4. From the table, we can conclude that both the block-diagonal smoother and the HX preconditioner are important for reducing the iteration number for convergence. The results also show that the direct solver at each iteration does not cost significant computational time compared with the total cost of the iterative solver. Next, we report the numerical errors in both the L^2 and $\mathbf{H}(\text{curl})$ norms in the left two plots of Figure 7.3, and the estimated convergence orders are also indicated in the plots which clearly demonstrate the optimality.

	Total # DoFs	80554	244424	547074	1028452	1734626	2703384	3980338
BD-PCG	# iteration	962	1465	1932	2385	2828	3238	3661
	Time(s)	42.72	166.46	427.70	905.39	1693.95	2871.83	4542.63
BD-HX	# iteration	144	142	146	140	148	142	145
$l = 0$	Time(s)	38.89	69.09	117.97	214.89	401.81	533.28	808.19
BD-HX	# iteration	75	76	81	77	80	83	90
$l = 1$	Time(s)	22.16	41.14	70.12	132.54	241.93	321.93	529.66

Table 4: CPU time and number of iterations for solving the $\mathbf{H}(\text{curl})$ linear system with the spherical interface and $(\alpha^-, \beta^-) = (1, 1)$ and $(\alpha^+, \beta^+) = (100, 200)$: block-diagonal HX (BD-HX) with $l = 0$ and 1 and the simple block-diagonal PCG (BD-PCG). The CPU time with respect to DoFs are approximately $\mathcal{O}((\# \text{DoF})^{1.19})$, $\mathcal{O}((\# \text{DoF})^{0.80})$, $\mathcal{O}((\# \text{DoF})^{0.82})$ for BD-PCG, BD-HX($l = 0$) and BD-HX($l = 1$), respectively.

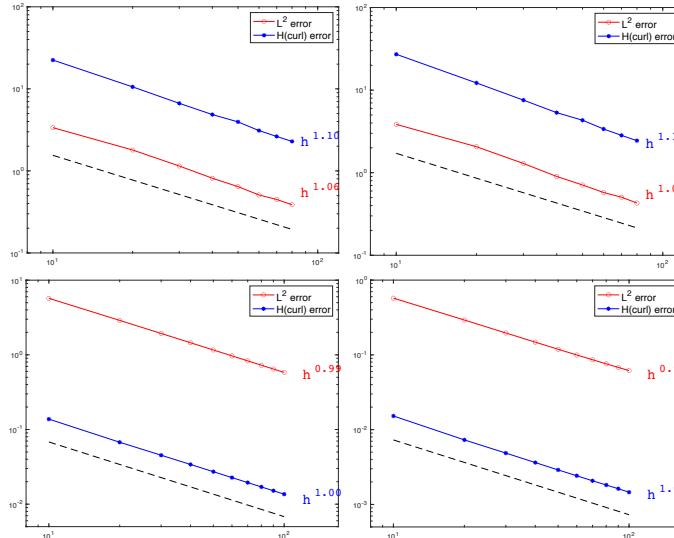


Fig. 7.3: Numerical errors and convergence order for the $\mathbf{H}(\text{curl})$ interface problem by the IVE method. Top two: the first example. Bottom two: the second example.

IFE spaces can be also used in a dG-type scheme, i.e., penalties are used to handle the discontinuities across faces. This scheme works very well for H^1 interface problems ^{56,58}, but results in only suboptimal convergence for $\mathbf{H}(\text{curl})$ problems. For the $\mathbf{H}(\text{curl})$ case, let us recall the scheme below. Let \mathcal{F}_h^i be the collection of the interface faces. Let \mathcal{S}_h be a space containing IFE functions which may not be continuous on faces in \mathcal{F}_h^i . Note that IFE functions can main tangential continuity

on non-interface faces, and thus penalties are only needed on interface faces. Then, the scheme is to find $\mathbf{u}_h \in \mathcal{S}_h$ such that

$$\tilde{a}_h(\mathbf{u}_h, \mathbf{v}_h) = \int_{\Omega} \mathbf{f} \cdot \mathbf{v}_h \, ds \quad (7.4)$$

where

$$\begin{aligned} \tilde{a}_h(\mathbf{u}_h, \mathbf{v}_h) := & (\alpha_h \operatorname{curl} \mathbf{u}_h, \operatorname{curl} \mathbf{v}_h)_{\Omega} + (\beta_h \mathbf{u}_h, \mathbf{v}_h)_{\Omega} \\ & - \sum_{F \in \mathcal{F}_h^i} \int_F \{\operatorname{curl} \mathbf{u}_h\} \cdot [\mathbf{v}_h \times \mathbf{n}] \, ds \\ & - \sum_{F \in \mathcal{F}_h^i} \int_F \{\operatorname{curl} \mathbf{v}_h\} \cdot [\mathbf{u}_h \times \mathbf{n}] \, ds \\ & + \gamma h^{-1} \sum_{F \in \mathcal{F}_h^i} \int_F [\mathbf{u}_h \times \mathbf{n}] [\mathbf{v}_h \times \mathbf{n}] \, ds, \quad \forall \mathbf{v}_h \in \mathcal{S}_h, \end{aligned} \quad (7.5)$$

where γ is a stabilization parameter which should be large enough and generally depends on α . We present the numerical results in Figure 7.4. For the semi- $\mathbf{H}(\operatorname{curl})$ norm, we can clearly observe the sub-optimal convergence. The convergence under the L^2 norm deteriorates a little as the mesh becomes finer. In some other setting, we can also observe much worse behavior for the L^2 norm.

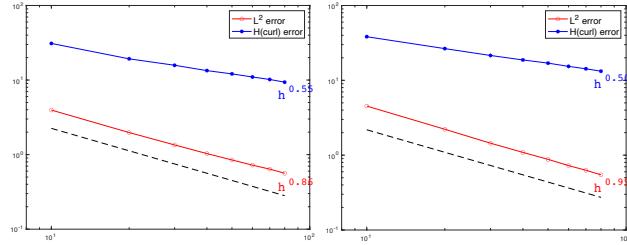


Fig. 7.4: Numerical errors and convergence order for the $\mathbf{H}(\operatorname{curl})$ interface problem by the penalty-type IFE method. Left two: the first example. Right two: the second example.

In the second example, we also consider the twisted tori in the right plot of Figure 7.1 on the domain $\Omega = (-1.3, 1.3)^3$. To construct a function that satisfies the corresponding jump condition on the torus surface, we let $f(\mathbf{x}) = \phi_1(\mathbf{x})\phi_2(\mathbf{x})((x_1 + 0.3)^2 + x_2^2)((x_1 - 0.3)^2 + x_3^2)$. Then, the exact solution is then defined as

$$\mathbf{u} = \frac{1}{\beta} \nabla f(\mathbf{x}) + \frac{1}{\alpha} \cos(f(\mathbf{x})) \mathbf{v}_0, \quad \text{with } \mathbf{v}_0 = [0, 0, 1]^{\top}, \quad (7.6)$$

where the boundary conditions and the source term are computed accordingly. The numerical results are reported in the right two plots of Figure 7.3 which also shows

the clear optimal convergence rate. These results demonstrate that the IVE method works well for complex surfaces.

At last, we note that the penalty-type method cannot achieve optimal convergence numerically including both the IFE method ⁵⁷ and the interface-penalty method ^{32,31}. Therefore, we believe the present method has distinguishing advantages in computational electromagnetism.

Appendix A IFE Spaces on Complicated Geometry

Here, we describe the IFE spaces for complicated interface element geometry, i.e., Γ^K may have multiple components. Let us assume that Γ^K consists of the multiple components $\Gamma^{K,m}$, $m = 1, 2, \dots, M-1$, each of which is a simply-connected smooth surface. As Γ is supposed not to intersect itself, $\Gamma^{K,m}$'s then do not intersect with one another. Without loss of generality, we assume the subelement containing A_1 is K_1 and $\Gamma^{K,1} = \partial K_1 \setminus \partial K$. Then, K_m is the subelement bounded by ∂K , $\Gamma^{K,m}$ and $\Gamma^{K,m+1}$, $m = 2, \dots, M-1$, and the remaining one is denoted by K_M . We show a 2D illustration of the geometry by the right plot in Figure 2.1. The parameters associated with the subelement K_m are denoted as α^m and β^m , which should take the values of α^- and α^+ alternatively.

For each $\Gamma^{K,m}$, we let $\Gamma_h^{K,m}$ be its planar approximation. Similarly, define the subelement containing A_1 as $K_{h,1}$, and the others, i.e., $K_{h,2}, \dots, K_{h,M}$, are defined in a similar manner as their counterparts K_1, \dots, K_M . Note that each of $K_{h,1}, \dots, K_{h,M}$ is a polyhedron. Let α_h and β_h be the piecewise constant functions defined on these polyhedral subelements, and denote $\alpha_{h,m} = \alpha_h|_{K_{h,m}}$ and $\beta_{h,m} = \beta_h|_{K_{h,m}}$, $m = 1, \dots, M$.

Similar to (4.6), we are able to derive explicit formulas for the functions in the spaces (4.3). For each linear interface component Γ_h^K , we further let $\bar{\mathbf{t}}_m^1$ and $\bar{\mathbf{t}}_m^2$ be the two orthogonal tangential unit vectors to $\Gamma_h^{K,m}$, and denote the matrix $T_m = [\bar{\mathbf{n}}_m, \bar{\mathbf{t}}_m^1, \bar{\mathbf{t}}_m^2]$. Then define the transformation matrices:

$$M_{K,m}^{f,c_h} = T \begin{bmatrix} 1 & 0 & 0 \\ 0 & \tilde{c}_m & 0 \\ 0 & 0 & \tilde{c}_m \end{bmatrix} (T_m)^\top \quad \text{and} \quad M_{K,m}^{e,c_h} = T \begin{bmatrix} \tilde{c}_m & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} (T_m)^\top, \quad (\text{A.1})$$

where $\tilde{c}_m = c_{h,m}/c_{h,m+1}$, with $m = 1, 2, \dots, M-1$, and define the spaces $\mathbf{P}_h^e(c_h; K)$ and $\mathbf{P}_h^f(c_h; K)$ as

$$\begin{aligned} \mathbf{P}_h^e(c_h; K) = \{ \mathbf{c} : \mathbf{c}_m = \mathbf{c}|_{K_{h,m}} \in \mathcal{P}_0(K_{h,m}), \quad m = 1, \dots, M, \\ \mathbf{c}_{m+1} = M_{K,m}^{e,c_h} \mathbf{c}_m, \quad m = 1, \dots, M-1 \}, \end{aligned} \quad (\text{A.2a})$$

$$\begin{aligned} \mathbf{P}_h^f(c_h; K) = \{ \mathbf{c} : \mathbf{c}_m = \mathbf{c}|_{K_{h,m}} \in \mathcal{P}_0(K_{h,m}), \quad m = 1, \dots, M, \\ \mathbf{c}_{m+1} = M_{K,m}^{f,c_h} \mathbf{c}_m, \quad m = 1, \dots, M-1 \}. \end{aligned} \quad (\text{A.2b})$$

Again, the constant vectors at different cut regions are related by the jump conditions and thus the dimension of both $\mathbf{P}_h^e(c_h; K)$ and $\mathbf{P}_h^f(c_h; K)$ is also 3. In this case,

the formulas of IFE functions are slightly more complicated which are presented in the following lemma.

Lemma A.1. *Let $\mathbf{x}_{K,m}$ be an arbitrary point at $\Gamma_h^{K,m}$. Let $\mathbf{a} = \mathbf{a}_m$ and $\mathbf{b} = \mathbf{b}_m$ be two arbitrary vectors in $\mathbf{P}_h^f(a_h; K)$ and $\mathbf{P}_h^e(b_h; K)$, and let c be an arbitrary constant. Then, the formulas for the functions in $S_h^n(b_h; K)$, $\mathbf{S}_h^e(a_h, b_h; K)$ and $\mathbf{S}_h^f(a_h; K)$, respectively, are*

$$v_h^n = \begin{cases} \mathbf{b}_m \cdot (\mathbf{x} - \mathbf{x}_{K,1}) + c & \text{in } K_{h,m}, \ m = 1, 2, \\ \mathbf{b}_m \cdot (\mathbf{x} - \mathbf{x}_{K,m-1}) + c + \sum_{l=2}^{m-1} \mathbf{b}_l \cdot (\mathbf{x}_{K,l} - \mathbf{x}_{K,l-1}), & \text{in } K_{h,m}, \ m \geq 3, \end{cases} \quad (\text{A.3})$$

$$\mathbf{v}_h^e = \begin{cases} \mathbf{a}_m \times (\mathbf{x} - \mathbf{x}_{K,1}) + \mathbf{b}_m & \text{in } K_{h,m}, \ m = 1, 2, \\ \mathbf{a}_m \times (\mathbf{x} - \mathbf{x}_{K,m-1}) + \mathbf{b}_m + \boldsymbol{\xi}_m, & \text{in } K_{h,m}, \ m \geq 3, \end{cases} \quad (\text{A.4})$$

with

$$\boldsymbol{\xi}_m := \sum_{l=2}^{m-1} (M_{K,m-1}^{e,b_h} \cdots M_{K,l}^{e,b_h}) [\mathbf{a}_l \times (\mathbf{x}_{K,l} - \mathbf{x}_{K,l-1})],$$

and

$$\mathbf{v}_h^f = \begin{cases} c(\mathbf{x} - \mathbf{x}_{K,1}) + \mathbf{a}_m & \text{in } K_{h,m}, \ m = 1, 2, \\ c(\mathbf{x} - \mathbf{x}_{K,m-1}) + \mathbf{a}_m + \boldsymbol{\eta}_m, & \text{in } K_{h,m}, \ m \geq 3, \end{cases} \quad (\text{A.5})$$

with

$$\begin{aligned} \boldsymbol{\eta}_m &:= c \sum_{l=2}^{m-1} (M_{K,m-1}^{f,a_h} \cdots M_{K,l}^{f,a_h}) (\mathbf{x}_{K,l} - \mathbf{x}_{K,l-1}) \\ &= c \sum_{l=2}^{m-1} \left(\prod_{n=m-1}^l M_{K,n}^{f,a_h} \right) (\mathbf{x}_{K,l} - \mathbf{x}_{K,l-1}). \end{aligned}$$

The formed IFE spaces also have the dimension 4, 6, and 4 for the H^1 , $\mathbf{H}(\text{curl})$ and $\mathbf{H}(\text{div})$ cases, respectively.

Proof. One can directly verify that these piecewisely-defined functions satisfy the corresponding jump conditions shown in Table 1 but on each $\Gamma_h^{K,m}$. The dimension can be simply counted by the number of free variables of \mathbf{a} , \mathbf{b} and c in the formulas above. \square

Note that in Lemma A.1, the points $\{\mathbf{x}_{K,m}\}_{m=1}^{M-1}$ should be chosen and fixed.

Acknowledgment

This work was supported in part by the National Science Foundation under grants DMS-1913080, DMS-2012465, and DMS-2136075. We would like to thank the anonymous reviewers for the suggestions on improving this article. We would like to thank Dr. Xuehai Huang (Shanghai University of Finance and Economics) for his numerous advices on revising the article.

References

1. G. Acosta and R. G. Durán, The maximum angle condition for mixed and nonconforming elements: Application to the Stokes equations, *SIAM J. Numer. Anal.* **37** (1999) 18–36. [3](#)
2. S. Adjerid, I. Babuška, R. Guo and T. Lin, An enriched immersed finite element method for interface problems with nonhomogeneous jump conditions, *Comput. Methods Appl. Mech. Engrg.* **404** (2023) 115770. [6](#)
3. S. Adjerid, N. Chaabane and T. Lin, An immersed discontinuous finite element method for Stokes interface problems, *Comput. Methods Appl. Mech. Engrg.* **293** (2015) 170–190, in press. [4](#)
4. B. Ahmad, A. Alsaedi, F. Brezzi, L. Marini and A. Russo, Equivalent projectors for virtual element methods, *Comput. Math. Appl.* **66** (2013) 376–391.
5. C. Amrouche, C. Bernardi, M. Dauge and V. Girault, Vector potentials in three-dimensional non-smooth domains, *Math. Meth. Appl. Sci.* **21** (1998) 823–864.
6. D. N. Arnold, R. S. Falk and R. Winther, Preconditioning in $H(\text{div})$ and applications, *Math. Comp.* **66** (1997) 957–984. [3](#)
7. D. N. Arnold, R. S. Falk and R. Winther, Multigrid in $H(\text{div})$ and $H(\text{curl})$, *Numer. Math.* **85** (2000) 197–217.
8. I. Babuška, The finite element method for elliptic equations with discontinuous coefficients, *Computing (Arch. Elektron. Rechnen)* **5** (1970) 207–213. [15](#)
9. I. Babuška and A. K. Aziz, On the angle condition in the finite element method, *SIAM J. Numer. Anal.* **13** (1976) 214–226. [3](#)
10. I. Babuška, G. Caloz and J. E. Osborn, Special finite element methods for a class of second order elliptic problems with rough coefficients, *SIAM J. Numer. Anal.* **31** (1994) 945–981. [6](#)
11. I. Babuška and J. E. Osborn, Generalized finite element methods: their performance and their relation to mixed methods, *SIAM J. Numer. Anal.* **20** (1983) 510–536. [6](#)
12. M. Barker, S. Cao and A. Stern, A nonconforming primal hybrid finite element method for the two-dimensional vector Laplacian, *arXiv preprint arXiv:2206.10567*. [5](#)
13. L. Beirão da Veiga, F. Brezzi, F. Dassi, L. Marini and A. Russo, Lowest order virtual element approximation of magnetostatic problems, *Comput. Methods Appl. Mech. Engrg.* **332** (2018) 343–362. [5, 25](#)
14. L. Beirão da Veiga, F. Brezzi, A. Cangiani, G. Manzini, L. Marini and A. Russo, Basic principles of virtual element methods, *Math. Models Methods Appl. Sci.* **23** (2013) 199–214. [4, 6, 25, 28, 32](#)
15. L. Beirão da Veiga, F. Brezzi, F. Dassi, L. Marini and A. Russo, Virtual Element approximation of 2D magnetostatic problems, *Comput. Methods Appl. Mech. Engrg.* **327** (2017) 173–195. [6, 25](#)
16. L. Beirão da Veiga, F. Brezzi, L. D. Marini and A. Russo, $H(\text{div})$ and $H(\text{curl})$ -conforming virtual element methods, *Numer. Math.* **133** (2016) 303–332. [6, 25](#)
17. L. Beirão da Veiga, F. Dassi, G. Manzini and L. Mascotto, Virtual elements for Maxwell's equations, *Comput. Math. Appl.* (2021) 82–99. [5, 6, 25](#)
18. L. Beirão da Veiga, C. Lovadina and A. Russo, Stability analysis for the virtual element method, *Math. Models Methods Appl. Sci.* **27** (2017) 2557–2594.
19. L. Beirão da Veiga and L. Mascotto, Interpolation and stability properties of low order face and edge virtual element spaces, *IMA J. Numer. Anal.* **drac008**. [5, 32](#)
20. F. Ben Belgacem, A. Buffa and Y. Maday, The mortar finite element method for 3D Maxwell equations: First results, *SIAM J. Numer. Anal.* **39** (2001) 880–901.
21. S. Borda, E. Burman, M. Larson and M. O. (Editors), *Geometrically Unfitted Finite Element Methods and Applications, Proceedings of the UCL Workshop 2016*, volume

121 of *Lecture Notes in Computational Science and Engineering* (Springer, 2017). 4

22. J. R. Brauer, J. J. Ruehl, M. A. Juds, M. J. V. Heiden and A. A. Arkadan, Dynamic stress in magnetic actuator computed by coupled structural and electromagnetic finite elements, *IEEE Trans. Magn.* **32** (1996) 1046 – 1049. 3
23. S. Brenner and L.-Y. Sung, Virtual element methods on meshes with small edges or faces, *Math. Models Methods Appl. Sci.* **28** (2018) 1291–1336. 28, 32
24. S. C. Brenner, J. Cui, F. Li and L. Y. Sung, A nonconforming finite element method for a two-dimensional curl–curl and grad–div problem, *Numer. Math.* **109** (2008) 509–533.
25. A. Buffa, M. Costabel and M. Dauge, Algebraic convergence for anisotropic edge elements in polyhedral domains, *Numer. Math.* **101** (2005) 29–65. 3
26. E. Burman, S. Claus, P. Hansbo, M. G. Larson and A. Massing, CutFEM: Discretizing geometry and partial differential equations, *Internat. J. Numer. Methods Engrg.* **104** (2015) 472–501. 4, 32
27. S. Cao and L. Chen, Anisotropic error estimates of the linear virtual element method on polygonal meshes, *SIAM J. Math. Anal.* **56** (2018) 2913–2939. 33
28. S. Cao and L. Chen, Anisotropic error estimates of the linear nonconforming virtual element methods, *SIAM J. Numer. Anal.* **57** (2019) 1058–1081. 28
29. S. Cao, L. Chen and R. Guo, A virtual finite element method for two dimensional Maxwell interface problems with a background unfitted mesh, *Math. Models Methods Appl. Sci.* (2021) 2907–2936. 5, 33
30. S. Cao, L. Chen, R. Guo and F. Lin, Immersed virtual element methods for elliptic interface problems, *J. Sci. Comput.* **93** (2022) 1–41. 6, 8, 33
31. R. Casagrande, R. Hiptmair and J. Ostrowski, An a priori error estimate for interior penalty discretizations of the Curl-Curl operator on non-conforming meshes, *J. Math. Ind.* **6** (2016) 4. 5, 43
32. R. Casagrande, C. Winkelmann, R. Hiptmair and J. Ostrowski, DG Treatment of Non-conforming Interfaces in 3D Curl-Curl Problems, in *Scientific Computing in Electrical Engineering* (Springer International Publishing, Cham, 2016), pp. 53–61. 5, 43
33. L. Chen, *iFEM*: an integrated finite element methods package in MATLAB, Technical report, University of California at Irvine, 2009. 35
34. L. Chen, R. Guo and J. Zou, A family of immersed finite element spaces and applications to three dimensional $H(\text{curl})$ interface problems, *arXiv preprint arXiv:2205.14127*.
35. L. Chen, M. Holst and J. Xu, The finite element approximation of the nonlinear Poisson–Boltzmann equation, *SIAM J. Numer. Anal.* **45** (2007) 2298–2320. 2
36. L. Chen and X. Huang, Finite element de Rham and Stokes complexes in three dimensions, *arXiv preprint arXiv:2206.09525*.
37. L. Chen, H. Wei and M. Wen, An interface-fitted mesh generator and virtual element methods for elliptic interface problems, *J. Comput. Phys.* **334** (2017) 327–348.
38. L. Chen, Y. Wu, L. Zhong and J. Zhou, Multigrid preconditioners for mixed finite element methods of the vector Laplacian, *J. Sci. Comput.* **77** (2018) 101–128. 7
39. Z. Chen, Q. Du and J. Zou, Finite element methods with matching and nonmatching meshes for Maxwell equations with discontinuous coefficients, *SIAM J. Numer. Anal.* **37** (2000) 1542–1570.
40. Z. Chen, Z. Wu and Y. Xiao, An adaptive immersed finite element method with arbitrary Lagrangian-Eulerian scheme for parabolic equations in time variable domains, *Int. J. Numer. Anal. Mod.* (2015) 567–591. 4
41. Z. Chen, Y. Xiao and L. Zhang, The adaptive immersed interface finite element method for elliptic and Maxwell interface problems, *J. Comput. Phys.* **228** (2009)

5000–5019. 3

42. Z. Chen and J. Zou, Finite element methods and their convergence for elliptic and parabolic interface problems, *Numer. Math.* **79** (1998) 175–202.
43. C.-C. Chu, I. G. Graham and T.-Y. Hou, A new multiscale finite element method for high-contrast elliptic interface problems, *Math. Comp.* **79** (2010) 1915–1955. 4, 6, 25
44. D. Colton and R. Kress, *Inverse Acoustic and Electromagnetic Scattering Theory* (Springer, NY, 1996). 3
45. M. Costabel, A remark on the regularity of solutions of maxwell's equations on lipschitz domains, *Mathematical Methods in the Applied Sciences* **12** (1990) 365–368.
46. M. Costabel, M. Dauge and S. Nicaise, Singularities of Maxwell interface problems, *ESAIM: Mathematical Modelling and Numerical Analysis* **33** (1999) 627–649.
47. F. Dassi, A. Fumagalli, D. Losapio, S. Scialò, A. Scotti and G. Vacca, The mixed virtual element method on curved edges in two dimensions, *Comput. Methods Appl. Mech. Engrg.* **386** (2021) 114098. 16
48. E. M. Dede, J. Lee and T. Nomura, *Multiphysics Simulation: Electromechanical System Applications and Optimization* (Springer, 2014). 3
49. P. Fernandes and G. Gilardi, Magnetostatic and electrostatic problems in inhomogeneous anisotropic media with irregular boundary and mixed boundary conditions, *Math. Models Methods Appl. Sci.* **7** (1997) 957–991.
50. V. Girault and P.-A. Raviart, *Finite Element Methods for Navier-Stokes Equations: Theory and Algorithms* (Springer Publishing Company, 2011), 1st edition.
51. Y. Gong, B. Li and Z. Li, Immersed-interface finite-element methods for elliptic interface problems with nonhomogeneous jump conditions, *SIAM J. Numer. Anal.* **46** (2008) 472–495. 4
52. Y. Gong and Z. Li, Immersed interface finite element methods for elasticity interface problems with non-homogeneous jump conditions, *Numer. Math. Theory Methods Appl.* **3** (2010) 23–39. 4
53. W. J. Gordon, Blending-function methods of bivariate and multivariate interpolation and approximation, *SIAM J. Numer. Anal.* **8** (1971) 158–177. 16
54. R. Guo, On the maximum angle conditions for polyhedra with virtual element methods, *arXiv preprint arXiv:2212.07241*.
55. R. Guo and T. Lin, A group of immersed finite element spaces for elliptic interface problems, *IMA J. Numer. Anal.* **39** (2017) 482–511. 4
56. R. Guo and T. Lin, An immersed finite element method for elliptic interface problems in three dimensions, *J. Comput. Phys.* **414** (2020) 109478. 15, 41
57. R. Guo, Y. Lin and J. Zou, Solving two dimensional $H(\mathbf{curl})$ -elliptic interface systems with optimal convergence on unfitted meshes, *European J. Appl. Math.* (in press). 5, 43
58. R. Guo and X. Zhang, Solving three-dimensional interface problems with immersed finite elements: A-priori error analysis, *J. Comput. Phys.* **441** (2020) 110445. 15, 41
59. R. Hiptmair, J. LI and J. Zou, Convergence analysis of finite element methods for $H(\text{div}; \Omega)$ -elliptic interface problems, *J. Numer. Math.* **18** (2010) 187–218.
60. R. Hiptmair and J. Xu, Nodal auxiliary space preconditioning in $H(\mathbf{curl})$ and $H(\text{div})$ spaces, *SIAM J. Numer. Anal.* **45** (2007) 2483–2509. 7
61. D. Holder, *Electrical impedance tomography: methods, history, and applications* (Institute of Physics Pub, 2005). 2
62. P. Houston, I. Perugia, A. Schneebeli and D. Schötzau, Interior penalty method for the indefinite time-harmonic Maxwell equations, *Numer. Math.* **100** (2005) 485–518.
63. P. Houston, I. Perugia and D. Schotzau, Mixed discontinuous galerkin approximation of the maxwell operator, *SIAM J. Numer. Anal.* **42** (2004) 434–459.

64. Q. Hu, S. Shu and J. Zou, A mortar edge element method with nearly optimal convergence for three-dimensional Maxwell's equations, *Math. Comp.* **77** (2008) 1333–1353.
65. J. Huang and J. Zou, Some new a priori estimates for second-order elliptic and parabolic interface problems, *J. Differential Equations* **184** (2002) 570–586. 12
66. J. Huang and J. Zou, Uniform a priori estimates for elliptic and static Maxwell interface problems., *Disc. Cont. Dynam. Sys., Series B* **7** (2007) 145–170. 12
67. P. Huang, H. Wu and Y. Xiao, An unfitted interface penalty finite element method for elliptic interface problems, *Comput. Methods Appl. Mech. Engrg.* **323** (2017) 439–460. 4
68. H. Ji, An immersed Raviart–Thomas mixed finite element method for elliptic interface problems on unfitted meshes, *J. Sci. Comput.* **91** (2022) 1–33.
69. R. Kafafy, T. Lin, Y. Lin and J. Wang, Three-dimensional immersed finite element methods for electric field simulation in composite materials, *Internat. J. Numer. Methods Engrg.* **64** (2005) 940–972. 15
70. J. Li, J. M. Melenk, B. Wohlmuth and J. Zou, Optimal a priori estimates for higher order finite elements for elliptic interface problems, *Appl. Numer. Math.* **60** (2010) 19–37. 16
71. T. Lin, Y. Lin and X. Zhang, Partially penalized immersed finite element methods for elliptic interface problems, *SIAM J. Numer. Anal.* **53** (2015) 1121–1144. 4, 32
72. H. Liu, L. Zhang, X. Zhang and W. Zheng, Interface-penalty finite element methods for interface problems in H^1 , $H(\text{curl})$, and $H(\text{div})$, *Comput. Methods Appl. Mech. Engrg.* **367** (2020) 113137. 4
73. D. Lo, *Finite Element Mesh Generation* (CRC Press, 2014).
74. L. D. Marini, An inexpensive method for the evaluation of the solution of the lowest order Raviart–Thomas mixed method, *SIAM J. Numer. Anal.* **22** (1985) 493–496. 3
75. P. Monk, *Finite Element Methods for Maxwell's Equations* (Oxford University Press, 2003). 3
76. J.-C. Nédélec, Mixed finite elements in \mathbf{R}^3 , *Numer. Math.* **35** (1980) 315–341. 20
77. J. Nitsche, Über ein Variationsprinzip zur Lösung von Dirichlet-Problemen bei Verwendung von Teilräumen, die keinen Randbedingungen unterworfen sind, *Abhandlungen aus dem Mathematischen Seminar der Universität Hamburg* **36** (1971) 9–15. 4
78. M. A. Olshanskii, A. Reusken and J. Grande, A finite element method for elliptic equations on surfaces, *SIAM J. Numer. Anal.* **47** (2009) 3339–3358. 21
79. C. Pflaum, Subdivision of boundary cells in 3d, . 4
80. P.-A. Raviart and J. M. Thomas, A mixed finite element method for 2nd order elliptic problems, in *Mathematical aspects of finite element methods (Proc. Conf., Consiglio Naz. delle Ricerche (C.N.R.), Rome, 1975)* (Springer, Berlin, 1977), pp. 292–315. Lecture Notes in Math., Vol. 606. 20
81. J. Saranen, On generalized harmonic fields in domains with anisotropic nonhomogeneous media, *J. Math. Anal. Appl.* **88** (1982) 104–115.
82. J. Saranen, On electric and magnetic problems for vector fields in anisotropic nonhomogeneous media, *J. Math. Anal. Appl.* **91** (1983) 254–275.
83. X.-C. Tai and R. Winther, A discrete de rham complex with enhanced smoothness, *Calcolo* **43** (2006) 287 – 306.
84. S. Vallaghé and T. Papadopoulo, A trilinear immersed finite element method for solving the electroencephalography forward problem, *SIAM J. Sci. Comput.* **32** (2010) 2379–2394. 2
85. J. Xu, The auxiliary space method and optimal multigrid preconditioning techniques for unstructured grids, *Computing* **56** (1996) 215–235.

86. J. Xu and S. Zhang, Optimal finite element methods for interface problems, in *Domain Decomposition Methods in Science and Engineering XXII*, eds. T. Dickopf, M. J. Gander, L. Halpern, R. Krause and L. F. Pavarino (Springer International Publishing, Cham, 2016), pp. 77–91.
87. J. Xu and Y. Zhu, Robust Preconditioner for $H(\text{curl})$ Interface Problems, in *Domain Decomposition Methods in Science and Engineering XIX* (Springer, Berlin, Heidelberg, 2011), pp. 173–180. 7
88. J. Ying and D. Xie, A new finite element and finite difference hybrid method for computing electrostatics of ionic solvated biomolecule, *J. Comput. Physics* **298** (2015) 636–651. 2
89. S. Zhao and G. W. Wei, High-order FDTD methods via derivative matching for Maxwell's equations with material interfaces, *J. Comput. Phys.* **200** (2004) 60–103. 4