

# Embedded solids of any dimension in the X-FEM.

## Part I - Building a dedicated P1 function space

F. Duboeuf and E. Béchet

*University of Liège, Aerospace & Mechanical Engineering Department - Computer-aided Geometric Design,  
Allée de la Découverte 9, B-4000 Liège, Belgium*

---

### Abstract

This paper focuses on the design of a dedicated P1 function space to model elliptic boundary value problem on a manifold embedded in a space of higher dimension. Using the traces of the linear P1 shape functions, it introduces an algorithm to reduce the function space into an equivalent space having the same properties than a P1 Lagrange approximation. Convergence studies involving problems of codimension one or two embedded in 2D or 3D show good accuracy with regard to classical finite element and analytical solutions. The effects of the relative position of the domain with respect to the mesh are studied in a sensitivity analysis; it illustrates how the proposed solution allows to keep the condition number bounded. A comparative study is performed with the method introduced by Olshanskii *et al.* 2009 on a closed surface to validate our approach. The robustness of the proposed approach is investigated with regard to their method and that of Burman *et al.* 2016. This paper is the first in a series of two, on the topic of embedded solids of any dimension within the context of the extended finite element method. It investigates problems involving borderless domains or domains with boundary subject to Dirichlet constraint defined only on the boundaries of the bulk mesh, while the forthcoming paper overcomes this limitation by introducing a new stable Lagrange multiplier space for Dirichlet boundary condition (and more generally stiff condition), that is valid for every combination of the background mesh and manifold dimensions. The combination of both algorithms allows to handle any embedding *i.e.* 1D, 2D and 3D problems embedded in 2D or 3D background meshes.

*Keywords:* Embedded solids, Extended finite element method, P1-equivalent space, Vertex space reducer

---

*E-mail address:*

[frederic.duboeuf@gmail.com](mailto:frederic.duboeuf@gmail.com), [fduboeuf@ulg.ac.be](mailto:fduboeuf@ulg.ac.be)(F. Duboeuf); [eric.bechet@ulg.ac.be](mailto:eric.bechet@ulg.ac.be)(E.Béchet).

*Preprint submitted to Finite Elements in Analysis and Design*  
*Article accepted for publication DOI:10.1016/j.finel.2016.12.001*

*July 30, 2016*  
*December 12, 2016*

**Contents**

<b>1</b>	<b>Introduction</b>	<b>2</b>
1.1	Background . . . . .	4
1.2	Objectives . . . . .	8
1.3	Outline . . . . .	9
<b>2</b>	<b>Model problem</b>	<b>10</b>
2.1	Weak formulation . . . . .	11
2.2	Finite element discretization . . . . .	12
<b>3</b>	<b>Design of the P1-equivalent space</b>	<b>16</b>
3.1	Terminology . . . . .	16
3.2	Description of the algorithm . . . . .	17
<b>4</b>	<b>Numerical validations</b>	<b>21</b>
4.1	Problem of codimension one in 2D - a straight line . . . . .	21
4.2	Problem of codimension one in 3D - a flat surface . . . . .	23
4.3	Problem of codimension two in 3D - a helix . . . . .	27
4.4	Condition number of the problem . . . . .	30
4.5	Validation on another problem of codimension one in 3D - a sphere . . . . .	34
4.6	Analysis of the matrix properties . . . . .	37
<b>5</b>	<b>Conclusion</b>	<b>50</b>

**1. Introduction**

Computer-Aided Design (CAD) is now the traditional design tool, able to combine manifolds of different dimensions to represent the geometry of a part. It is usually associated with Computer Aided Engineering (CAE) tools such as the Finite Element (FE) method in an overall optimization process. Now ubiquitous in industry, the standard version of FE method requires a discretized representation of a manifold that conforms to its geometry. This is done with specific mesh generation algorithms and often implies tedious user interactions.

The same discretization is then used for the field approximation, see Figure 1(a). According to [1], this preliminary step may even represent up to 80% of the time taken by the whole CAE process. To reduce these costs in the transition from the continuous CAD model to the discrete FE model, various meshless techniques and approaches using non-matching meshes have been proposed in the literature. Among these, the eXtended Finite Element Method (X-FEM) has originally been developed to simulate the propagation of cracks in solids [2, 3]. A crack is indeed a submanifold of codimension one and is considered as a discontinuity of one dimension lower than that of the solid. By analogy, boundaries or internal boundaries of the solid also define geometric discontinuities and may be embedded in a background mesh. The mesh thus no longer depends on the position of the interfaces in the domain. This allows to dissociate the field approximation and the boundary representation, see Figure 1(b).

For the sake of completeness, the issue relating to the CAD–CAE transition is also tackled in a different way in IsoGeometric Analysis (IGA). In this method, the purpose is to reuse the CAD description (instead of a mesh) to represent the computational domain and approximate the unknown field, see Figure 1(c). Starting with the same paradigm for the geometric description, a Geometry-Independent Field approximation (GIFT) introduced in [4] suggests using a different discretization space for the field approximation, see Figure 1(d). This allows to adapt the approximation of the solution regardless of the geometric description, preserving thereby similar flexibility given by h- or p-refinement in the FEM. However, it remains challenging for both of these methods to generate volume parameterizations for 3D manifolds by employing existing CAD models.

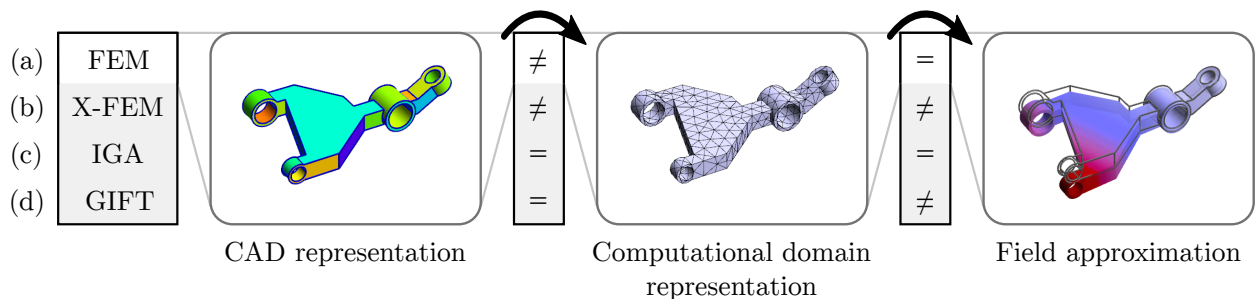


Figure 1: Paradigms introduced by the FEM, the X-FEM, IGA and GIFT through the CAD–CAE transition.

Currently, the X-FEM has only been applied to embed domains having the same dimension as the FE mesh [5], *i.e.* submanifolds of codimension zero. Thus, a two-dimensional geometry must be contained in a planar mesh, just like a three-dimensional solid may be embedded into a 3D bulk mesh. But how to represent a curve, or a curved surface ? In this paper we propose a new approach to embed a manifold of codimension one or two. For this, new function spaces must be specifically designed to solve classical PDEs defined on the manifolds using a background mesh of a higher dimension, either 2D or 3D.

Several questions arise when using a non-conforming mesh to carry out a simulation on a domain of zero or higher codimension:

- (i) Which geometric representation should be used to describe all CAD manifolds within the X-FEM, in particular, embedded curves in 3D ?
- (ii) Which function space (possibly based on the linear shape functions of the FEM) allows to model a solid embedded in a space of higher dimension ?
- (iii) How to enforce boundary conditions, especially Dirichlet like boundary conditions, in such a setting ?

The general treatment of boundary conditions is actually a problem in its own right, we will therefore propose some solutions in a subsequent paper [6]. This paper is focused on the definition of an optimal way of representing fields on an embedded manifold of codimension one to two.

### *1.1. Background*

In order to reduce the dependence of the FE approximation on the geometric description, various approaches have been proposed in the literature. Here, we describe some of them, starting with the case of solids embedded in a background mesh of the same dimension.

When boundaries cross most of the elements of the mesh, a first approach is to use classical FEs, solving a problem defined on a conforming mesh. This consists in remeshing the narrow-band of elements neighbouring the interface. Two possibilities arise: either (i)



move the nodes of the mesh in order to be aligned with the boundaries, as proposed in [7] — this requires specific algorithms to insert the geometry into the mesh [8, 9] — or (ii) roughly approximate the domain by the set of elements covering it, resulting in an approximated interface (the interface is approximated toward the exterior of the domain). To recover a good approximation of the embedded boundary, the discretization may be locally adapted with multi-level refinement [10] or anisotropic mesh adaptation [11].

Another approach combines different meshes associated with each domain so as to avoid the constant update of the interface, e.g. in case of a moving boundary in large displacements with fluid-structure interaction. Among them, let us mention in particular embedded meshes [12], overlapping domain decomposition also called chimera [13, 14], or more atypical: the s-method [15] and the Arlequin method [16]. The combination of overlapping finite element meshes induces an additional memory cost and requires to update several meshes, e.g. in case of solid deformation. Moreover, these meshes are subject to continuity constraints at the boundary, for example, between the foreground embedded mesh and the background domain. These coupling constraints are similar to boundary condition enforcement and can be dealt with in different ways (mortar method [17] or Lagrange multipliers [18], Nitsche’s methods [19], discontinuous Galerkin methods [12]).

Nevertheless, those approaches do not completely dissociate the mesh generation step from the representation of the domain. To achieve this, some others use a unique mesh, which is independent of the geometry of the problem. These approaches differ from FEM in specific treatment of the narrow-band of elements neighbouring the interface, when both physical and non-physical parts are defined on an element. Depending on how these elements are treated, two approaches are possible. On the one hand, spread interface approaches have been initially introduced in [20]; among them the Immersed Boundary Method (IBM) in [21] and the Fat Boundary Method (FBM) in [22]. The equations are smoothed on whole elements near the interface and not only on the physical part of these elements. On the other hand, various thin interface approaches have been proposed in the literature, e.g. the Fictitious Domain Method (FDM) [23] which nevertheless requires an interfacial mesh, the Immersed Interface Method (IIM) [24, 25] modifying the discrete operator near the interface

with Taylor series, and the X-FEM [26, 27] benefiting from Partition of Unity Methods (PUM) [28, 29]; where the approximated interface has the same dimension as the original interface.

Here, in case of a solid embedded in a bulk mesh of the same dimension, it is appropriate to simply use the classical FE function space built on the background mesh and integrate only on the interior of the domain. Therefore, the mesh used for the field approximation can be kept, and only the domain of integration must be adapted in order to be body-fitted. This allows to preserve the mesh topology and avoid time-consuming remeshing processes. To this end, a partition of the elements intersected by an interface into sets of sub-elements is used [30]. This procedure allows to introduce a geometric discontinuity into an element without any enrichment of the function space. Starting with shape functions defined over classical FE (called parent elements in the sequel — see Figure 2a), restrictions of the parent shape functions are used on sub-elements of the domain (see Figure 2b).

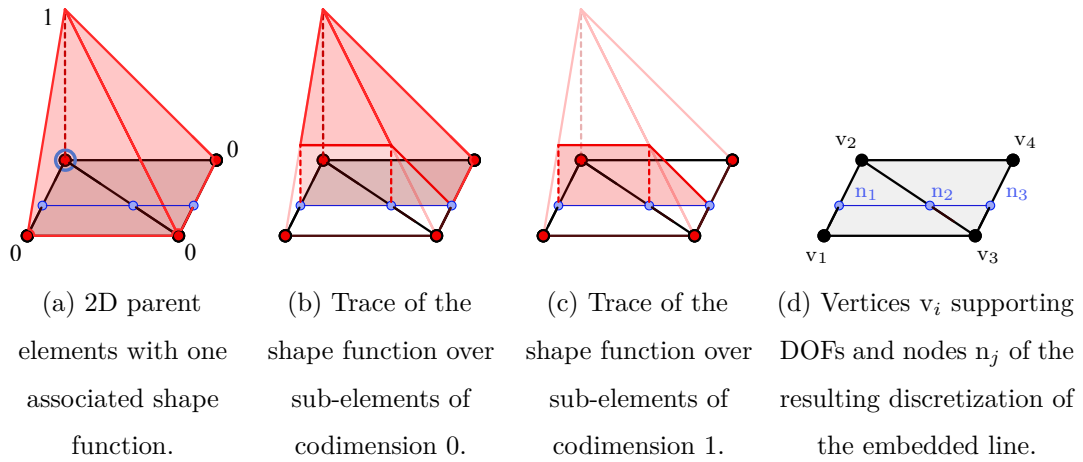


Figure 2: Two triangular elements with an embedded line.

Moreover, embedded interfaces may require especial treatment if boundary conditions have to be enforced. Because these interfaces have a higher codimension than that of the domain, dedicated integration rules should be used. Furthermore, if Lagrange multipliers are to be used for boundary condition enforcement, an appropriate function space must be designed. As we will see, this is not the only case where a dedicated function space is necessary.

The direct use of the FE shape functions of the bulk mesh is possible for a problem domain of codimension zero. However, this approach is no longer possible in other configurations, when the problem domain and the discretization do not have the same dimension.

To tackle this difficulty, one may return to a problem of codimension zero. Similarly to the spread interface approaches introduced in codimension zero, this approach extends the domain of integration to a narrow-band of elements [31, 32]. Although robust, it lacks consistency by solving a different problem on a domain of a higher dimension (that of the mesh).

To preserve the nature of the problem, one might instead try to build the function space on the resulting intersection with the background mesh. On this induced surface triangulation, the quality of the approximation depends on a maximum angle condition [33], for which a sufficient condition results in a minimum angle condition on a tetrahedral background mesh, as proved in [34]. The latter condition is reasonable since the mesh generation algorithms are freed from the respect of boundaries. Unfortunately, the induced surface triangulation may be formed of arbitrarily small triangles (e.g. at the vicinity of the background mesh vertices), which numerically lead to an ill-conditioned system. A specific preconditioning based on diagonal scaling has been proposed in [34] to address this problem. But, this approach does not take advantage of the background mesh, since new elements supporting the shape functions must be generated on the surface.

In order to address 1D or 2D problems without remeshing (especially benefiting with evolving domains), one may use the trace of a higher dimensional space built upon the bulk mesh. An illustration of the trace of a shape function used in 2D is shown in Figure 2c. Concerning the codimension one with closed curves in 2D and closed surfaces in 3D, the resulting discrete method based on P1 FE has already been studied in [35].

In that case, the complete function space defined on the mesh is not suitable to model piecewise linear fields. To illustrate this fact, a simple case of codimension one in 2D is shown in Figure 2d. For two triangles connected by an edge and intersected by a straight line, there are four vertices (the vertices  $v_i$  of the triangles) and only three intersection

points (the nodes  $n_j$ ). These intersection points may be used as support for degrees of freedom (DOFs) to get a piecewise linear approximation along the line. There are obviously more DOFs associated with vertices than nodes along the line. We thus need to deplete the function space that is, in this particular case, allowing one given field to be represented in multiple independent ways. The resulting system of equations would be singular if nothing is done.

To overcome this ill-posed problem, the algebraic system may be treated to reduce the kernel to be null and ensure that all equations of the linear system are independent as proposed in [36]. The method proposed by Olshanskii *et al.* [35] considers a specific solver using a Gauss-Seidel preconditioned Conjugate Gradient method, which has been applied on surface PDEs [37] with local adaptation [38, 39] and higher order approximation [40]. Another method consists in additional stabilisation terms computed over the bulk mesh. In this way, we tend to recover a well-posed problem of codimension zero. This has been achieved by Burman *et al.* [41] with a face-based stabilisation, and more recently, with a full gradient stabilisation [42]. Here, we propose to directly build a linear algebraic system free of singularities by modifying the function space.

Concerning the case of the codimension two, very few studies do exist, and most of them recover a body-fitted mesh or use independent overlapping meshes, cf. [43, 44].

### 1.2. Objectives

Here, we choose to use only one mesh in which manifolds of various dimensions can be embedded and coupled. This choice may be justified by:

- the reduction of the generation of the mesh supporting the FE approximation to its simplest form. It is then possible to drastically optimize both its memory footprint and the CPU (Central Processing Unit) cost, by taking advantage of grid structures or GPU (Graphic Processing Unit) computational power for instance.

- the *a priori* compatibility for function spaces defined on each manifold, which alleviates many practical problems. For example, to evaluate integrals mixing discrete spaces defined on different grids with an adequate quadrature, or to satisfy inf–sup condition potentially introduced by a mixed-formulation on two grids. These problems happen, e.g., with fluid-structure interaction.

Of course, the accuracy of the approximation is constrained by the h- or p-refinement of the background mesh. Consequently, each manifold cannot be refined independently, leading the whole coupled problem to be treated on an equal footing.

The objectives of this paper are hence fourfold:

1. Propose an approach dealing with any embedding in a single framework, neither modifying the variational formulation, nor requiring a dedicated solver;
2. Demonstrate its optimal convergence through several numerical examples covering all codimension configurations;
3. Compare with methods available in the literature on borderless domains, and show that our approach is compatible with the enforcement of boundary conditions using former algorithms based on Lagrange multipliers.
4. Show the robustness of the approach with regard to alternative methods from the literature.

### 1.3. Outline

The remainder of this paper is outlined as follows. We formulate the model problem in the next section and highlight compatibility requirements for its discretization. In order to design P1-equivalent function space for codimension one and two, Sect. 3 introduces concepts suitable for each dimension into a new dedicated algorithm. Numerical results demonstrating the accuracy and robustness of our approach are presented in Sect. 4. Finally, the last section provides a summary and some concluding remarks.

## 2. Model problem

Let us consider a Poisson equation as an elliptic boundary value problem posed on a manifold  $\Omega$  of dimension  $m$  embedded in a Cartesian domain  $\mathcal{T} \subset \mathbb{R}^n$ , with  $1 \leq m \leq n$  and  $n = 1, 2$  or  $3$ . The case of embeddings in 1D is straightforward and will not be developed in the sequel (see [45] for instance).

In order to solve the model problem, a special care should be given to the difference of dimensions between  $\Omega$  and the underlying domain  $\mathcal{T}$ , the codimension denoted  $\text{codim}_{\mathcal{T}}(\Omega) := \dim \mathcal{T} - \dim \Omega = m - n$ . Indeed, the space, in which to search for a solution, will not be directly defined on  $\Omega$  but in the ambient domain  $\mathcal{T}$ . Therefore, different operators must be introduced depending on the codimension of the problem. For instance, a tangent space to  $\Omega$  may be recovered by using tangential differential calculus.

When the codimension is equal to zero, the Poisson problem just involves the ordinary Laplace operator. In case of a higher codimension, the extension of the Euclidean Laplacian to an arbitrary Riemannian manifold — the Laplace-Beltrami operator — is used. Both cases may be handled by the following notation: let  $g : \mathcal{T} \rightarrow \mathbb{R}$  be a sufficiently smooth function, let  $\mathbf{n}_{\Omega}$  and  $\mathbf{t}_{\Omega}$  be the normal and tangent vectors to  $\Omega$ , we define the derivative of  $g$  along  $\Omega$  by:

$$\nabla_{\Omega} g := \begin{cases} \nabla g & \text{if } \text{codim}_{\mathcal{T}}(\Omega) = 0, \\ \nabla g - \nabla g \cdot \mathbf{n}_{\Omega} \mathbf{n}_{\Omega} & \text{if } \text{codim}_{\mathcal{T}}(\Omega) = 1, \\ \nabla g \cdot \mathbf{t}_{\Omega} \mathbf{t}_{\Omega} & \text{if } \text{codim}_{\mathcal{T}}(\Omega) = 2. \end{cases} \quad (1)$$

This definition allows to write the Laplacian in a more general operator  $\Delta_{\Omega} := \nabla_{\Omega} \cdot \nabla_{\Omega}$ . In this way, the different configurations of the model problem may be written in a single set of equations:

$$\begin{aligned} -\Delta_{\Omega} u &= f & \text{in } \Omega, \\ u &= u_D & \text{on } \Gamma_D, \\ \nabla_n u &= t_N & \text{on } \Gamma_N. \end{aligned} \quad (2)$$

Here,  $\nabla_n u$  is the outward flux through the boundary  $\Gamma_N$  of  $\Omega$ . Furthermore,  $u_D$  and  $t_N$  define respectively Dirichlet and Neumann BCs.

In the definition (1), the first case is associated with solids embedded in a space of the same dimension, the main difference with a standard FE problem being the use of a non-conforming mesh. The second case corresponds to the hyperplanes of  $\mathbb{R}^n$ , e.g. with curves embedded in a two dimensional grid or surfaces in a 3D mesh. The last case is associated with curves in 3D. The latter two cases have potential applications in mixed-dimensional coupling, or when the  $m$ -dimensional problem domain has curvatures in the ambient space of dimension  $n$ . Then, the model problem covers any embedding as depicted in Figure 3.

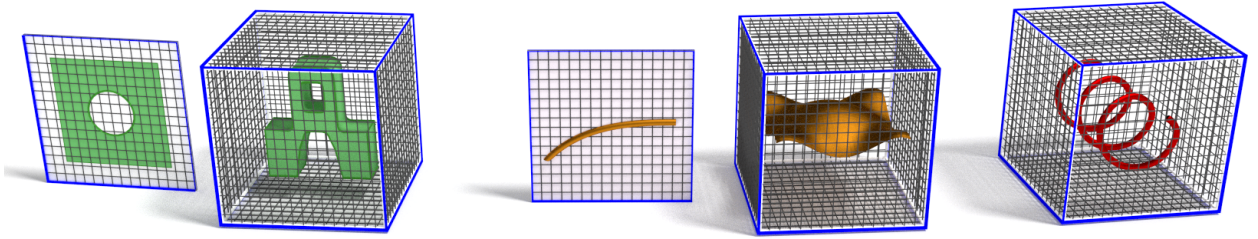


Figure 3: Embedded solids (from left to right): 2D and 3D problem domains of codimension zero, 1D and 2D problem domains of codimension one, 1D problem domain of codimension two.

### 2.1. Weak formulation

The weak form of the model problem given in Equation (2) is presented below. Considering  $\mathcal{U} := H^1(\Omega)$  with its usual norms, we define two subsets  $\mathcal{U}_D := \{u \in \mathcal{U} : u = u_D \text{ on } \Gamma_D\}$  and  $\mathcal{U}_0 := \{u \in \mathcal{U} : u = 0 \text{ on } \Gamma_D\}$ .

$$\text{Find } u \in \mathcal{U}_D : \int_{\Omega} \nabla_{\Omega} u \cdot \nabla_{\Omega} v \, d\Omega = \int_{\Omega} f v \, d\Omega + \int_{\Gamma_N} t_N v \, d\Gamma, \quad \forall v \in \mathcal{U}_0. \quad (3)$$

Note that, from an implementation perspective, the gradient operator defined on  $\Omega$  may be expressed using projection operators. In the case of codimension one (e.g. a surface embedded in 3D), the projector onto the surface  $\Omega$  is  $\underline{\underline{P}}_{\Omega} := \underline{\underline{I}} - \underline{\underline{n}}_{\Omega} \otimes \underline{\underline{n}}_{\Omega}$ . For a line embedded in 3D (codimension two), the projector onto the tangent line of  $\Omega$  is  $\underline{\underline{P}}_{\Omega} := \underline{\underline{t}}_{\Omega} \otimes \underline{\underline{t}}_{\Omega}$ . In both cases the projected gradient may be written as  $\nabla_{\Omega} = \underline{\underline{P}}_{\Omega} \nabla$  for scalar fields.

## 2.2. Finite element discretization

To begin with, the discretization of the geometry must be introduced. In order to dissociate the problem domain from the finite element mesh discretization, the bulk domain must completely contain the problem domain. Let  $h$  be the characteristic mesh size. We consider a quasi uniform mesh  $\mathcal{T}^h$  of shape-regular linear elements (lines in 1D, triangles in 2D or tetrahedra in 3D) in which the problem domain is embedded. It results in an unfitted discretized domain  $\Omega^h$  of the problem.

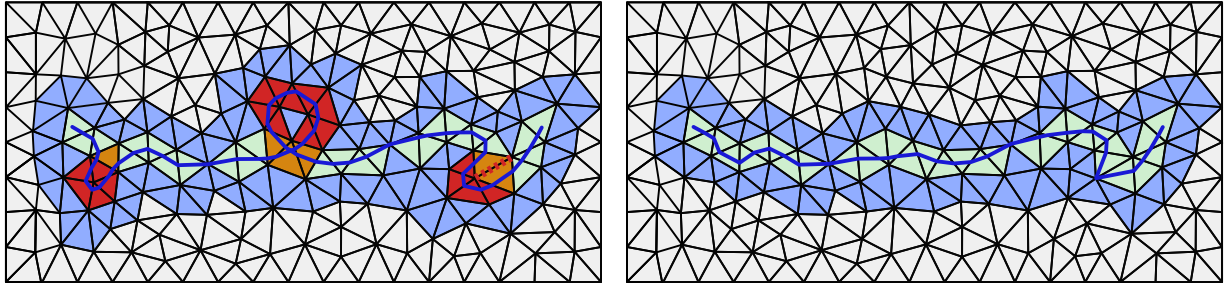
### *Representation of embedded solids*

The geometry of the problem domain can be represented either implicitly or explicitly. Both approaches can be used within the X-FEM. The use of level-sets [46, 47], for instance, is valuable in the case of moving interfaces, such as cracks [2, 3], two-phase and free-surface flows [48] with potential topology changes. The iso-zero surface of a signed distance scalar function is used to describe manifolds of codimension one. It may be extended to the codimension  $m$  by using  $m$  level-sets as proposed with vector valued level-set functions in [49]. Considering a CAD geometry handled in the context of the X-FEM, the set of surfaces defining the boundaries of a solid could be described by level-sets as well. However, most of—if not all—CAD representations are based on B-Rep (Boundary Representation), which makes a conversion to level-sets difficult because the flexibility of an implicit representation is limited in the case of 3D edges. In fact, due to numeric error in CAD data, the edge identification by level-set intersections in the case of tangent surfaces leads to a lack of robustness. Moreover, the representation of 1D models with fixed topology such as beams is not well suited with this type of implicit approach. We suggest an explicit approach to describe 3D curves, in a way similar to that introduced in [50]. In order to represent all the CAD manifolds in a robust way, an extension of this kind of explicit approach may be investigated, but this is not the scope of the present work.

The resulting approximation of the domain must be suitable to define a function space. Indeed, the supports of relevant function space based only on FE shape functions should be free of overlap, like those (dark shaded) shown in Figure 4a. Otherwise, locking will occur



on the domain because the function space cannot represent linear functions any more. The identification of these overlapping regions and the creation of an adequate geometry for the FE computation may be done by post-processing the embedded solid representation, see Figure 4b. The overlapping regions are detected by finding, in each element, the upper and lower bounds of the parametric coordinates associated with the line. The left and centre patterns shown in Figure 4a present elements for which bounds define a subset already covered by those of another element. The right pattern introduces two elements connected by an edge (hatched) and not consecutively crossed by the line.



(a) Incompatible discretization with some overlapping regions.

(b) Post-processed geometry free of overlap.

Figure 4: Illustration of some locking on an embedded line in 2D and solution retained after post-treatments.

Note that most of these patterns disappear naturally when the mesh is refined. Others are due to bad properties of the input geometry, with a lack of injectivity on the immersion (*i.e.* the curve crossing itself). Here, we consider only embedding and not immersion. At this stage, the final discretization is compatible with the representation of any linear field on the embedded domain.

#### *Definition of the discrete function space*

The discretization of the weak form given in Equation (3) is made using finite element. Let  $\mathcal{U}^h$  be the discrete space of the field of unknowns, and  $N_i|_{\Omega^h}$  the trace on  $\Omega^h$  of the linear

shape function associated with the vertex  $i$ :

$$\mathcal{U}^h := \left\{ \sum_i q_i N_i|_{\Omega^h}, q_i \in \mathbb{R} \right\}. \quad (4)$$

Let  $\mathcal{U}_D^h$  and  $\mathcal{U}_0^h$  be two subsets of  $\mathcal{U}^h$  with adequate constraints, these spaces are used for the discretized weak formulation:

Find  $u^h \in \mathcal{U}_D^h$ :

$$\int_{\Omega^h} \nabla_{\Omega^h} u^h \cdot \nabla_{\Omega^h} v^h \, d\Omega = \int_{\Omega^h} f v^h \, d\Omega + \int_{\Gamma_N^h} t_N v^h \, d\Gamma, \quad \forall v^h \in \mathcal{U}_0^h. \quad (5)$$

As mentioned in the introduction, the definition of a function space using traces of finite element shape functions defined on  $\mathcal{T}^h$  must be done with caution when the problem domain is embedded in a non-conforming mesh of higher dimension, *i.e.* if  $\text{codim}_{\mathcal{T}^h}(\Omega^h) > 0$ . Using all the FE shape functions for the discretization of the bulk field  $u^h$ , the resulting full P1 space leads to a singular linear system. In case of a line crossing tetrahedra, as shown in Figure 5 for instance, the three equations at each intersection point involve five unknowns associated with vertex shape functions, resulting in an ill-posed problem with non unique solutions.

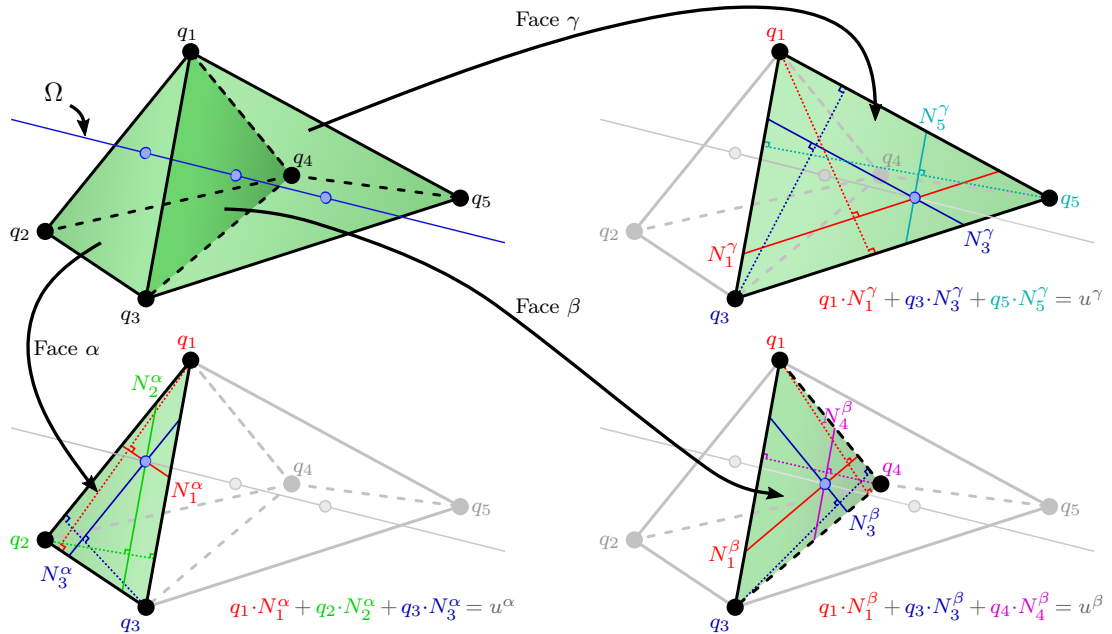


Figure 5: A domain of interest  $\Omega$  embedded in a 3D background mesh not matching the line.

To overcome this problem, we propose to recover the algebraic independence of the equations by reducing the function space in line with algorithms found in [51, 52, 53]. These were designed to generate a stable Lagrange multiplier space from P1 FE. The resulting space is then appropriate to enforce, along an embedded interface, Dirichlet type boundary conditions for linear interpolating FE space. They rely on graph theory and take advantage of the fixed mesh topology to satisfy the inf-sup condition. However, this dedicated space is not able to represent a linear field (only a constant field). Therefore, we denote it P0-equivalent. To build a function space able to represent any linear field, we introduce in Sect. 3 a new reduction algorithm that is suitable to any combination of dimension and codimension. The resulting space will be called P1-equivalent.

The idea itself and also several initial 2D and 3D examples were first presented at the WCCM in Barcelona in July 2014. It is interesting to draw a parallel with a paper [54] about the design of a stable Lagrange multiplier space to solve an embedded interface problem with quadratic X-FEM. In the linear case, a P1 approximation of the interface geometry may be combined with a P1 space for the primal field and a stable space P0-equivalent for the dual field of Lagrange multipliers. In the quadratic case, both the geometric description and the Lagrange multipliers must be carefully selected with the aim of reaching optimal convergence rates. Thus, a reducer algorithm has been introduced to build a P1-equivalent Lagrange multiplier space in the 2D case. Interpolation properties of the resulting space are also analysed.

Although the design of a dedicated P1-equivalent space gives a non-singular linear system, it may be arbitrarily ill-conditioned. A sensitivity analysis is performed Sect. 4.1 on the position of the surface in the background mesh in order to illustrate this, and a solution is suggested.

Finally, a specific integration quadrature is required to accurately compute integrals given in Equation (5) and defined over embedded domains of different codimensions. To this end, the selected geometric representation is linearly approximated. This choice is discussed Sect. 4.5 and compared with the quadratic approximation suggested by Olshanskii [35]. In the numerical applications, appropriate Gauss integration quadratures will be used over both

classical and sub-elements.

### 3. Design of the P1-equivalent space

In order to define a unique algorithm, valid in any configuration of the embedded problem, we briefly introduce some concepts.

#### 3.1. Terminology

Here, we define the terminology which is used with the discrete problem. Illustrations of the relative concepts are given in Figure 6.

**Definition 3.1.** *Mesh component.* An element (tetrahedron, triangle, line, point) is a mesh component (volume, face, edge, vertex, see Figure 6a). A mesh component may be composed of other components of strictly lower dimension (Figure 6b). We can extend this notion to sub-elements with *sub-components*.

Note that sub-elements are especially introduced where  $\Omega$  is non-conform to the mesh, otherwise no distinction with classical elements is made.

**Definition 3.2.** *Mesh connectivity.* Each mesh component is linked to others. Its connectivity is defined by these connected components sharing all its vertices. In this way, the hierarchy of the connectivities is defined with strictly increasing dimension (Figure 6c).

**Definition 3.3.** *Support.* A sub-component is built on a single component: its support (Figure 6d). A component can support multiple sub-components: its children.

**Definition 3.4.** *Support connectivity.* According to the mesh connectivity, the support connectivity may be defined from the vertices of the underlying mesh component.

Here, the term "support" has a different meaning from that classically used in the FE context (*i.e.* non-zero definition domain of a shape function). This is introduced to describe the relation between the mesh  $\mathcal{T}^h$  and the induced discretization  $\Omega^h$ . Combined with the mesh topology (components and connectivities), it gives access to the bulk shape functions in order to design the function space.

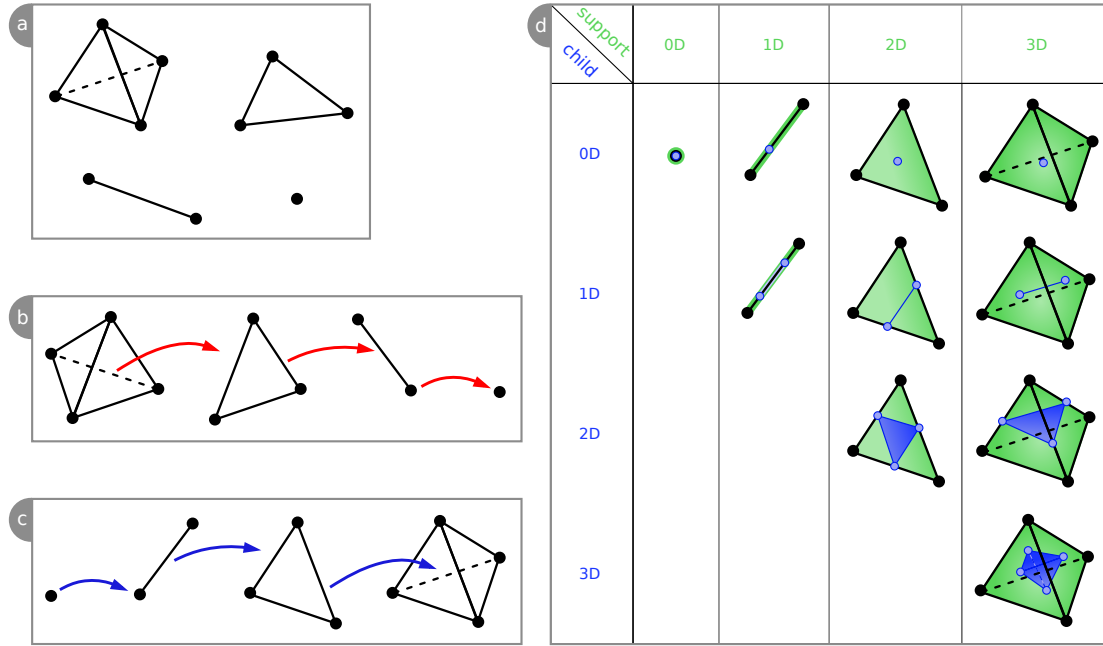


Figure 6: Illustration of the concepts. On the left: mesh components (a) with the hierarchy of components (b) and the hierarchy of connectivities (c). On the right: supports of sub-components for each dimension (d).

These concepts are not specific to the dimensions of the problem or the background mesh. This way, we will be able to use the same algorithm for embeddings of codimension one or two in 2D and 3D.

### 3.2. Description of the algorithm

In order to set an optimal approximation over a domain with boundary, a linear extension of the domain is used in the elements intersected by the boundary. The discretized domain  $\Omega^h$  is extended to a domain  $\overline{\Omega^h}$  whose boundaries fit with those of the elements of the background mesh. It is exclusively built for the purpose of defining the function space properly; the domain of integration ( $\Omega^h$ ) remains unchanged.

The intersections between the linear extension of the embedded line and the mesh yield a set of supports. On each support, one or more vertex shape functions are defined. Groups of independent shapes functions are detected by using the support connectivities (rely on graph theory). For each of those, the number of redundant shapes functions is identified, according

to the topology of the mesh. Indeed, in the simple 2D example given in the introduction (Figure 2d), only one DOF was in excess; while the 3D example of the previous section (Figure 5) presents two DOFs that should be merged to avoid linear dependence of the traces of the shapes functions. The proposed algorithm is illustrated in Figures 7 and 8 and is detailed below in Algorithm 1.

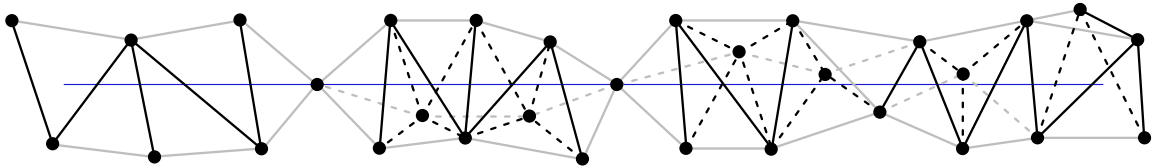
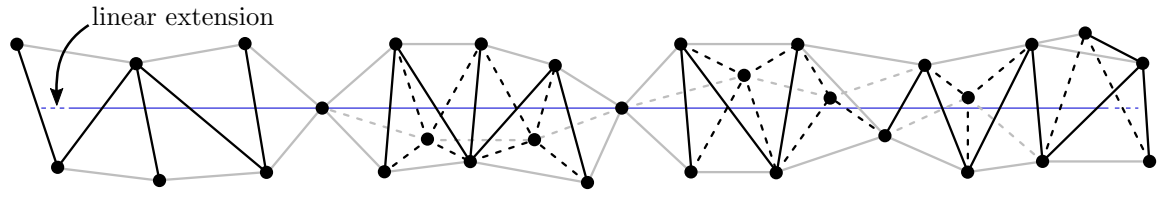
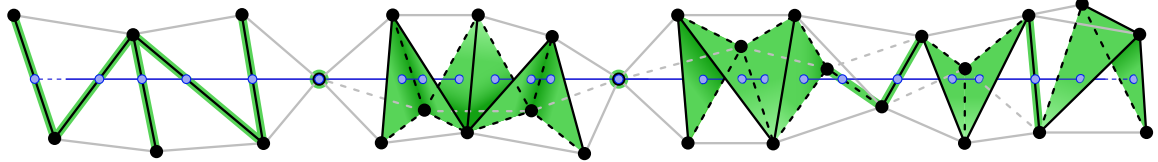


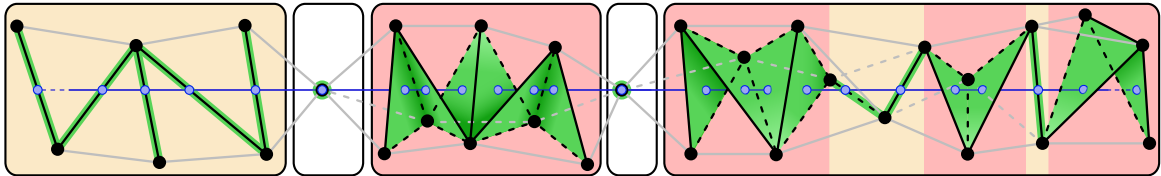
Figure 7: An embedded line that is unfitted to the narrow band of elements.



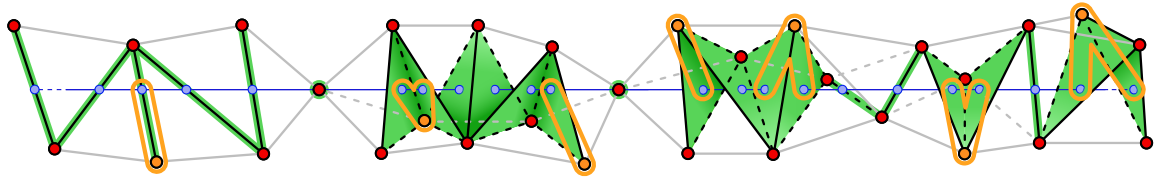
(a) Linear extension of the embedded line.



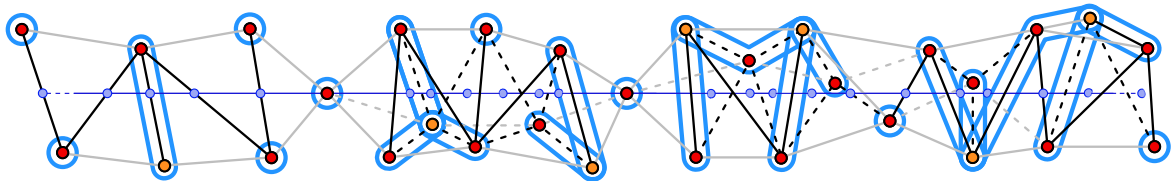
(b) Supports of the intersection points between the linear extension of the embedded line and the mesh.



(c) Five sets of connected supports (with various dimensions of supports in the right set).



(d) `Full_Shape_Functions` (drawn in red) and `Distributed_Shape_Functions` (outlined in orange) associated with the vertices.



(e) Resulting shape functions.

Figure 8: The four steps of the algorithm and the final P1-equivalent space.

---

**Algorithm 1:** Vertex space reducer.

---

**Input:** (Figure 7)

A set  $\Omega^h$  of elements or sub-elements embedded in a background mesh.

**Output:** Identification of groups of independent shape functions.

Characterization of the vertex shape functions: `Full_Shape_Function` (resp. `Distributed_Shape_Function`) associated with its vertex (resp. distributed on the other vertices of the connected supports of dimension  $\leq \text{codim}_{\mathcal{T}^h}(\Omega^h)$ ).

1: (Figure 8a)

**if**  $\partial\Omega \neq \emptyset$  **then**

    └ Build  $\overline{\Omega^h}$  the linear extension of  $\Omega^h$ ,

**else**

    └ Define  $\overline{\Omega^h} := \Omega^h$ .

2: (Figure 8b)

Build the connectivities and the supports of the set  $\overline{\Omega^h}$ .

3: (Figure 8c)

For every support of dimension zero (*i.e.* vertices living on  $\Omega^h$ ), flag the associated shape function as `Full_Shape_Function`. Identify the sets of connected supports with a dimension  $\text{dim} \leq \text{codim}_{\mathcal{T}^h}(\Omega^h)$  among the others. Each set is composed by supports of the same dimension if  $\text{codim}_{\mathcal{T}^h}(\Omega^h) = 1$ , or may be composed by multiple connected groups of supports of dimension one or two if  $\text{codim}_{\mathcal{T}^h}(\Omega^h) = 2$ .

4: (Figure 8d)

**for each** *set of connected supports* **do**

**if** *all the connected supports have the same dimension d* **then**

        └ Decimate a number of shape functions equal to  $d$  on this set of supports.

**else**

        └ Identify the groups of supports of dimension two which are only connected through supports of dimension one or vertices of supports of dimension one.

        └ Decimate two shape functions for the first group of supports of dimension two,

        └ and only one for the following groups.

The decimated shape functions are called `Distributed_Shape_Functions`.

The rest defines the set of `Full_Shape_Functions`.

---



The above example (Figures 7 and 8) only illustrates the Algorithm 1 applied on an embedded line. However, we contend that the algorithm holds with embedded surfaces and show numerical applications in the next section.

#### 4. Numerical validations

The versatility of the proposed approach is shown in the sequel. Problems on manifolds of codimension one (a straight line or a flat surface), and of codimension two (a helix) are solved. The classical convergence rate is computed through convergence studies, by decreasing the bulk mesh size  $h$ . Ill-conditioning caused by small intersections of the domain with the mesh is investigated in the 2D case. Validation with another approach available in the literature is carried out on a sphere. The accuracy of all results is then discussed in relation with classical FEM results using body-fitted meshes. Finally, the robustness of the method is investigated with regard to two alternative methods from the literature.

##### 4.1. Problem of codimension one in 2D - a straight line

Let us start with a Poisson problem of codimension one defined over a straight line:

$$\begin{aligned} -\Delta_{\Omega} u &= f & \text{in } \Omega := ]-1, 1[ \times \{0\}, \\ u &= u_D & \text{on } \Gamma_D := \partial\Omega. \end{aligned} \tag{6}$$

The source term  $f$  and the Dirichlet boundary condition  $u_D$  are defined thanks to an analytical solution given in the 2D Cartesian coordinates by:

$$u(x, y) = (\cos(\pi x) + 1) / 2. \tag{7}$$

Here, the Dirichlet boundary condition is enforced by using a linear combination of the shape functions of the element containing each of the endpoints of the line.

We compare the results obtained with four different approaches on a sequence of meshes with increasing (uniform) density:

- the classical FEM with P1-conforming elements as reference,
- the FE space resulting from the P1-hat shape functions built on the induced discretization,

- the method initially introduced in [53] to build stable Lagrange multipliers for boundary condition enforcement, and used here for the discrete field  $u^h$  as a P0-equivalent space,
- our newly proposed method constructing a P1-equivalent space.

The first non-matching unstructured mesh used in this study is shown in Figure 9 with the related numerical results.

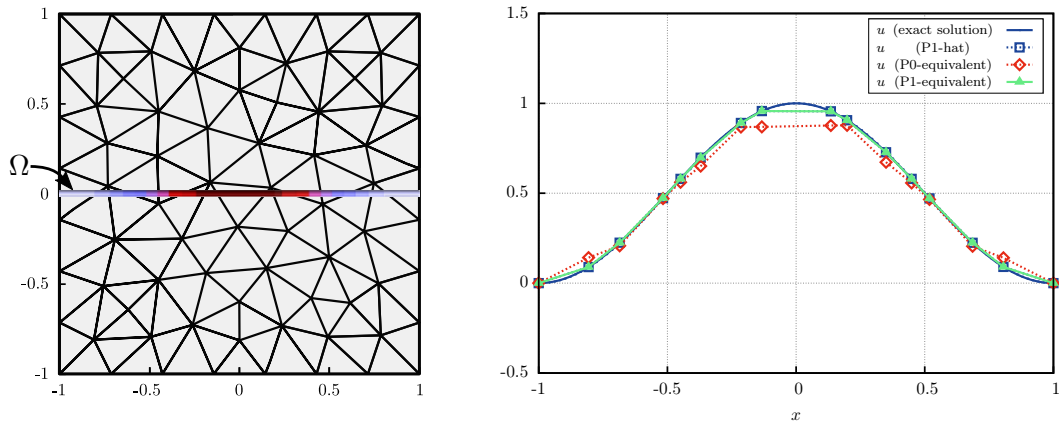


Figure 9: Geometry of the computational domain (left) and solutions of the Poisson problem of codimension one in 2D (right) related to the coarsest discretization.

The P0-equivalent space is not interpolating at some nodes. More precisely some intermediate nodes, called "non-vital" in the algorithm of [53], have their value fixed by the topology of the mesh. This locking is not present with the proposed P1-equivalent space, which gives equivalent results to the P1-hat space, since their basis are constructed as linear combinations. We give in Figure 10 the convergence results.

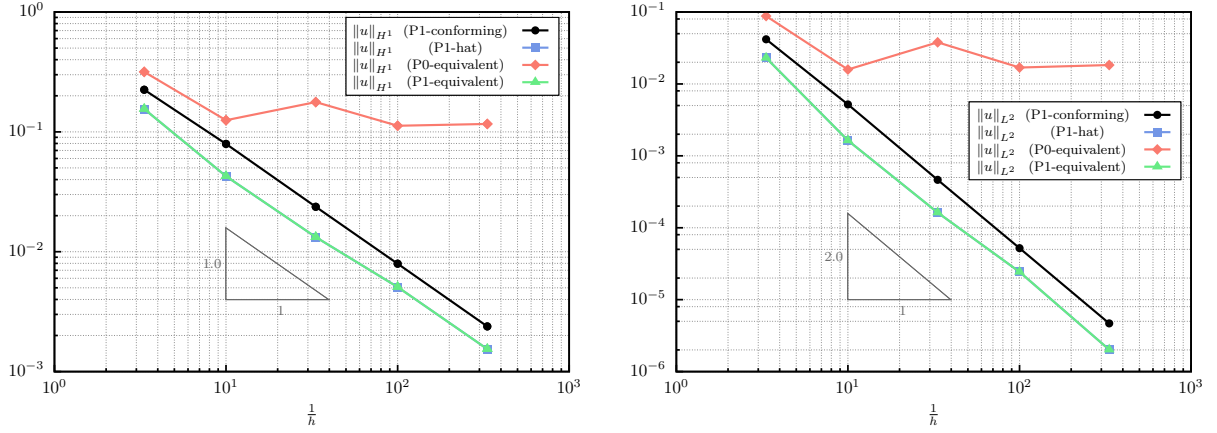
(a) Error in the  $H^1$ -norm of the solution.(b) Error in the  $L^2$ -norm of the solution.

Figure 10: Convergence study for the Poisson problem solved on a straight line of codimension one embedded in a 2D bulk mesh.

As suggested by the preliminary results, the errors obtained with the P0-equivalent space do not converge on the  $H^1$ - and  $L^2$ -norms. Furthermore, the results obtained with the P1-hat space are indistinguishable from that of the P1-equivalent space. The performance of the P1-equivalent space appears to be in good agreement with the P1-conforming case, and yields optimal rates of convergence in both norms.

#### 4.2. Problem of codimension one in 3D - a flat surface

Let us consider a Laplace problem defined over a unit-square embedded in a 3D bulk mesh. It has already been studied in [55, 51, 53] with a surface of codimension zero embedded in a 2D mesh. Here, the surface has a codimension equal to one. The governing equations and the boundary conditions are detailed below:

$$\begin{aligned}
 -\Delta_{\Omega} u &= 0 & \text{in } \Omega &:= ]0, 1[ \times ]0, 1[ \times \{0\}, \\
 u &= u_D & \text{on } \Gamma_D &:= \{0\} \times ]0, 1[ \times \{0\}, \\
 \nabla_n u &= t_N & \text{on } \Gamma_N &:= \partial\Omega \setminus \overline{\Gamma_D}.
 \end{aligned} \tag{8}$$

The boundary conditions are defined according to the exact solution given by the analytical expression:

$$u(x, y, z) = [\cosh(\pi x) - \coth(\pi) \sinh(\pi x)] \sin(\pi y). \tag{9}$$

Illustrations of the embedding of the domain in a typical non-conforming mesh and the exact solution are shown in Figure 11.

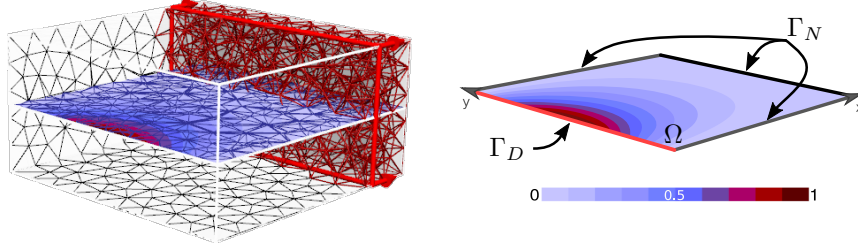


Figure 11: Geometry of the computational domain and exact solution of the unit-square Laplace problem of codimension one in 3D, with a Dirichlet condition defined on a boundary of the mesh.

In order to enforce the Dirichlet boundary condition along the line  $\Gamma_D$  not necessarily matched by the mesh, we introduce Lagrange multipliers. Let  $u|_{\Gamma_D} \in H^{1/2}(\Gamma_D)$  be the primal field defined in the trace space of  $\mathcal{U}$  restricted to  $\Gamma_D$ . Let  $\lambda \in \mathcal{L} := H^{-1/2}(\Gamma_D)$  be the Lagrange multipliers belonging to the dual space of  $H^{1/2}(\Gamma_D)$ . The new weak formulation is given by:

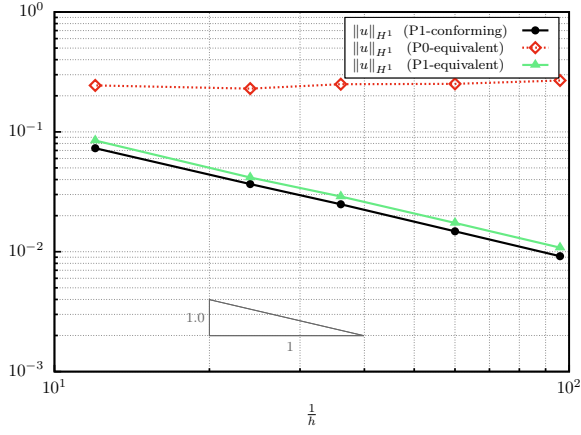
Find  $(u, \lambda) \in \mathcal{U} \times \mathcal{L}$  :

$$\begin{aligned} \int_{\Omega} \nabla_{\Omega} u \cdot \nabla_{\Omega} v \, d\Omega - \int_{\Gamma_D} \lambda v|_{\Gamma_D} \, d\Gamma &= \int_{\Omega} f v \, d\Omega + \int_{\Gamma_N} t_N v \, d\Gamma, \quad \forall v \in \mathcal{U}; \\ - \int_{\Gamma_D} \mu u|_{\Gamma_D} \, d\Gamma &= - \int_{\Gamma_D} \mu u_D \, d\Gamma, \quad \forall \mu \in \mathcal{L}. \end{aligned} \tag{10}$$

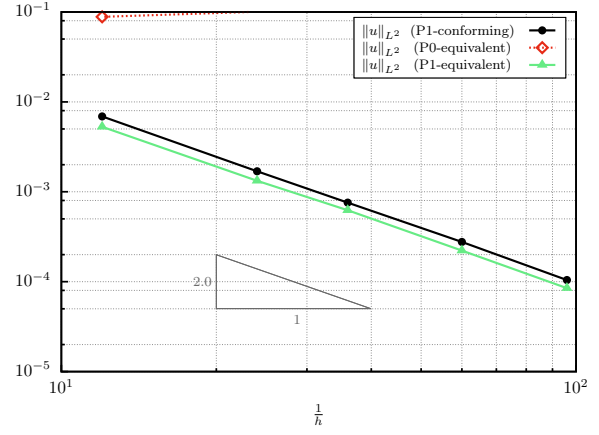
Here, the stable Lagrange multiplier space introduced in [53] will be used. This allows to deplete the Lagrange multipliers to satisfy the inf-sup condition associated within the framework of a mixed formulation. Unfortunately, the associated algorithm is only available for a boundary of dimension  $(n - 1)$  embedded in a mesh of dimension  $n$ , *i.e.* a boundary of codimension one. Here, the surface of codimension one has boundaries of codimension two. Therefore, we will consider only cases where the boundaries are conforming with the

mesh own boundaries. In this way, the 1D boundary  $\Gamma_D$  will be supported by faces of the discretization. Lagrange multipliers are not defined in codimension two, but simply in codimension one on these 2D elements. This makes it possible to apply directly the same algorithm. In order to address the general problem, another algorithm is needed: this will be the topic of a follow-up article.

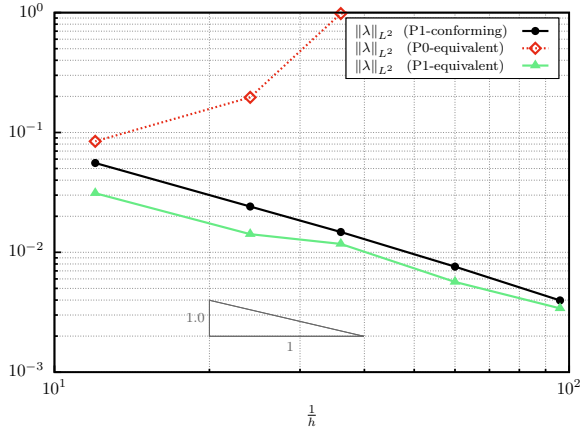
In this paper, we are interested in the relative performances of several function spaces for  $u^h$  exclusively. As in the previous section, three different approaches (P1-conforming, P0-equivalent and P1-equivalent) are considered to define the primal space; while keeping the same Lagrange multiplier space as in [53]. We give in Figure 12 the results of a convergence study.



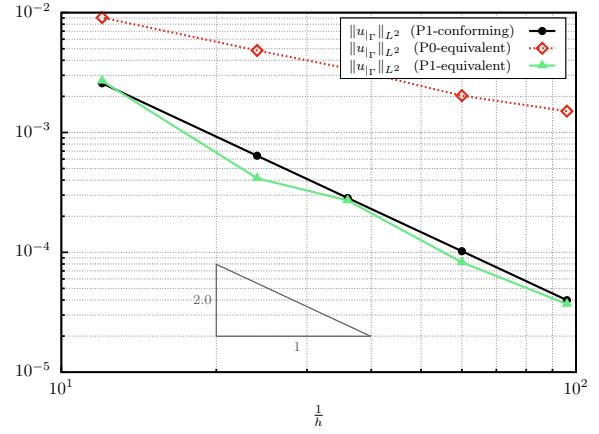
(a) Error in the  $H^1$ -norm of the solution.



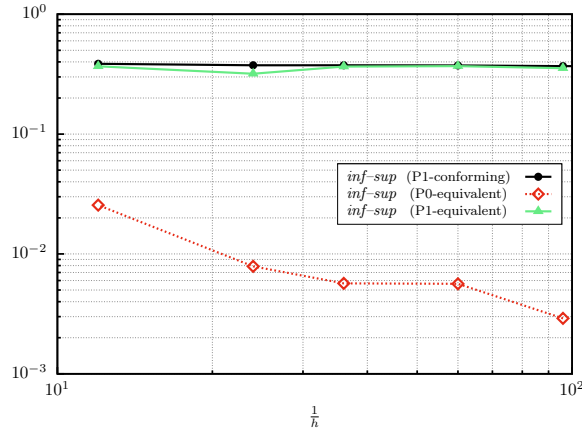
(b) Error in the  $L^2$ -norm of the solution.



(c) Error in the  $L^2$ -norm on the interfacial flux.



(d) Error in the  $L^2$ -norm on the interfacial field.



(e) Value of the inf-sup parameter.

Figure 12: Convergence study for the Laplace problem solved on a unit-square domain of codimension one in 3D, with a Dirichlet condition defined on a boundary of the mesh.

The error in the  $H^1$ - and  $L^2$ -norms decay optimally with the P1-equivalent space as shown in Figures 12a and 12b. The P0-equivalent space does not get convergent results as expected. Figures 12c and 12d present the error in the  $L^2$ -norm along the interface  $\Gamma_D$  for both the flux (given by Lagrange multipliers) and the field. The proposed approach gives comparable results with the conforming case on this boundary. In Figure 12e, results of a numerical inf–sup test proposed in [56] are shown, allowing to check the compatibility between the primal field and the dual field of Lagrange multipliers. The inf–sup test considers a constant — the solution of a eigenvalue problem — that should be mesh-independent, otherwise the inf–sup condition is not satisfied. Previous applications of this test on stable Lagrange multipliers with non-conforming mesh and further details of implementation can be found in [51, 53] for conciseness. While using a stable Lagrange multiplier space defined on  $\Gamma_D$  to discretize the dual field, the use of a P0-equivalent space as primal field does not satisfy the inf–sup condition. However, the P1-equivalent and P1-conforming spaces used for the primal field provide a similar inf–sup constant and are compatible.

#### 4.3. Problem of codimension two in 3D - a helix

We consider again the Poisson problem introduced in Sect. 4.1, but the geometry of the domain is now a helix explicitly defined by the parametric equation:

$$l(t) = [\sin(2\pi t)/2, \cos(2\pi t)/2, t]. \quad (11)$$

The problem is posed as follows:

$$\begin{aligned} -\Delta_{\Omega} u &= f & \text{in } \Omega := \{(x, y, z) \in \mathbb{R}^3, \exists t \in ]-1, 1[ : (x, y, z) = l(t)\}, \\ u &= u_D & \text{on } \Gamma_D := \partial\Omega. \end{aligned} \quad (12)$$

The solution may be expressed in the 3D Cartesian coordinates by:

$$u(x, y, z) = (\cos(\pi z) + 1) / 2. \quad (13)$$

In comparison with the two previous cases, the discretized problem involves a largest gap of dimension between the domain  $\Omega^h$  and the background mesh  $\mathcal{T}^h$ . In Figure 13, we show the embedding of the helix in a non-matching structured mesh using an explicit representation,

and the exact solution. Please note that the 3D aspect helps displaying the solution, it is not the actual geometry of the helix which has zero thickness.

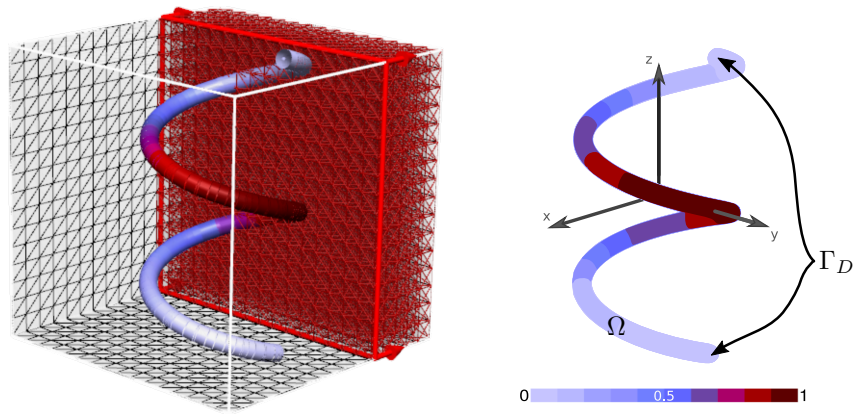
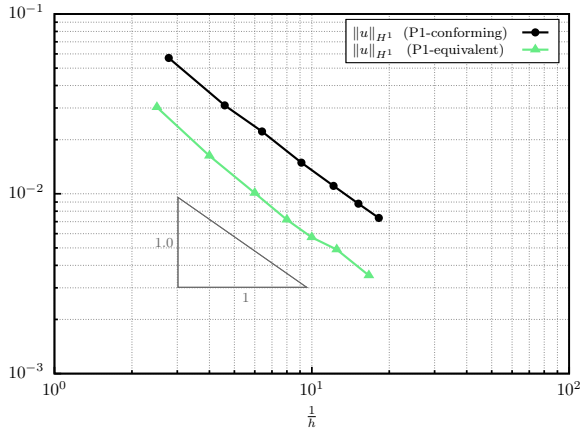
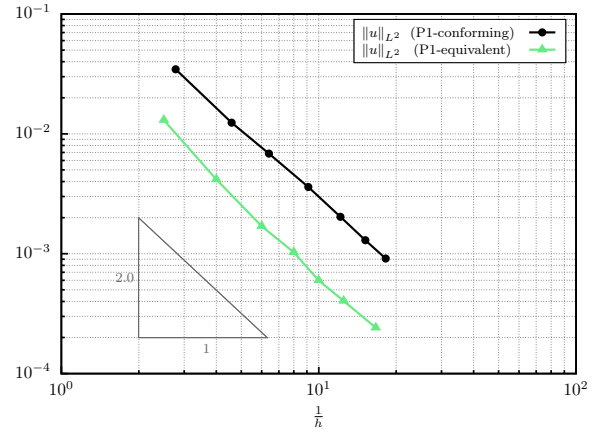
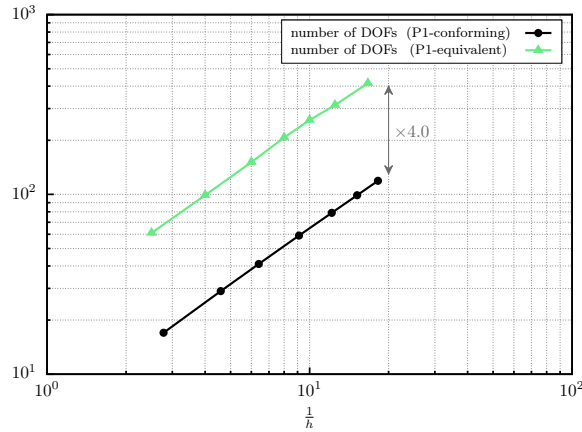


Figure 13: Geometry of the computational domain and exact solution of the helix Poisson problem of codimension two in 3D.

A convergence study is performed in a similar way to previous studies. However, the  $P_0$ -equivalent space is not available in this case (codimension two) as explained in the comment made on the stable Lagrange multipliers in Sect. 4.2. The results of convergence are given in Figure 14.



(a) Error in the  $H^1$ -norm of the solution.(b) Error in the  $L^2$ -norm of the solution.

(c) Number of DOFs.

Figure 14: Convergence study for the Poisson problem solved on a helix of codimension two embedded in a 3D bulk mesh.

Figures 14a and 14b show errors in  $H^1$ - and  $L^2$ -norms. On both, the P1-equivalent space achieves optimal rates of convergence. At first glance, the error generated with the FEM using 1D elements along the line seems greater. However, a more detailed analysis must be conducted with respect to the number of DOFs associated with the problem, see Figure 14c. Indeed, for a same element size  $h$ , the number of DOFs obtained with an embedded curve in a 3D grid is far greater than that obtained with a 1D conforming mesh — by a factor of about 4.0 in codimension two.

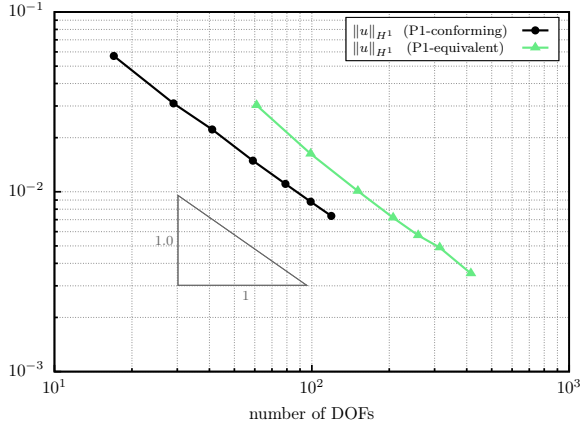
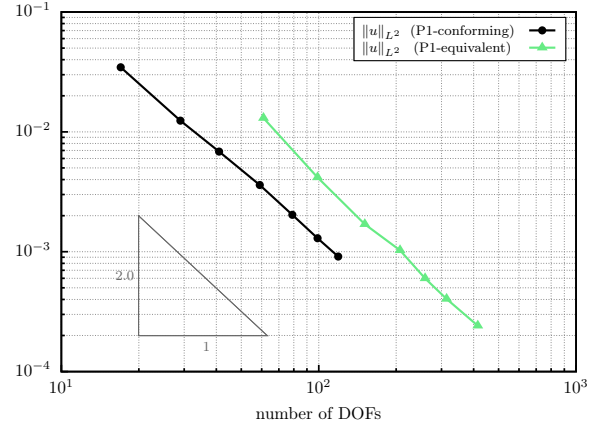
(a) Error in the  $H^1$ -norm of the solution.(b) Error in the  $L^2$ -norm of the solution.

Figure 15: Results of the convergence study as functions of the number of DOFs.

Figures 15a and 15a reveal, not surprisingly, that the P1-conforming space provides results in  $H^1$ - and  $L^2$ -norms with a better accuracy than the P1-equivalent space, for a given number of DOFs.

#### 4.4. Condition number of the problem

Owing to a mismatch between the mesh and the domain of integration, intersections leaving very small supports within elements may arbitrarily occur. If one designs a function space over it without caution, the resulting matrix of the finite element system may be ill-conditioned. Specific treatments of the linear algebraic system of equations may be possible by a preconditioning approach [57]. This problem has also been addressed with stabilised formulations [41, 42].

In both cases, problems of codimension two have not been investigated. Clearly the design of an *a priori* stable method is attractive because it does not pollute the original formulation with additional terms. Therefore, the quality of the approximation depends only on geometrical parameters. It also allows us to use off-the-shelf solvers without any algebraic treatment. Problems involving non-linearities or multi-dimensional coupling are some of the potential applications which may benefit from this.

Two approaches can be considered to control the conditioning of the resulting linear system:

- by modifying the geometry. At each vertex of the elements supporting the embedded domain, the distance  $d^h$  to  $\Omega^h$ , as measured in the reference element, is computed. If  $d^h$  is less than a given parameter  $\tau^h$ , then we force the distance to be equal to zero. An illustration on two cases with different mesh configurations is shown in Figure 16. The initial geometry is an embedded curve (dash line), that is nearly intersecting one of the vertices of the mesh at the top of the figure. In this case, the line is forced to pass through the vertex. This approach was first proposed with the X-FEM in [58], taking advantage of the flexibility given by using level-set functions. Although this method is straightforward to implement, the geometry of the domain problem is, however, slightly modified. This increases the error on the geometric approximation. It can cause some oscillations in convergence studies, without changing the convergence rate.
- by modifying the function space. Here, the DOF of the vertex is eliminated when  $d^h < \tau^h$ . The naive solution of simply removing the DOF does not preserve the partition-of-unity. A linear combination with the other active DOFs of the element must therefore be applied. The geometry remains unchanged. This approach stays in line with [53]. It does not create any new independent group of shape functions, but the linear combinations which are defined can reduce the kernel. However, the approximation may lose the ability to represent a linear field in the vicinity of the linear combination. An illustration of this problem is shown in case 2 of Figure 16 for instance, where linear combinations between four DOFs define only a constant value over two small sub-elements. Since this only happens on a very small area for  $\tau^h$  small enough, it has no effect on the convergence rate. This approach was studied in [54].

Approach modifying:	case 1	case 2
the geometry		
the space		

Figure 16: Solutions of the nearly-matching-mesh problem with both approaches: by modifying the geometry or the space.

In this paper, the approach modifying the geometry has been implemented. Let us mention, however, that any other approach may be used with the new function space designed in this paper, without trouble.

In order to compare the level of conditioning with, and without specific control, a simple problem is considered: the one introduced in Sect. 4.1. A study is performed to illustrate the effect of the position of the line on the condition number of the resulting linear system. For this, a straight line is embedded in a structured-triangular mesh, one-element thick. Let  $d^h$  the distance between  $\Omega^h$  and the (horizontal) bottom edges of the background mesh. The first mesh of the convergence study is shown in Figure 17.

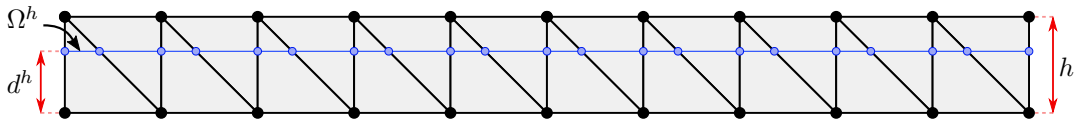
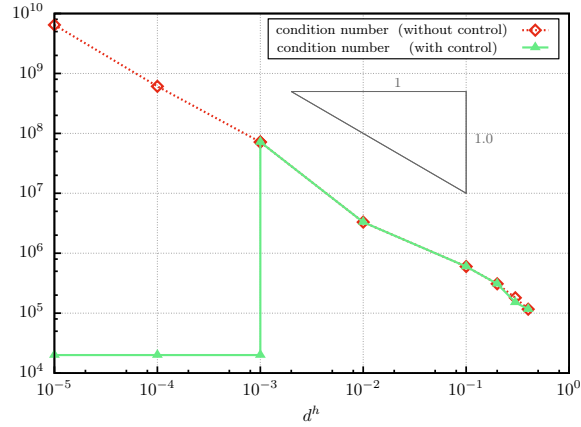
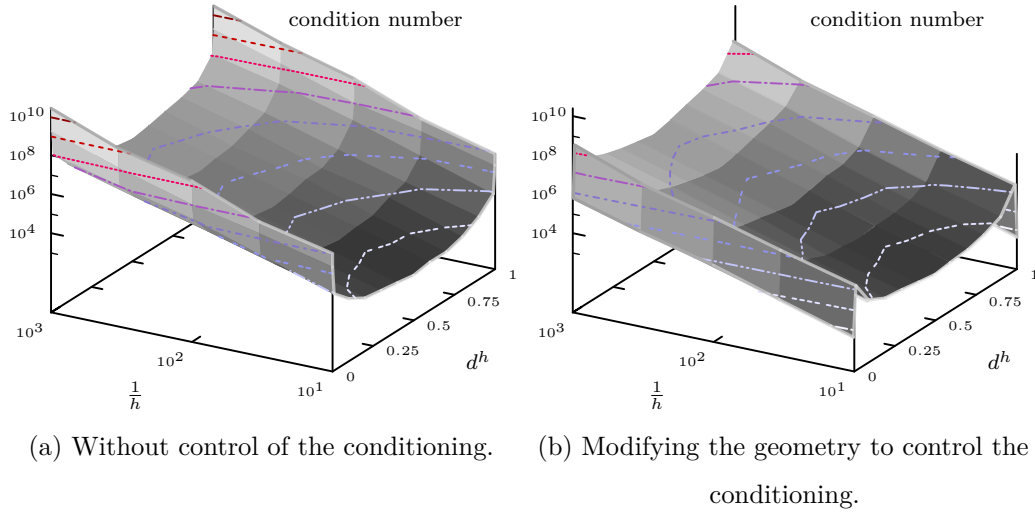


Figure 17: Geometry of the computational domain of the Laplace problem of codimension one in 2D, with parameters of the sensitivity analysis.

We compare the results obtained using the approach modifying the geometry and those obtained doing nothing at all. For the current study, the parameter  $\tau^h$  associated with this approach is equal to  $10^{-3}$ . The value may actually be derived from criteria proposed in [59]. The results of this sensitivity analysis are shown in Figure 18.



(c) Comparative results obtained with  $h = 0.01$  for  $d^h \in ]0, 0.5]$  in log scale.

Figure 18: Convergence surfaces on the condition number of the resulting stiffness matrix vs the position of the embedded line.

Figure 18a reports a progressive increase in the conditioning as the embedded line gets closer and closer to the mesh vertices. The value of the condition number may be bounded by modifying the geometry as shown in Figure 18b. The Figure 18c clearly shows that the unit slope of the condition number in a log-log scale peculiar to this configuration can be reduced to constant using the aforementioned procedure. Furthermore, it can be seen from the analysis of the convergence surfaces as functions of characteristic length  $h$ , that the

contribution to ill-conditioning from small intersections may be uniformly limited by the procedure regardless of  $h$ .

#### 4.5. Validation on another problem of codimension one in 3D - a sphere

The purpose of this validation is twofold:

- (i) compare our approach with the method proposed by Olshanskii *et al.* in [35],
- (ii) confront results obtained with two different approaches to represent the geometry: one using a linear approximation, which was used throughout Sect. 4, and the other, introduced in [60] and used in [35], based on a quadratic approximation.

Let us consider a Poisson problem defined on the unit sphere, which has already been treated in [35]:

$$-\Delta_{\Omega}u = f \quad \text{in } \Omega := \{(r, \theta, \varphi) \in \mathbb{R}^3, r = 1\}. \quad (14)$$

The analytical solution given in the spherical coordinates is as follows:

$$u(r, \theta, \varphi) = 12\sin^3(\theta)\sin(3\varphi), \quad (15)$$

which allows to fully characterize the source term  $f$ . Note that its integral over the domain is zero. This additional condition may be enforced in order to ensure uniqueness of the solution of the problem.

Figure 19 represents the geometry of the problem with the analytical solution.

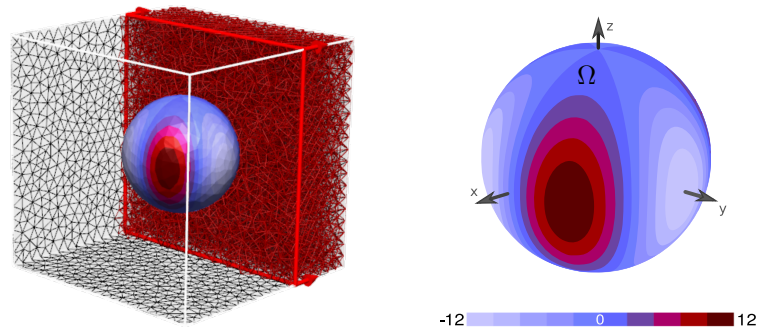
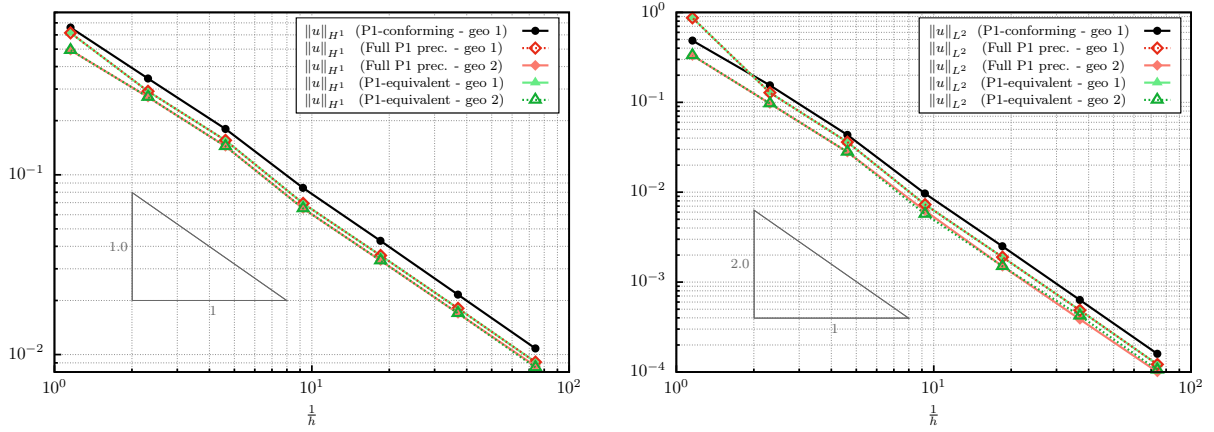


Figure 19: Geometry of the computational domain and exact solution of the unit-sphere Laplace problem embedded in a bulk mesh.

A convergence study is performed and allows to compare three different function spaces:

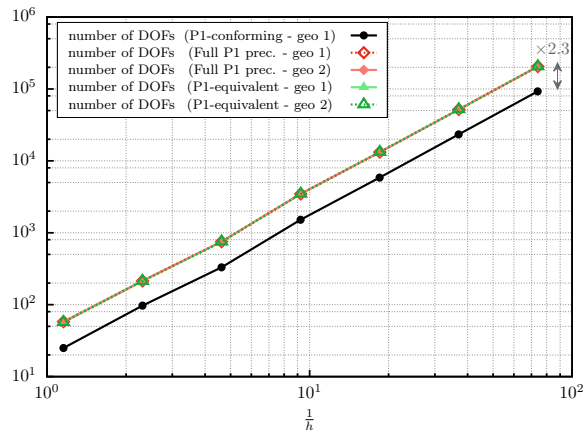
- by using a body-fitted mesh on which a P1-conforming space is built,
- by means of the method introduced by Olshanskii *et al.*, which will be denoted Full P1 prec. hereafter,
- or by constructing a dedicated P1-equivalent space as introduced in this paper.

For the two latter approaches based on non-conforming meshes, two different approximations of the geometry will be used: a first order approximation called geo 1, and a second order used by Olshanskii *et al.* noted geo 2. The associated results are given in Figure 20.



(a) Error in the  $H^1$ -norm of the solution.

(b) Error in the  $L^2$ -norm of the solution.



(c) Number of DOFs.

Figure 20: Convergence study for the Laplace problem solved on a spherical surface embedded in a bulk mesh.

Optimal results are obtained with all procedures on  $H^1$ - and  $L^2$ -norms, see Figures 20a and 20b. The choice of a linear geometric description does not degrade the convergence rate. It only causes a minor additional error on the geometry if we compare geo 1 and geo 2 curves. The choice of a quadratic approximation was initially motivated by a suboptimal convergence rate of the surface tension force in a P2-P1 mixed formulation [60]. It is not essential in the numerical applications presented in this paper. For each kind of geometric approximation, the results obtained with both non-conforming approaches are indistinguishable. We perform an analysis similar to that of Sect. 4.3 in order to check the relative performances of the different spaces. In this case, the number of DOFs associated with an embedded surface is greater than that obtained with a tessellation of the surface (about 2.3 time), see Figure 20c. Figure 21 illustrates how these previous results depend on the number of DOFs.

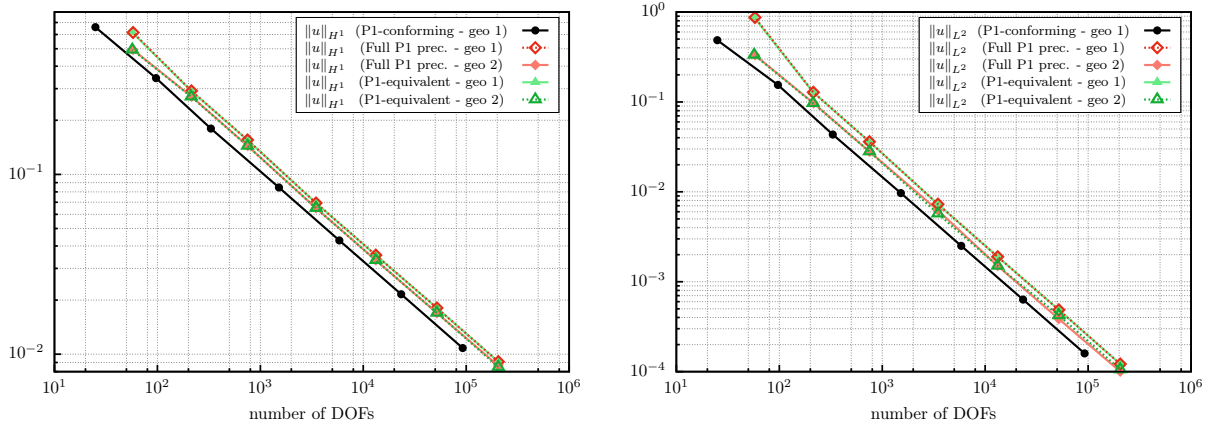
(a) Error in the  $H^1$ -norm of the solution.(b) Error in the  $L^2$ -norm of the solution.

Figure 21: Results of the convergence study as functions of the number of DOFs.

As expected, the error of the FEM is actually slightly lower than that of non-conforming approaches. In conclusion, the proposed approach combined with a linear geometry description provides optimal results, close to those obtained with FEM and equivalent to the solution proposed by Olshanskii *et al.*.



#### 4.6. Analysis of the matrix properties

This last section aims to highlight different settings bringing ill-posed problems, therefore requiring a specific treatment. After a brief presentation, each setting is investigated numerically. This highlights the benefit on robustness of the proposed approach, with regard to other approaches developed in the literature.

For the sake of clarity and simplification, let us simply consider embeddings in 2D structured meshes for our presentation. Let us start with a straight line as illustrated in Figure 22a. A similar example was proposed in [51] (for the design of stable Lagrange multipliers). The resulting discretization of  $\Omega$  is composed of  $2N - 1$  nodes, while more (vertex) shape functions are defined on the mesh ( $2N$ ). This leads to the non-uniqueness in field approximation. Let us now consider another configuration with a closed loop (e.g. a circle) shown in Figure 22b. This setting was also studied in [35]. As in the previous example,  $2N$  shape functions are involved here. However, an equivalent number of nodes is used to discretize the line, due to the cycle in the topology. Nevertheless, there is still no unique way to represent a linear field in this setting, since any opposite constant values associated with  $v_i^+$  and  $v_i^-$  generate an appropriate field.

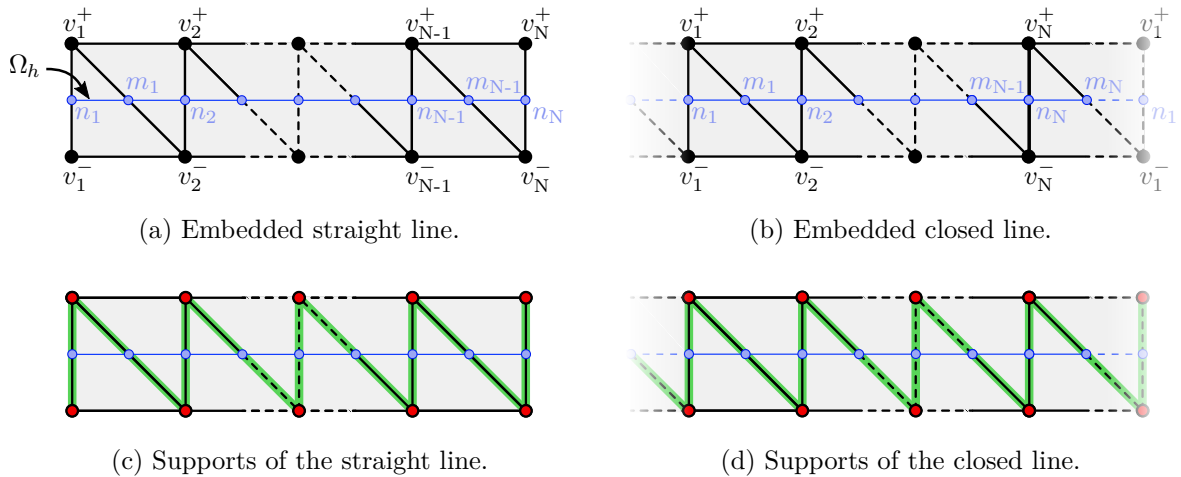


Figure 22: Two embeddings into structured meshes resulting in a mass matrix with one zero eigenvalue.

Figures 22c and 22d show that all 1D supports are connected in one group of independent

shape functions. Applying Algorithm 1 in both cases leads to eliminate one zero eigenvalue in the mass matrix.

Other mesh configurations may lead to more than one zero eigenvalue, e.g. with meshes given in Figures 23a and 23b. The lines (straight or closed) are now going through vertices of the mesh. Both settings have  $3N - 1$  shape functions and  $N$  zero eigenvalues, one per independent 1D support (see Figures 23c and 23d) ( $2N - 1$  nodes along the straight line, and  $2N$  nodes with one cycle for the loop).

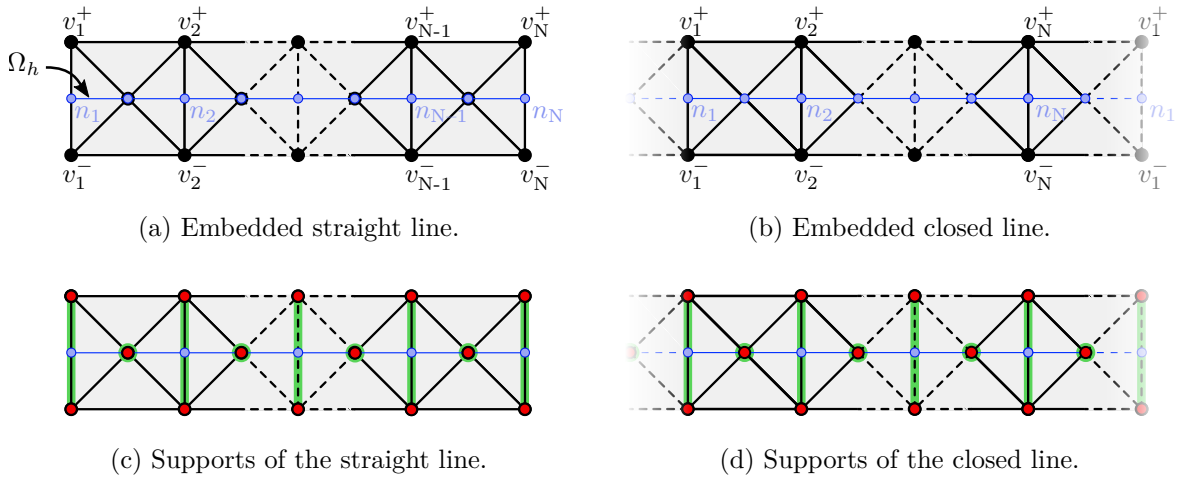


Figure 23: Two embeddings into structured meshes resulting in a mass matrix with  $N$  zero eigenvalues.

From these observations, it is clear that the number of zero eigenvalues in the mass matrix may potentially be greater than one, up to be proportional to the number of elements. Such a case appears when the embedded domain may be divided into disjoint sets separated by conforming vertices, thus leading to independent groups of shape functions.

In the sequel, numerical experiments covering the different settings previously introduced are carried out. The matrix properties are first analysed on the Poisson problem defined on a straight line of codimension one in 2D and introduced in Sect. 4.1. Then, we investigate closed topologies with Poisson problems posed on circles in 2D and 3D (resp. codimension one and two). Finally, an analysis of the eigenvalues obtained with a flat surface embedded in a structured mesh is given.

In order to be exhaustive, we compare results obtained by our methodology with two alternative methods from the literature:

- modifying the variational formulation with a stabilisation term recently introduced by Burman *et al.* [42],
- using a dedicated preconditioned conjugate gradient solver as proposed by Olshanskii *et al.* [35, 57].

We now briefly describe the key features of each of the aforementioned methods.

For the first one, Burman *et al.* propose a full gradient stabilised approach for the Laplace-Beltrami operator. This consists of an additional stabilisation term — with a positive parameter  $\tau$  — which is defined as follows:

$$\tau \int_{\mathcal{T}^h} h \nabla u^h \cdot \nabla v^h \, d\mathcal{T} \quad (16)$$

This term is computed on the entire elements of  $\mathcal{T}^h$  containing  $\Omega$ , which allows to recover a problem of codimension zero. Hence, this tends to bypass the problem of zero eigenvalue (ill-posed problem); the parameter  $\tau$  being selected to prevent ill-conditioned linear system without significant loss of accuracy. Let us denote this approach Full P1 stab. in the sequel.

In the second approach, Olshanskii *et al.* apply diagonal scaling to improve the conditioning properties of the matrices. Let  $[M]$  be the mass matrix and  $[S]$  be the stiffness matrix of the Laplace-Beltrami equation. They introduce the diagonally scaled mass and stiffness matrices, respectively defined by:

$$\begin{aligned} [\tilde{M}] &:= [D_M]^{-1/2} [M] [D_M]^{-1/2}, \quad \text{with } [D_M] := \text{diag}([M]) \quad \text{and } M_{i,j} := \int_{\Omega^h} N_i N_j \, d\Omega; \\ [\tilde{S}] &:= [D_S]^{-1/2} [S] [D_S]^{-1/2}, \quad \text{with } [D_S] := \text{diag}([S]) \quad \text{and } S_{i,j} := \int_{\Omega^h} \nabla_{\Omega^h} N_i \cdot \nabla_{\Omega^h} N_j \, d\Omega. \end{aligned} \quad (17)$$

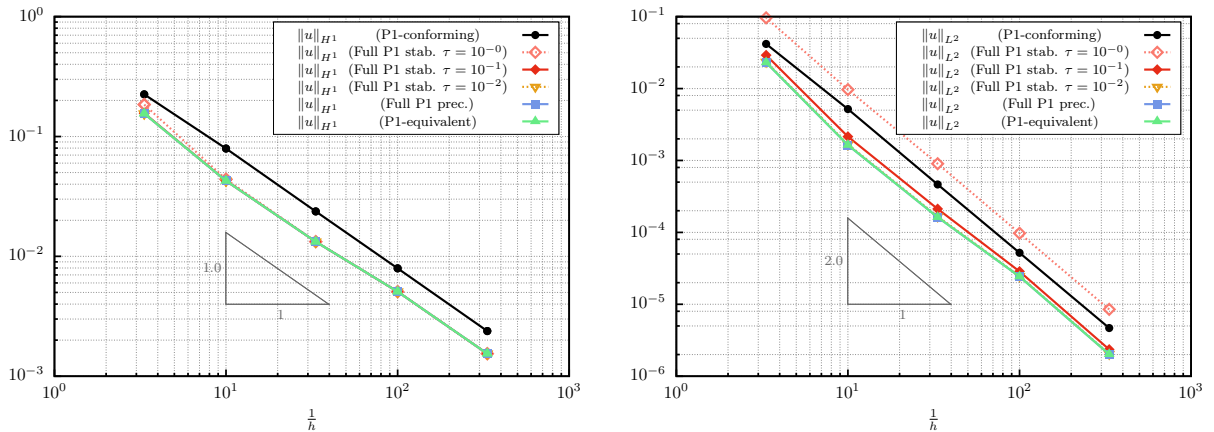
Solving PDEs by means of a Gauss-Seidel preconditioned conjugate gradient solver with a relative tolerance  $t$ , they alleviate difficulties linked to singularities in the linear algebraic system thanks to a judicious choice of this parameter.

Studies on these two approaches have never mentioned—even on a simple 2D case—the potential presence of multiple zero eigenvalues in the matrices, which may affect the robustness of these approaches. For instance, studies reported in [40, 41] identify a single additional zero eigenvalue induced on a closed domain.

Let us investigate these settings through the following numerical experiments. They will outline the interest in building a dedicated function space, which defines *a priori* well-adapted basis from the linear FE shape functions.

*Problem of codimension one in 2D - a straight line*

Let us go back to the 2D case study introduced in Sect. 4.1. In order to compare the matrix properties of the three non-conforming approaches previously introduced (Full P1 stab., Full P1 prec. and P1-equivalent), the analysis of the eigenvalues will be systematically performed. We begin by analysing the convergence of these approaches, and select the value of the parameter  $\tau$  involved within the stabilised approach. This stabilisation parameter must be chosen sufficiently large to stabilise the approach, without significantly degrading the convergence of the solution. Figure 24 shows the convergence results obtained with the different approaches.



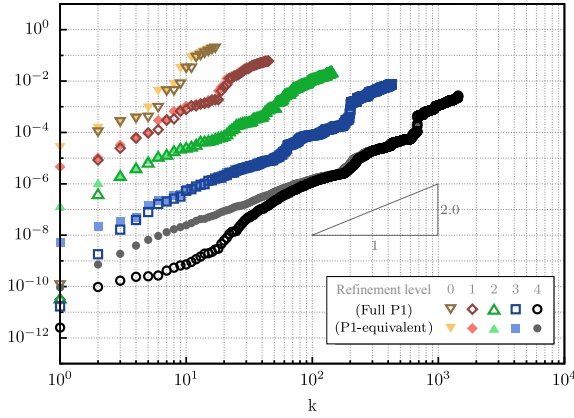
(a) Error in the  $H^1$ -norm of the solution.

(b) Error in the  $L^2$ -norm of the solution.

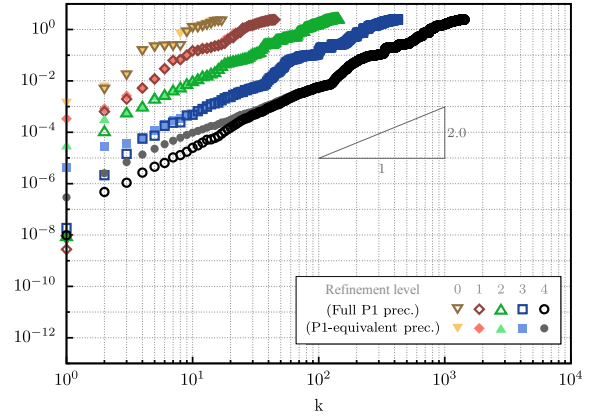
Figure 24: Identification of the stabilisation parameter  $\tau$  through a convergence study for the Poisson problem solved on a straight line embedded in a 2D unstructured mesh.

The three non-conforming approaches lead to almost indistinguishable errors in the  $H^1$ -norm (see Figure 24a). From the Figure 24b, we select the stabilisation parameter  $\tau = 10^{-1}$  as a sufficiently small value to preserve good accuracy of the results. In the sequel, we will show that a lower value of the parameter does not stabilise the problem correctly.

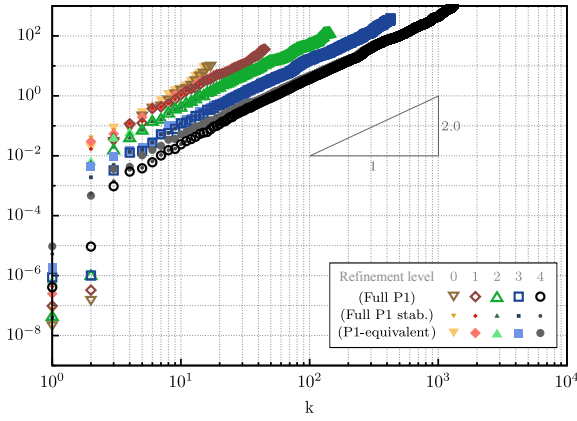
In line with Olshanskii and Reusken [57], we analyse the distribution of eigenvalues for the different approaches. For the mass matrix, the Full P1 and the P1-equivalent approaches are investigated. For the stiffness matrix, we consider the Full P1, the Full P1 stab. and the P1-equivalent approaches. In both cases, the approaches will also be investigated with diagonal preconditioning (identified by the suffix prec.). In Figure 25, the related results are summarized.



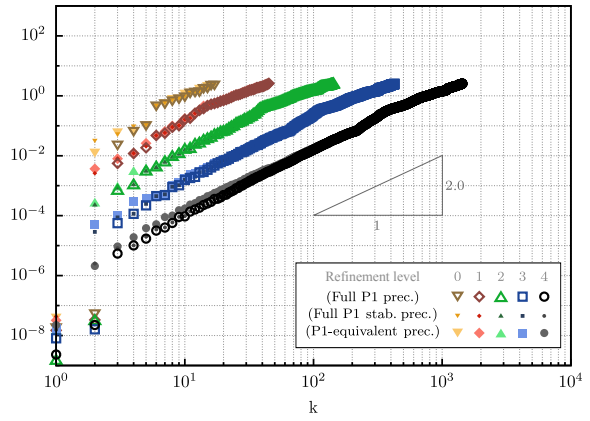
(a) Mass matrix eigenvalues without scaling.



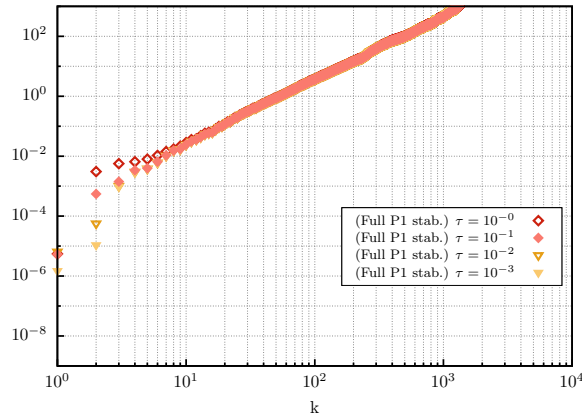
(b) Mass matrix eigenvalues with scaling.



(c) Stiffness matrix eigenvalues without scaling.



(d) Stiffness matrix eigenvalues with scaling.



(e) Parametric study of the eigenvalues obtained with the finest mesh.

Figure 25: Eigenvalue analysis for the Laplace problem solved on a straight line of codimension one embedded in a 2D bulk mesh.

The general distribution of the eigenvalues behaves in  $O(k^2)$ . The Full P1 approach presents one zero eigenvalue for the mass matrix. This persists after scaling. This number is reduced to zero using the P1-equivalent approach. For the stiffness matrix, the Full P1 approach also has one zero eigenvalue more than the others. Note that a smaller value of the stabilisation parameter  $\tau$  (cf. Figure 25e) leads to higher condition number. This supports our previous choice.

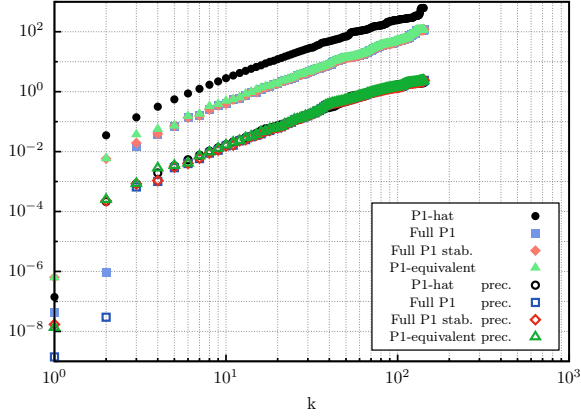
For high refinement levels (3 and 4), we observe some worse eigenvalues by using the Full P1 approach than with P1-equivalent one (see e.g. Figure 25e). Analysis of the mesh intersection reveals that some nodes of the induced discretization of  $\Omega$  are almost coincident with vertices of the meshes. Applying a small perturbation on the meshes, we recover conformity of these few vertices and test the robustness of the approaches. Let us denote  $v \in \Omega^h$  a vertices considered as being put of the line. We give in Table 1 details on the discretization and the zero eigenvalues.

Table 1: Details on the zero eigenvalues of the eigenvalue analysis in 2D.

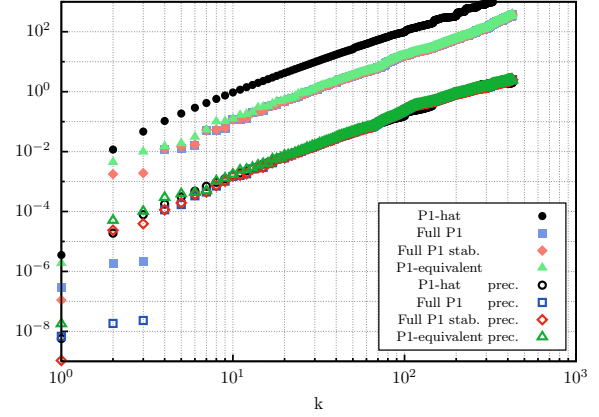
Refinement level	Number of DOFs			$v \in \Omega^h$	Mass matrix zero eigenvalues			Stiffness matrix zero eigenvalues		
	P1-hat	Full P1	P1-eq.		P1-hat	Full P1	P1-eq.	P1-hat	Full P1	P1-eq.
2	142	143	142	0	0	1	0	1	2	1
3	424	426	424	1	0	2	0	1	3	1
4	1414	1424	1414	9	0	10	0	1	11	1

The P1-hat function space — directly built on the induced discretization — is considered as reference. It does not generate zero eigenvalue for the mass matrix and has only one for the stiffness matrix (similar to a rigid body motion). Here, three different configurations may be distinguished. For the refinement level 2, a single set of connected 1D supports is defined, which generates one additional zero eigenvalue within the Full P1 approach. For the refinement level 3,  $\Omega^h$  intersects the mesh in one vertex. It results two groups of independent shape functions defined on 1D supports, with two additional zero eigenvalues within the Full P1 approach. For the refinement level 4, ten zero eigenvalues are added within the Full P1 approach, due to nine vertices living on  $\Omega^h$ . In any case, the P1-equivalent approach cancels these additional zero eigenvalues and recovers matrix properties similar to those of

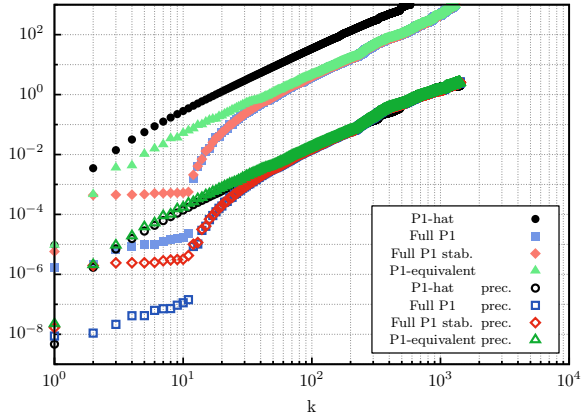
the P1-hat approach. This is illustrated in Figure 26.



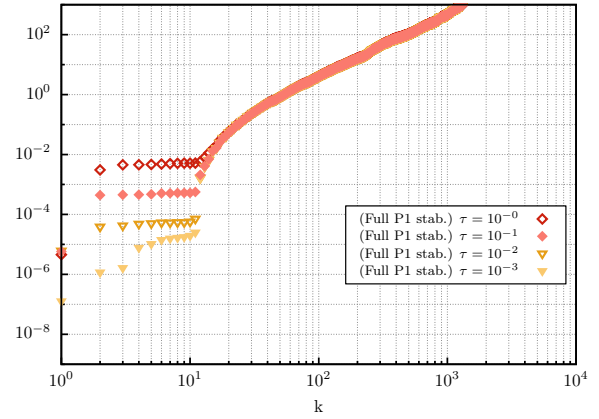
(a) Eigenvalues obtained with  $\frac{1}{h} = 65$ .



(b) Eigenvalues obtained with  $\frac{1}{h} = 198$ .



(c) Eigenvalues obtained with  $\frac{1}{h} = 665$ .



(d) Parametric study of the eigenvalues obtained with the finest mesh.

Figure 26: Detailed comparison of the stiffness matrix eigenvalue analysis in 2D.

Only the newly proposed approach seems to preserve the  $O(k^2)$  behaviour in high refinement levels. Within the other approaches, multiple small eigenvalues appear.

#### *Problem of codimension one in 2D - a circle*

Let us now focus on closed domains. We consider the setting illustrated in Figure 22b, involving a closed line (e.g. a circle) embedded in a structured mesh. The discretization is made of 32 linear triangles. The results of an eigenvalue analysis involving the previously introduced approaches are given Figure 27.



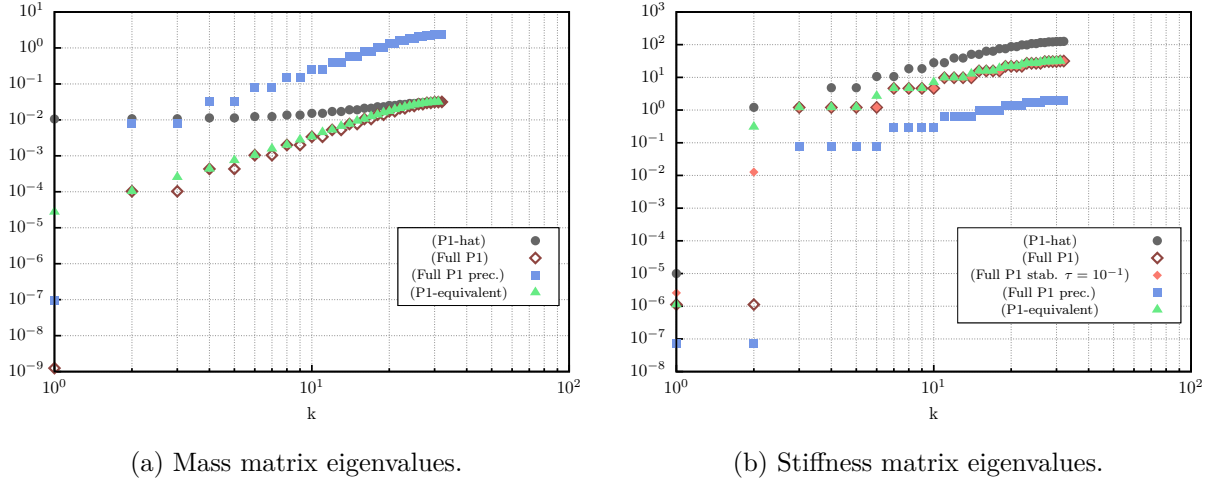


Figure 27: Eigenvalue analysis for the Laplace-Beltrami equation posed on a circle of codimension one embedded in a 2D structured mesh.

Comparing with the P1-hat approach, the Full P1 presents one additional zero eigenvalue in both matrices. Applying preconditioning technique does not reduce this number, while stabilisation technique seem to remedy to it, as shown in Figure 27b. Furthermore, the P1-equivalent approach addresses this issue by "cleaning" the kernel, which avoids propagation of numerical error and leads to a better condition number.

We can conclude that the number of zero eigenvalues is not directly depending on the topology of the domain, but it is rather linked to the number of independent groups of shape functions. We show in the next section that it also depends on the codimension of the embedding.

#### *Problem of codimension two in 3D - a circle*

Let us investigate the previous setting in codimension two (*i.e.* in 3D). We consider a circle of radius 1.8 embedded in a structured mesh composed of unit-cubes divided into six tetrahedra. Figure 28 represents the discretization and the embedded domain.

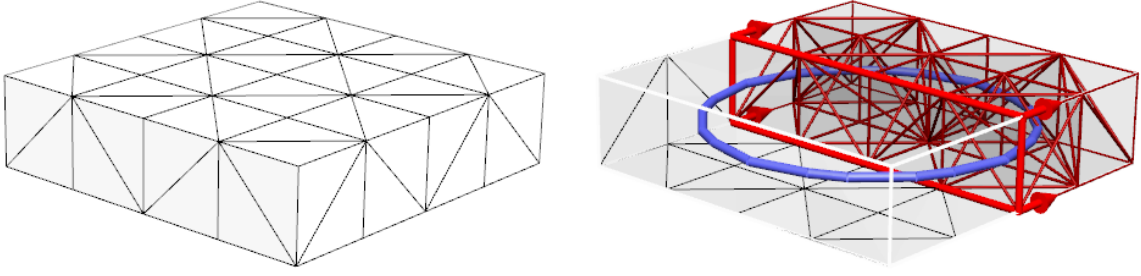


Figure 28: 3D Structured mesh and embedded geometry of the circular computational domain of codimension two.

We give in Figure 29 the results of the eigenvalue analysis.

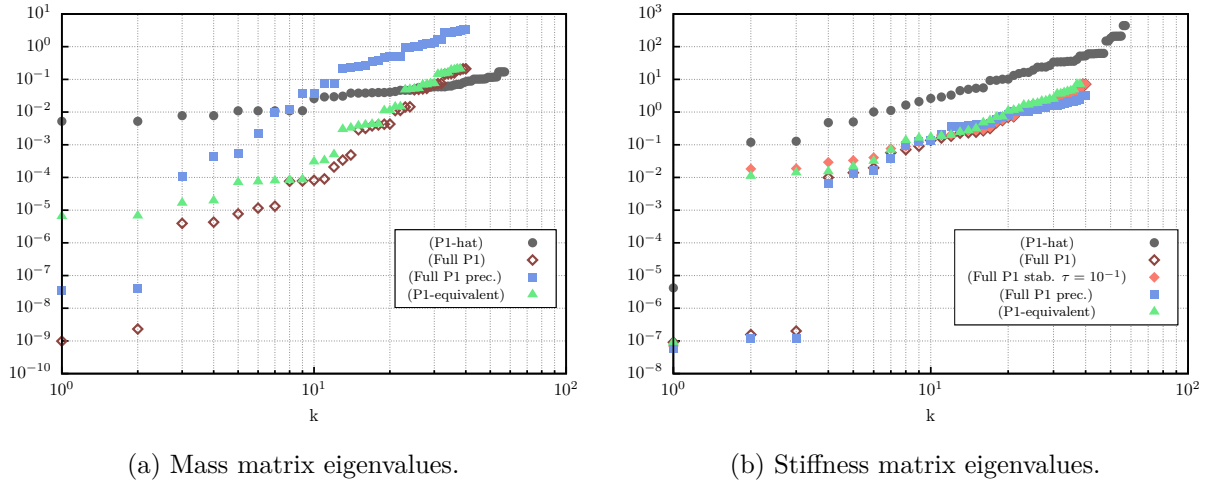


Figure 29: Eigenvalue analysis for the Laplace-Beltrami equation posed on a circle of codimension two embedded in a 3D structured mesh.

Once again, additional vanishing eigenvalues (two more) are introduced by the Full P1 approach in comparison with P1-hat approach. The P1-equivalent approach allows to reduce this number, with similar effect to the Full P1 stab. approach, but without introducing any additional parameter.

#### *Problem of codimension one in 3D - a flat surface*

The aim of this section is to illustrate, for a flat surface embedded in a 3D background mesh, that an arbitrarily large number of vanishing eigenvalues may be involved and altered

the matrix properties. We consider the Laplace equation within the setting depicted in Figure 30.

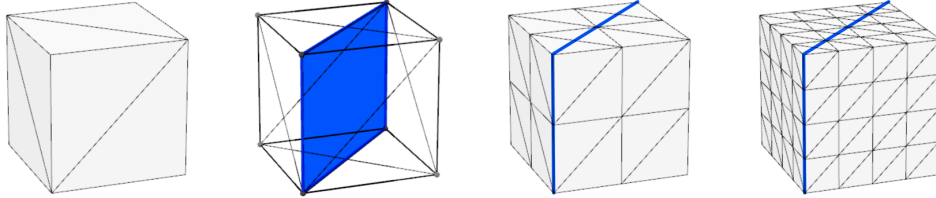
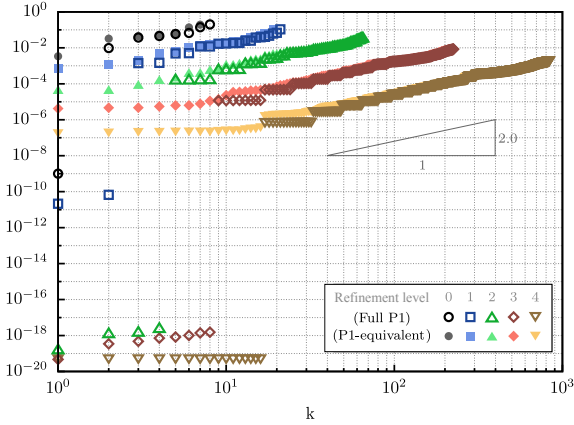
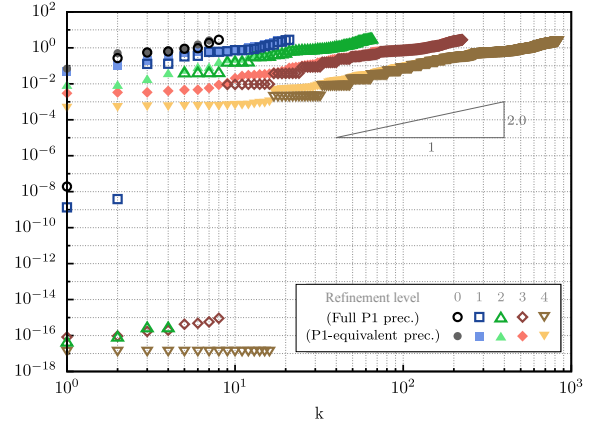


Figure 30: 3D structured meshes with an embedded flat surface; from left to right: initial mesh, embedding of the surface into the mesh of refinement level 0, 1 and 2.

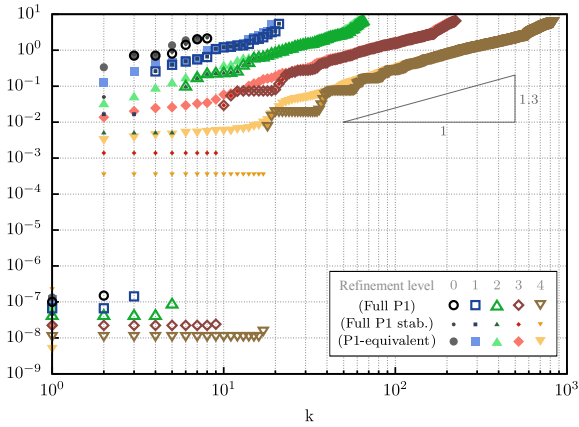
Following the same scheme as for the straight line in 2D, we compare the distribution of eigenvalues for the different approaches (see Figure 31, using  $\tau = 10^{-1}$ ).



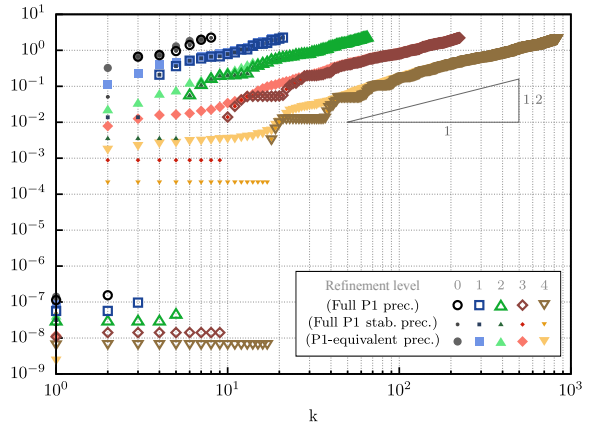
(a) Mass matrix eigenvalues without scaling.



(b) Mass matrix eigenvalues with scaling.



(c) Stiffness matrix eigenvalues without scaling.



(d) Stiffness matrix eigenvalues with scaling.

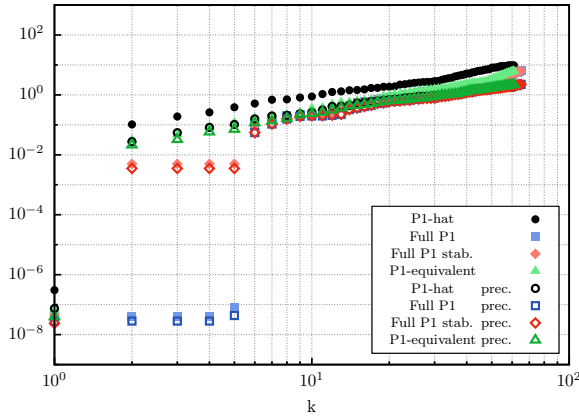
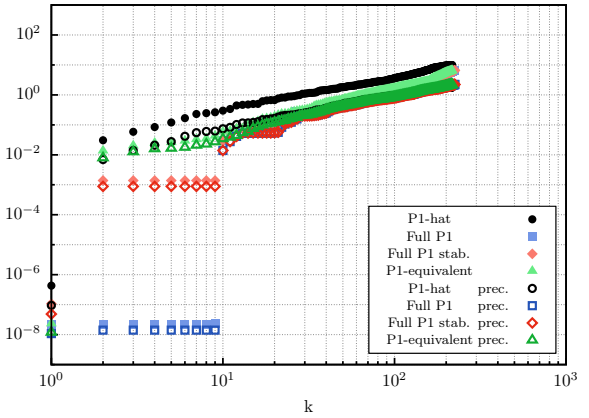
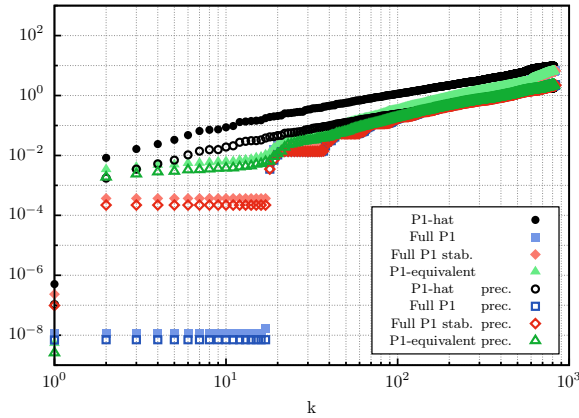
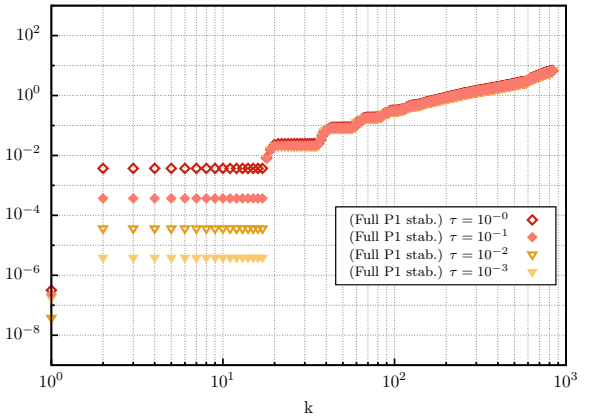
Figure 31: Eigenvalue analysis for the Laplace problem solved on a flat surface of codimension one embedded in a 3D structured mesh.

We can observe that the distributions of the eigenvalues evolve globally in  $O(k^2)$  for the mass matrix and close to  $O(k)$  for the stiffness matrix. The few first eigenvalues are quite small for the Full P1. This number increases with the refinement level. This issue is not addressed by diagonal scaling; it depends on the parameter  $\tau$  of the stabilisation, while the proposed approach is effective straight out of the box. Table 1 gives more details on the zero eigenvalues.

Table 2: Details on the zero eigenvalues of the eigenvalue analysis in 3D.

Refinement level	Number of DOFs			Mass matrix zero eigenvalues			Stiffness matrix zero eigenvalues		
	P1-hat	Full P1	P1-eq.	P1-hat	Full P1	P1-eq.	P1-hat	Full P1	P1-eq.
0	7	8	7	0	1	0	1	2	1
1	19	21	19	0	2	0	1	3	1
2	61	65	61	0	4	0	1	5	1
3	217	225	217	0	8	0	1	9	1
4	817	833	817	0	16	0	1	17	1

Starting with one additional zero eigenvalue for the Full P1 approach, each refinement level doubles this number. This is due to the structure of the intersection between  $\Omega$  and the mesh, which contains a number of independent groups of shape functions proportional to the value of refinement level. The P1-equivalent approach removes linear dependent shape functions and recovers matrix properties similar to the P1-hat approach (see Figure 32).

(a) Eigenvalues obtained with  $\frac{1}{h} = 4$ .(b) Eigenvalues obtained with  $\frac{1}{h} = 8$ .(c) Eigenvalues obtained with  $\frac{1}{h} = 16$ .

(d) Parametric study of the eigenvalues obtained with the finest mesh.

Figure 32: Detailed comparison of the stiffness matrix eigenvalue analysis in 3D.

## 5. Conclusion

In this paper, we introduced a P1-equivalent space to solve elliptic problems defined on embedded solids within the context of the extended finite element method. The main results may be summarized as follows:

1. The proposed approach is general and handles any embedding in a single framework. Its originality is to build an *a priori* well-adapted space, preserving thereby the variational formulation and the ability to use off-the-shelf solvers. A new algorithm, based on

concepts suitable for each dimension, is introduced to reduce the full P1 vertex spaces defined on the background meshes by using linear combinations.

2. Numerical validations on several examples with analytical solutions and covering all the codimensions involved are presented. The accuracy of the results is discussed by taking the classical FEM with body-fitted mesh as a reference. Quadratic rates of convergence in the  $L^2$ -norm and linear rates of convergence in the  $H^1$  are achieved by the P1-equivalent space in all configurations. Specific treatments are investigated in a sensitivity analysis to take care of the conditioning of the linear system.
3. A comparative study is carried out with another method available in the literature on a borderless domain. This method, proposed by Olshanskii *et al.* [35], uses a Conjugate Gradient solver with specific preconditioning. Our approach alleviates the resolution routines from ill-posed algebraic system of equations. Therefore, it is possible to use any type of solver. For instance, when using Lagrange multipliers, the new system is no longer definite positive, thus preventing the use of Conjugate Gradient solvers.
4. Detailed eigenvalue analyses are performed for different settings of topology or codimension and compared with two alternatives: the methodology of Olshanskii *et al.* and the full gradient stabilised method proposed by Burman *et al.* [42]. The proposed approach dealt with the zero eigenvalues issue and shown robustness without any stabilisation parameter or specific solver.

In order to solve problems defined on embedded surfaces with boundary, we only considered cases of boundaries belonging to mesh surfaces. In this way, the algorithm proposed by [53] has been directly applied to enforce Dirichlet boundary conditions. The compatibility between the new P1-equivalent space built on the embedded domain and the stable Lagrange multiplier space defined along the embedded boundary has been numerically verified by using inf-sup test. To our knowledge the issue of boundary conditions applied on a boundary of codimension two in 3D has not been investigated up to now without a body-fitted mesh. To overcome this limitation, we explore — in a Part II [6] following this paper — a new way of building stable Lagrange multipliers with the objective of addressing all the range of

problems with embedded solids.

Future investigations may include vector-valued problems, like embedded membranes, non-linear and hyperbolic PDEs.

## Acknowledgements

The authors thank the reviewers for providing particularly detailed reports, which contributed to improve this article. The open source meshing software Gmsh [61] has been used to generate the background meshes involved in the numerical validations.

## References

- [1] T. J. R. Hughes, J. A. Cottrell, Y. Bazilevs, Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement, *Computer Methods in Applied Mechanics and Engineering* 194 (39–41) (2005) 4135 – 4195.
- [2] T. Belytschko, T. Black, Elastic crack growth in finite elements with minimal remeshing, *International Journal for Numerical Methods in Engineering* 45 (5) (1999) 601–620.
- [3] N. Moës, J. E. Dolbow, T. Belytschko, A finite element method for crack growth without remeshing, *International Journal for Numerical Methods in Engineering* 46 (1) (1999) 131–150.
- [4] G. Xu, E. Atroshchenko, W. Ma, S. Bordas, Geometry-Independent Field approximation: CAD-Analysis Integration, geometrical exactness and adaptivity *Preprint*.
- [5] T. Belytschko, C. Parimi, N. Moës, N. Sukumar, S. Usui, Structured extended finite element methods for solids defined by implicit surfaces, *International Journal for Numerical Methods in Engineering* 56 (4) (2003) 609–635.
- [6] F. Duboeuf, E. Béchet, Embedded solids of any dimension in the X-FEM. Part II - Imposing Dirichlet boundary conditions, *Finite Elements in Analysis and Design* *Submitted*.
- [7] V. K. Saul’ev, On the solution of some boundary value problems on high performance computers by fictitious domain method, *Siberian Mathematical Journal* 4 (4) (1963) 912–925.
- [8] R. Rangarajan, A. J. Lew, Universal meshes: A method for triangulating planar curved domains immersed in nonconforming meshes, *International Journal for Numerical Methods in Engineering* 98 (4) (2014) 236–264.
- [9] D. W. Zaide, C. F. Ollivier-Gooch, Inserting a surface into an existing unstructured mesh: Surface insertion into unstructured meshes, *International Journal for Numerical Methods in Engineering* 106 (6) (2016) 484–500.



- 
- [10] I. Ramière, P. Angot, M. Belliard, A general fictitious domain method with immersed jumps and multilevel nested structured meshes, *Journal of Computational Physics* 225 (2) (2007) 1347–1387.
- [11] D.-L. Quan, T. Toulorge, E. Marchandise, J.-F. Remacle, G. Bricteux, Anisotropic mesh adaptation with optimal convergence for finite elements using embedded geometries, *Computer Methods in Applied Mechanics and Engineering* 268 (2014) 65–81.
- [12] J. Sanders, M. A. Puso, An embedded mesh method for treating overlapping finite element meshes, *International Journal for Numerical Methods in Engineering* 91 (3) (2012) 289–305.
- [13] J. A. Benek, P. G. Buning, J. L. Steger, A 3-D chimera grid embedding technique, in: 7th Computational Fluid Dynamics Conference, New York, AIAA, Cincinnati, OH, 1985, pp. 322–331.
- [14] S. Keeling, R. Tramel, J. Benek, A theoretical framework for Chimera Domain Decomposition, in: *Advances in Flow Simulation Techniques - A conference dedicated to the memory of Joseph L. Steger*, Davis, California, 1997.
- [15] J. Fish, The s-version of the finite element method, *Computers & Structures* 43 (3) (1992) 539–547.
- [16] H. B. Dhia, G. Rateau, The Arlequin method as a flexible engineering design tool, *International Journal for Numerical Methods in Engineering* 62 (11) (2005) 1442–1462.
- [17] C. Bernardi, Y. Maday, A. T. Patera, A new nonconforming approach to domain decomposition: the mortar element method, in: *Nonlinear partial differential equations and their applications: Collège de France Seminar, Vol. XI of Pitman Research Notes in Mathematics Series 299*, H. Brezis and J.-L. Lions, Longman Scientific & Technical, Harlow, 1994, pp. 13–51.
- [18] I. Babuška, The finite element method with Lagrangian multipliers, *Numerische Mathematik* 20 (3) (1973) 179–192.
- [19] A. Hansbo, P. Hansbo, M. G. Larson, A finite element method on composite grids based on Nitsche’s method, *ESAIM: Mathematical Modelling and Numerical Analysis* 37 (03) (2003) 495–514.
- [20] L. Rukhovets, A remark on the method of fictive domains, *Differential Equations* 3 (4) (1967) 114–121.
- [21] C. S. Peskin, Flow patterns around heart valves: A numerical method, *Journal of Computational Physics* 10 (2) (1972) 252–271.
- [22] B. Maury, A Fat Boundary Method for the Poisson Problem in a Domain with Holes, *Journal of Scientific Computing* 16 (3) (2001) 319–339.
- [23] R. Glowinski, T.-W. Pan, J. Periaux, A fictitious domain method for Dirichlet problem and applications, *Computer Methods in Applied Mechanics and Engineering* 111 (3–4) (1994) 283–303.
- [24] R. LeVeque, Z. Li, The immersed interface method for elliptic equations with discontinuous coefficients and singular sources, *SIAM Journal on Numerical Analysis* 31 (4) (1994) 1019–1044.
- [25] Z. Li, The immersed interface method using a finite element formulation, *Applied Numerical Mathematics* 27 (3) (1998) 253–267.

- 
- [26] N. Moës, M. Cloirec, P. Cartraud, J.-F. Remacle, A computational approach to handle complex microstructure geometries, *Multiscale Computational Mechanics for Materials and Structures* 192 (28–30) (2003) 3163–3177.
- [27] M. Mournnassi, S. Belouettar, E. Béchet, S. P.-A. Bordas, D. Quoirin, M. Potier-Ferry, Finite element analysis on implicitly defined domains: An accurate representation based on arbitrary parametric surfaces, *Computer Methods in Applied Mechanics and Engineering* 200 (5-8) (2011) 774–796.
- [28] I. Babuška, J. Melenk, The partition of unity method, *International Journal for Numerical Methods in Engineering* 40 (4) (1997) 727–758.
- [29] J. Melenk, I. Babuška, The partition of unity finite element method: Basic theory and applications, *Computer Methods in Applied Mechanics and Engineering* 139 (1-4) (1996) 289–314.
- [30] J. Dolbow, N. Moës, T. Belytschko, Discontinuous enrichment in finite elements with a partition of unity method, *Finite Elements in Analysis and Design* 36 (3-4) (2000) 235–260.
- [31] M. Bertalmío, L.-T. Cheng, S. Osher, G. Sapiro, Variational Problems and Partial Differential Equations on Implicit Surfaces, *Journal of Computational Physics* 174 (2) (2001) 759–780.
- [32] K. Deckelnick, G. Dziuk, C. M. Elliott, C.-J. Heine, An h-narrow band finite-element method for elliptic equations on implicit surfaces, *IMA Journal of Numerical Analysis* 30 (2) (2010) 351–376.
- [33] I. Babuška, A. K. Aziz, On the Angle Condition in the Finite Element Method, *SIAM Journal on Numerical Analysis* 13 (2) (1976) pp. 214–226.
- [34] M. A. Olshanskii, A. Reusken, X. Xu, On surface meshes induced by level set functions, *Computing and Visualization in Science* 15 (2) (2013) 53–60.
- [35] M. Olshanskii, A. Reusken, J. Grande, A finite element method for elliptic equations on surfaces, *SIAM Journal on Numerical Analysis* 47 (5) (2009) 3339–3358.
- [36] S. Amdouni, P. Hild, V. Lleras, M. Moakher, Y. Renard, A stabilized Lagrange multiplier method for the enriched finite-element approximation of contact problems of cracked elastic bodies, *ESAIM: Mathematical Modelling and Numerical Analysis* 46 (04) (2012) 813–839.
- [37] G. Dziuk, C. M. Elliott, Finite element methods for surface PDEs, *Acta Numerica* 22 (2013) 289–396.
- [38] A. Demlow, M. A. Olshanskii, An Adaptive Surface Finite Element Method Based on Volume Meshes, *SIAM Journal on Numerical Analysis* 50 (3) (2012) 1624–1647.
- [39] A. Y. Chernyshenko, M. A. Olshanskii, An adaptive octree finite element method for PDEs posed on surfaces, *Computer Methods in Applied Mechanics and Engineering* 291 (2015) 146–172.
- [40] A. Reusken, Analysis of trace finite element methods for surface partial differential equations, *IMA Journal of Numerical Analysis* 35 (4) (2015) 1568–1590.
- [41] E. Burman, P. Hansbo, M. G. Larson, A stabilized cut finite element method for partial differential equations on surfaces: The Laplace–Beltrami operator, *Computer Methods in Applied Mechanics and*

- 
- Engineering 285 (2015) 188–207.
- [42] E. Burman, P. Hansbo, M. G. Larson, A. Massing, S. Zahedi, Full gradient stabilized cut finite element methods for surface partial differential equations, *Computer Methods in Applied Mechanics and Engineering* 310 (2016) 278–296.
- [43] C. D’Angelo, A. Quarteroni, On the coupling of 1d and 3d diffusion-reaction equations: application to tissue perfusion problems, *Mathematical Models and Methods in Applied Sciences* 18 (08) (2008) 1481–1504.
- [44] C. D’Angelo, Finite Element Approximation of Elliptic Problems with Dirac Measure Terms in Weighted Spaces: Applications to One- and Three-dimensional Coupled Problems, *SIAM Journal on Numerical Analysis* 50 (1) (2012) 194–215.
- [45] F. Auricchio, D. Boffi, L. Gastaldi, A. Lefieux, A. Reali, A study on unfitted 1d finite element methods, *Computers & Mathematics with Applications* 68 (12) (2014) 2080–2102.
- [46] S. Osher, J. A. Sethian, Fronts propagating with curvature dependent speed: Algorithms based on Hamilton-Jacobi formulations, *Journal of Computational Physics* 79 (1) (1988) 12–49.
- [47] J. A. Sethian, Level set methods and fast marching methods: Evolving interfaces in computational geometry, fluid mechanics, computer vision, and materials science, Vol. 3 of Cambridge monographs on applied and computational mathematics, Cambridge University Press, Cambridge, UK, 1999.
- [48] H. Sauerland, T.-P. Fries, The extended finite element method for two-phase and free-surface flows: A systematic study, *Journal of Computational Physics* 230 (9) (2011) 3369–3390.
- [49] P. Burchard, L.-T. Cheng, B. Merriman, S. Osher, Motion of Curves in Three Spatial Dimensions Using a Level Set Approach, *Journal of Computational Physics* 170 (2) (2001) 720–741.
- [50] R. Durand, M. M. Farias, D. M. Pedroso, Computing intersections between non-compatible curves and finite elements, *Computational Mechanics* 56 (3) (2015) 463–475.
- [51] N. Moës, E. Béchet, M. Tourbier, Imposing Dirichlet boundary conditions in the extended finite element method, *International Journal for Numerical Methods in Engineering* 67 (12) (2006) 1641–1669.
- [52] S. Geniaut, P. Massin, N. Moës, A stable 3D contact formulation using X-FEM, *European Journal of Computational Mechanics* 16 (2) (2007) 259–275.
- [53] E. Béchet, N. Moës, B. Wohlmuth, A stable Lagrange multiplier space for stiff interface conditions within the extended finite element method, *International Journal for Numerical Methods in Engineering* 78 (8) (2009) 931–954.
- [54] G. Ferté, P. Massin, N. Moës, Interface problems with quadratic X-FEM: design of a stable multiplier space and error analysis, *International Journal for Numerical Methods in Engineering* 100 (11) (2014) 834–870.
- [55] S. Fernández-Méndez, A. Huerta, Imposing essential boundary conditions in mesh-free methods, *Com-*

- puter Methods in Applied Mechanics and Engineering 193 (12-14) (2004) 1257–1275.
- [56] D. Chapelle, K. Bathe, The inf-sup test, *Computers & Structures* 47 (4-5) (1993) 537–545.
- [57] M. A. Olshanskii, A. Reusken, A finite element method for surface PDEs: matrix properties, *Numerische Mathematik* 114 (3) (2010) 491–520.
- [58] N. Moës, A. Gravouil, T. Belytschko, Non-planar 3D crack growth by the extended finite element and level sets. Part I: Mechanical model, *International Journal for Numerical Methods in Engineering* 53 (11) (2002) 2549–2568.
- [59] M. Siavelis, M. L. E. Guiton, P. Massin, N. Moës, Large sliding contact along branched discontinuities with X-FEM, *Computational Mechanics* 52 (1) (2013) 201–219.
- [60] S. Groß, V. Reichelt, A. Reusken, A finite element based level set method for two-phase incompressible flows, *Computing and Visualization in Science* 9 (4) (2006) 239–257.
- [61] C. Geuzaine, J.-F. Remacle, Gmsh: A 3-D finite element mesh generator with built-in pre- and post-processing facilities, *International Journal for Numerical Methods in Engineering* 79 (11) (2009) 1309–1331.