

Parameter-free, weak imposition of Dirichlet boundary conditions and coupling of trimmed and non-conforming patches

S. Kollmannsberger^{1*}, A. Özcan¹, J. Baiges², M. Ruess³, E. Rank¹, A. Reali⁴

¹*Computation in Engineering, Technische Universität München; Munich, Germany*

²*International Center for Numerical Methods in Engineering, Universitat Politècnica de Catalunya, Barcelona, Spain*

³*Aerospace Structures and Computational Mechanics, Delft University of Technology, Delft, Netherlands*

⁴*Department of Civil Engineering and Architecture, University of Pavia, Italia*

SUMMARY

We present a parameter-free domain sewing approach for low- as well as high-order finite elements. Its final form contains only primal unknowns, i.e., the approach does not introduce additional unknowns at the interface. Additionally, it does not involve problem dependent parameters which require an estimation. The presented approach is symmetry-preserving, i.e. the resulting discrete form of an elliptic equation will remain symmetric and positive definite. It preserves the order of the underlying discretization and we demonstrate high order accuracy for problems of non-matching discretizations concerning the mesh size h as well as the polynomial degree of the order of discretization p . We also demonstrate how the method may be used to model material interfaces which may be curved and for which the interface does not coincide with the underlying mesh. This novel approach is presented in the context of the p- and B-spline versions of the finite cell method, an embedded domain method of high order, and compared to more classical methods such as the penalty method or Nitsche's method. Copyright © 0000 John Wiley & Sons, Ltd.

Received . . .

KEY WORDS: Dirichlet boundary conditions, embedded domain, trimmed patches, Isogeometric Analysis, p -FEM, finite cell method

1. INTRODUCTION

We reconsider the imposition of Dirichlet boundary conditions in the context of the finite element method. We focus on the case, where the Dirichlet boundary conditions are imposed independent of the underlying discretization, i.e., they do not coincide with nodes, edges or faces of the mesh but may cut elements in an arbitrary fashion.

Whereas this problem has been investigated for decades, it has gained increased attention recently, for example in the context of Isogeometric Analysis (IGA) [1, 2]. The main motivation in IGA is to facilitate the transition from geometric models to computational analysis and thereby render mesh-generation unnecessary. This is achieved by an isoparametric approach, based on applying typical functions (e.g., B-Splines and NURBS) from Computer Aided Design (CAD) systems directly in order to discretize the differential equations that describe the underlying physics. The Dirichlet boundary conditions, however, are not always associated with the boundary of the geometric model.

*Correspondence to: Computation in Engineering, Technische Universität München; Munich, Germany. E-mail: kollmannsberger@bv.tu-muenchen.de

Contract/grant sponsor: S. Kollmannsberger and A. Reali greatly acknowledge the funds granted under DAAD grant No. 54577652 'Patching trimmed surfaces in isogeometric analysis'. The last author also acknowledges the funds granted under the ERC grant 'ISOBIO', No. '259229'. These grants made this contribution possible.

Another problem class in which the need for a mesh independent imposition of Dirichlet boundary conditions arises are embedded domain methods. By definition, they do not represent the underlying physical domain with a boundary conforming mesh. In this contribution, we consider the finite cell method [3, 4], an embedded domain method of high-order which is also applicable to IGA [5].

Numerous approaches and a vast body of literature exist for the imposition of Dirichlet boundary conditions, each of them with different properties, advantages and disadvantages. None of them is perfect for all applications. Our wish-list for an ideal imposition of Dirichlet boundary conditions would require the method

- to be independent of the underlying discretization not only geometrically but such that, for example, a change of the approximation basis does not require a modification of the method;
- to be numerically stable and variationally consistent;
- to preserve a symmetric, positive definite system matrices for elliptic problems;
- to be applicable to hyperbolic and parabolic problems;
- not to introduce additional degrees of freedom;
- not to deteriorate the order of approximation of the underlying discretization;
- to be suitable for high-order accuracy;
- to be free from problem-dependent parameters.

We will first provide a quick and non-exhaustive review of some methods which comply with at least some of the properties mentioned above. For a more general review see, for example, [6, 7]. In the contribution at hand we are only interested in methods which were shown to be suitable for high-order discretizations and/or set the stage for the approach to be presented.

In principle, there are two choices for the imposition of Dirichlet boundary conditions. First, the Dirichlet boundary conditions may be directly built into the basis functions by modification, extrapolation or augmentation of the basis. Methods belonging to the first group include web-splines [8], i-splines [9, 10], max-ent interpolation [11], the extended finite element method (X-FEM), the generalized finite element method (GFEM) and level-set like approaches [12, 13, 14]. Second, one may enforce Dirichlet boundary conditions by augmenting the governing equations with suitable constraint conditions. This second group emanates from enforcing constraints in variational problems and is, thus, very general. The most commonly used is the penalty method [15, 16], which is easy to implement and which produces a positive definite system matrix for elliptic problems. However, it is not a variationally consistent method which renders the results to be strongly dependent on the choice of the penalty parameter. Additionally, the need to select a high penalty value results in an ill-conditioned system of equations.

A classic alternative to the penalty method is to enforce the constraints through Lagrange multipliers [17]. This produces a variationally consistent method with additional degrees of freedom. However, the Lagrange multiplier space must satisfy the inf-sup condition [17, 18]. Thus, the choice of the discretization of the Lagrange multiplier is not always obvious. It depends on the discretization including the mesh, which turns out to be quite restrictive. The restrictions can be circumvented by means of especially developed stabilized Lagrange multiplier methods. They can be interpreted as an attempt to combine penalty and Lagrange multiplier methods and have become a popular choice in the context of embedded domain modeling [19, 20, 21, 22, 23].

Another attractive and very general alternative is Nitsche's method [24]. It can be interpreted as a stabilized Lagrange multiplier technique in which the Lagrange multiplier is directly identified to be the flux normal to the Dirichlet boundary. The close relation of Nitsche's method to the Lagrange multipliers method was first analyzed by Stenberg [25]. Even though Nitsche's method eliminates the need to discretize the Lagrange multipliers with additional variables, the stabilization term remains in the formulation and continues to contain a problem-dependent parameter. Nitsche's method is much less sensitive to the choice of a stabilization parameter than the simple penalty method. However, for optimal performance, such a parameter still needs to be estimated. A suitable estimation for a lower bound is, for example, given by Griebel and Schweitzer [26] who propose to solve an auxiliary generalized eigenvalue problem. Beyond the Poisson's equation originally investigated by Nitsche, his idea has been extended to other problem classes. The potential of Nitsche's method is nicely laid out in a review article by Hansbo [6]. It has been further extended,

for example, to thin plate problems [27], three-dimensional elasticity [28], thermo-elasticity [29], and fluid dynamics [30].

Recently, different schemes have been proposed that are neither dependent on additional degrees of freedom nor on the solution of auxiliary problems to estimate the stabilization terms. Among these are schemes using degrees of freedom of certain nodes of the finite element mesh to minimize the difference between the exact and the approximated boundary condition [31]. They are robust but yield a non-symmetric equation, also for symmetric problems. In the contribution at hand, we follow a different approach, namely to use Discontinuous Galerkin techniques. Herein, the material law is discretized with functions being discontinuous across element boundaries. The additional degrees of freedom can then be condensed out at the discrete level prior to the solution of the system matrix, so they will not appear in the final formulation. These approaches can yield automatic stabilization terms, i.e., terms that add stability without the need to compute auxiliary eigenvalue problems, see e.g. [7]. The method proposed in [32] is a recently proposed variant but again leads to unsymmetric systems for symmetric problems. However, the formulation was symmetrized in [33]. This formulation will mark the point of departure in the paper at hand in which we will start by investigating its suitability for high-order embedded domain discretizations.

Imposing Dirichlet boundary conditions is also of importance for weak coupling of different discretizations. Geometric models in CAD systems, for example, are usually comprised of many trimmed patches of splines of all types, such as B-Splines, NURBS, T-Splines, etc. These trimmed patches should serve as the geometric model in IGA. If this model is to be directly transferred to computational analysis without mesh generation, the trimmed patches have to be glued at their common and usually non-matching interface by means of enforcing continuity of the unknowns and/or their derivatives to yield a consistent computational model.

The same task needs to be addressed in embedded domain analysis. They are popular for geometries which are difficult to mesh or for cases in case a geometry and/or topology changes over time. Chimera approaches, for example, resolve the domain with one or multiple overlapping meshes each of which represents a different physical domain. The continuity at their common boundary is then enforced either iteratively or monolithically. Throughout this treatise, we only consider monolithic coupling, i.e., the coupling terms will directly appear in the resulting system matrix and the solution of the coupled system is obtained in one step only.

Monolithic approaches using Lagrange multipliers to satisfy continuity constraints along the domain interfaces are mostly named Mortar methods. Many variants of this technique have been proposed [34, 35, 36, 37, 38], to list just a few. As in the case of the weak imposition of Dirichlet boundary conditions with Lagrange multipliers, its straightforward application results in rather restrictive stability conditions, which is why stabilized versions are advocated. Again, an alternative for monolithic coupling is to use Nitsche's method to impose the constraints. The use of Nitsche's method to couple subdomains was extensively analysed in [39, 40]. Hansbo et al. [41] extended this concept to overlapping domains. It was also used to represent strong and weak singularities [42]. The concept of using Nitsche's method for coupling domains has been applied to a variety of fields such as linear elasticity [43, 44], contact [45] and fluid-structure interaction [46]. As in the case of imposing Dirichlet boundary conditions, using Nitsche's idea for coupling requires the solution of an auxiliary generalized eigenvalue problem for an estimation of the optimal stabilization parameter. In this contribution, we derive a method which sets out to eliminate the need for solving auxiliary equations to estimate stabilization parameters and, at the same time, leads to symmetric system matrices for elliptic problems.

In order to set up our main field of application of our proposed technique we begin with the introduction of the finite cell method in section 2 before continuing with the weak imposition of Dirichlet boundary conditions in section 3, embarking from classical Nitsche's formulation. Next, we recall the recently introduced boundary conditions for Poisson's problem as presented in [33] but put them in the context of the finite cell method. We show that the concept in [33] can be straightforwardly utilized in the framework of high-order finite elements in general. We investigate the performance of the boundary conditions and compare their results to the penalty method as well as to Nitsche's method for Poisson's equation in one and two dimensions. The two-dimensional

examples are computed by discretizing the continuous problems with the classical p -version of the finite element method utilizing the hierarchical integrated Legendre polynomials as introduced in [47]. Additionally, we evaluate the performance of this new type of boundary conditions in the context of patches of B-Splines which form the basis for Isogeometric Analysis.

Section 4 is devoted to the new, parameter-free and symmetry-preserving weak coupling of domains. Again, we start by recalling the more classical Nitsche's method for the problem under consideration. The derivation of the new scheme is carried out in sections 4.3 and 4.4. We prove the scheme to be stable in section 4.5 and show that in all tested cases the resulting discrete system is positive definite. Utilizing the presented methodology, we demonstrate high convergence rates in all cases. Specifically, we first evaluate the new method's capabilities for Poisson's equation with non-conform discretizations in h and p and constant coefficients in a boundary-conform setting (section 4.6.1) before we move to an overlapping discretization with an inclined coupling boundary (section 4.6.2). We then demonstrate the method for a bimetal strip. Naturally, this example possesses discontinuous coefficients and we apply non-conforming discretizations in h and p (section 4.6.3). The last example is an inclusion problem. It has discontinuous coefficients and we discretize it in a Chimera-like fashion by using an overlapping discretizations (section 4.6.4). All examples are computed by using B-Spline patches and classical p -FEM and are compared to Nitsche's method. Finally, we draw conclusions in section 5.

2. FINITE CELL METHOD

2.1. Applications of the finite cell method

The finite cell method was first introduced in [3, 4], where its potential was demonstrated for linear-elastic examples in two and three dimensions. Various extensions of the FCM confirm its versatility in the context of topology optimization [48], geometrically nonlinear continuum mechanics [49], adaptive mesh-refinement [50, 51, 52], computational steering [53, 54], biomedical engineering [55], numerical homogenization [56], elastoplasticity [57], wave propagation in heterogeneous materials [58], local enrichment for material interfaces [14], convection-diffusion problems [59, 60], thin-walled structures [61], design-through-analysis and isogeometric analysis [51, 52, 5], and multi-physical applications [29].

A free FCM-MATLAB-toolbox is provided at <http://fcmlab.cie.bgu.tum.de/> under the GNU Licence along with extensive online documentation [62] as well as numerous examples. It serves as a library to solve one-, two-, and three-dimensional elasto-static and elasto-dynamic problems. It utilizes advanced software development techniques such as Design Patterns [63] to provide maximum flexibility to users as well as Unit Testing to ensure code consistency throughout the development process.

2.2. Formulation

The finite cell method is an embedded domain method based on higher order shape functions. The primary goal of the method is to avoid the generation of boundary-conforming meshes. Nevertheless, it provides high-order convergence rates. For this purpose, it assumes a three-dimensional physical domain Ω and an embedding domain Ω_e such that $\Omega_e \supset \Omega$ whereby Ω_e is of a simpler, typically rectangular shape. Their boundaries are defined as $\partial\Omega = \Gamma$ and $\partial\Omega_e = \Gamma_e$.

Let us consider the following bilinear form $\mathcal{B}(\cdot, \cdot)_{\Omega_e}$ defined on Ω_e :

$$\mathcal{B}(\mathbf{u}, \mathbf{v})_{\Omega_e}^\alpha = \int_{\Omega_e} [\mathbf{L}\mathbf{v}]^T \mathbf{C}_e^\alpha [\mathbf{L}\mathbf{u}] d\Omega_e \quad (1)$$

in which \mathbf{u} is a scalar- or vector-valued function, \mathbf{v} is a test function, and \mathbf{L} is a differential operator. The material matrix of the embedding domain is defined as $\mathbf{C}_e^\alpha = \alpha \mathbf{C}$ where \mathbf{C} is the material

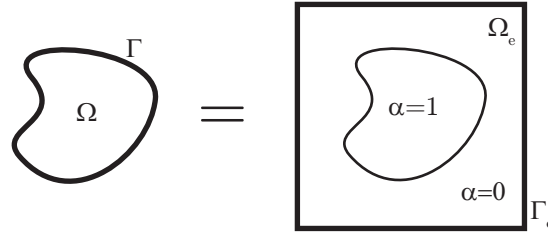


Figure 1. The finite cell method: Physical domain Ω and embedding domain Ω_e .

matrix of the physical domain Ω and α is an indicator function defined as:

$$\alpha(\mathbf{x}) = \begin{cases} 1 & \forall \mathbf{x} \in \bar{\Omega} \\ 0 & \forall \mathbf{x} \in \Omega_e \setminus \Omega \end{cases} \quad (2)$$

where $\bar{\Omega}$ denotes the closure of Ω . The finite cell method is depicted in fig. 1. It is important to note that eq. (2) implies $\alpha(\mathbf{x}) = 1.0 \quad \forall \mathbf{x} \in \Gamma$ as, otherwise, it would not be possible to apply non homogeneous boundary conditions on Γ . Furthermore, the definition of $\alpha(\mathbf{x})$ in eq. (2) ensures that the bilinear forms defined on the embedding domain Ω_e and on the physical domain Ω are equivalent:

$$\mathcal{B}(\mathbf{u}, \mathbf{v})_{\Omega_e}^\alpha = \int_{\Omega_e} [\mathbf{L}\mathbf{v}]^T \alpha \mathbf{C} [\mathbf{L}\mathbf{u}] d\Omega \quad (3)$$

$$= \int_{\Omega} [\mathbf{L}\mathbf{v}]^T \mathbf{C} [\mathbf{L}\mathbf{u}] d\Omega + \int_{\Omega_e \setminus \Omega} [\mathbf{L}\mathbf{v}]^T \mathbf{0} [\mathbf{L}\mathbf{u}] d\Omega \quad (4)$$

$$= \int_{\Omega} [\mathbf{L}\mathbf{v}]^T \mathbf{C} [\mathbf{L}\mathbf{u}] d\Omega \quad (5)$$

$$= \mathcal{B}(\mathbf{u}, \mathbf{v})_{\Omega} \quad (6)$$

The linear functional $\mathcal{F}(\mathbf{v})$

$$\mathcal{F}(\mathbf{v}) = \int_{\Omega_e} \mathbf{v}^T (\alpha \mathbf{f}) d\Omega + \int_{\Gamma_n} \mathbf{v}^T \mathbf{t} d\Gamma \equiv \mathcal{F}(\mathbf{v})_e^\alpha \quad (7)$$

is identical for the extended and the physical domain and also takes the volume sources \mathbf{f} inside Ω and prescribed fluxes on the Neumann part of the boundary of the physical domain Γ_n into account. The solution of

$$\mathcal{B}(\mathbf{u}, \mathbf{v})_{\Omega_e}^\alpha = \mathcal{F}(\mathbf{v})_e^\alpha \quad (8)$$

is equivalent to solving $\mathcal{B}(\mathbf{u}, \mathbf{v})_{\Omega} = \mathcal{F}(\mathbf{v})$. In the sequel we will, therefore, drop the subindex e unless we want to specifically stress one viewpoint or the other.

A discretized view of the finite cell method is depicted in fig. 2. For the numerical approximation of eq. (8) it is convenient to define a computational grid on the extended domain Ω_e . This grid forms the support of the basis functions. In principle, we can use a uniform Cartesian grid, as was applied in the first implementations of the FCM [3, 4]. Hierarchically refined grids can be defined alike [64, 65, 66], if local features of the solution are to be resolved in more detail. Note that the boundary of the physical domain is not resolved or even approximated by the grid. In order not to change the problem, the geometry needs to be recovered at the integration level. A natural strategy is the following:

Let us denote elements of the computational grid such as cells Ω_C to distinguish them from classical finite elements. It is convenient to categorize cells into three different types which are depicted in fig. 2:

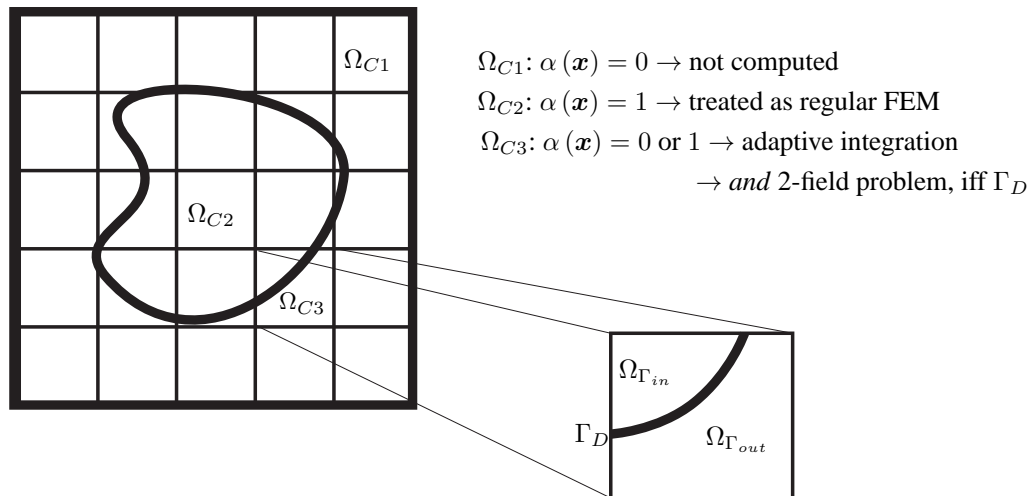


Figure 2. The discrete point of view of the finite cell method. The cells Ω_C provide the support of the shape functions much like finite elements do. The important case is Ω_{C3} . In addition to an adaptive integration we advocate solving the 2-field problem on Ω_{C3} if and only if a Dirichlet boundary condition is imposed on Γ parts of it (see section 3.2). The part of Γ on which Dirichlet boundary conditions are imposed is labeled Γ_D .

1. Ω_{C1} : Cells where $\alpha(\mathbf{x}) = 0 \quad \forall \mathbf{x} \in \Omega_c$, which are not computed.
2. Ω_{C2} : Cells where $\alpha(\mathbf{x}) = 1 \quad \forall \mathbf{x} \in \Omega_c$, which are computed as if they were regular finite elements.
3. Ω_{C3} : Cells cut by the boundary Γ , which receive special treatment as described in the following.

Cells of type Ω_{C3} are decomposed into two parts $\Omega_{\Gamma_{in}}$ in which $\alpha(\mathbf{x}) = 1$, and $\Omega_{\Gamma_{out}}$ in which $\alpha(\mathbf{x}) = 0$. In practical computations, we assume the parameter $\alpha(\mathbf{x})$ not to be exactly zero. Instead, we chose $\epsilon \ll 1$ to ease numerical round-off issues in the solution process. Thus, we explicitly accept a modeling error of the order ϵ which is controllable as it tends to zero if ϵ approaches zero. We usually choose ϵ to have very small magnitude, typically about 10^{-8} smaller than the material constant.

The numerical integration of the bilinear form must be carried out accurately enough on the cut cells as to recover the bilinear form of eq. (6) at the discrete level. For this purpose, the integration of the bilinear form is ideally only carried out on $\Omega_{\Gamma_{in}}$. One way to perform the domain-integration numerically is to approximate $\Omega_{\Gamma_{in}}$ adaptively by recursive bisection, allowing for a simple and efficient refinement towards the boundary [64, 5, 67]. Geometrically more involved, but also more efficient schemes utilize straight sided triangles, —see, e.g., [28]. Precise numerical integration schemes for complex geometries based on the blending function method [68] are also possible.

Homogeneous Neumann boundary conditions are easy to deal with. They are equivalent to assuming material with zero stiffness in the domain $\Omega_e \setminus \Omega$ and, therefore, need no special treatment in the framework of the finite cell method. Inhomogeneous Neumann boundary conditions are realized by including the second term in eq. (7), i.e., by directly integrating over Γ . The integrand is simply a product of a function living on the surface (the load) and the shape functions \mathbf{u} defined over the cells, yet evaluated at the surface Γ .

The imposition of Dirichlet type boundary conditions on Γ or on parts of it, i.e., Γ_D cuts Ω_{C3} , is a central topic of this paper and is discussed in section 3.

3. WEAK IMPOSITION OF DIRICHLET BOUNDARY CONDITIONS

3.1. Nitsche's method

We consider the following scalar-valued single field problem:

$$-k\Delta u = f \quad \text{in } \Omega \quad (9)$$

$$u = \bar{u} \quad \text{on } \Gamma = \partial\Omega \quad (10)$$

For simplicity, we assume a pure Dirichlet-problem, i.e. $\Gamma_D = \Gamma$. If Dirichlet conditions are imposed only on parts of the boundary, corresponding integrals in the following formulations have to be taken only over these parts. If the boundary conditions eq. (10) are imposed by Nitsche's method [24], the potential corresponding to eqs. (9) and (10) reads:

$$\Pi(u, \boldsymbol{\sigma}) = \int_{\Omega} \left(\frac{1}{2}k(\nabla u)^2 - fu \right) d\Omega - \int_{\Gamma} \mathbf{n} \cdot \boldsymbol{\sigma} (u - \bar{u}) d\Gamma + \frac{1}{2}\beta \int_{\Gamma} (u - \bar{u})^2 d\Gamma \quad (11)$$

The first term in eq. (11) is clearly the potential associated to eq. (9). The second term enforces the boundary conditions of eq. (10) by means of a Lagrange multiplier λ which has already been identified to be equal to the fluxes at the boundary $\lambda = -\mathbf{n} \cdot \boldsymbol{\sigma}$. This allows a formulation in primal variables, i.e., no additional degrees of freedom are needed to approximate λ . However, the negative sign of the Lagrange multiplier may lead to a loss of coercivity which is restored by the third term in the potential of eq. (11) which again enforces u to be equal to \bar{u} at Γ by means of a penalization. Coercivity is only restored for a sufficiently high choice of β . Unfortunately, a closer look reveals that the penalty parameter β is problem-dependent. Fortunately, it has turned out that the precise choice of β does not have a large influence on the solution above a certain threshold. However, an optimal value for β is not easily obtained and the threshold above which its influence on the result is of only minor importance is unknown a priori.

The weak form of the imposition of the boundary conditions by Nitsche's method may either be obtained by multiplication of eqs. (9) and (10) by a suitable test function and integration by parts, or by variation of the potential given in eq. (11). In any case, both approaches lead to the following weak form [24]:

$$k(\nabla u, \nabla v)_{\Omega} - k(\mathbf{n} \cdot \nabla u, v)_{\Gamma} - k(u, \mathbf{n} \cdot \nabla v)_{\Gamma} + k\beta(u, v)_{\Gamma} = \langle f, v \rangle_{\Omega} - k\langle \bar{u}, \mathbf{n} \cdot \nabla v \rangle_{\Gamma} + k\beta\langle \bar{u}, v \rangle_{\Gamma} \quad (12)$$

where we used the common short hand notation of the L^2 product (\cdot, \cdot) and $\langle \cdot, \cdot \rangle$ denotes the integral of two functions over a domain.

A lower bound to estimate the parameter β in eq. (12) to ensure coercivity may be obtained by solving the following auxiliary eigenvalue problem as, for example, suggested in [69, 28]:

$$\mathbf{A}\mathbf{x} = \Lambda\mathbf{B}\mathbf{x} \quad (13)$$

$$\mathbf{A} = \int_{\Gamma} (\nabla u \cdot \mathbf{n})(\nabla v \cdot \mathbf{n})d\Gamma$$

$$\mathbf{B} = \int_{\Omega} (\nabla u \cdot \nabla v)d\Omega$$

3.2. Alternative to Nitsche's method

The symmetric form of the imposition of boundary conditions presented in this section closely follows the work of Baiges et al. [33] but puts the methodology into the context of the finite cell method. Even though the presentation starts from a mixed-field formulation, its final form contains only the primal unknowns.

3.2.1. *Weak form* Equations (9) and (10) can be recast into the following two-field problem:

$$-k\Delta u = f \quad \text{in } \Omega \quad (14)$$

$$\frac{1}{k}\boldsymbol{\sigma} = \nabla u \quad \text{in } \Omega_{\Gamma_{in}} \quad (15)$$

$$u = \bar{u} \quad \text{on } \Gamma \quad (16)$$

As usual for mixed-field formulations, u and $\boldsymbol{\sigma}$ in eqs. (14) to (16) are discretized independently. The associated potential to eqs. (14) to (16) is

$$\Pi(u, \boldsymbol{\sigma}) = \int_{\Omega} \left(\frac{1}{2}k(\nabla u)^2 - fu \right) d\Omega - \int_{\Gamma} \mathbf{n} \cdot \boldsymbol{\sigma} (u - \bar{u}) d\Gamma - \frac{1}{2nk} \int_{\Omega_{\Gamma_{in}}} (\boldsymbol{\sigma} - k\nabla u)^2 d\Omega_{\Gamma_{in}} \quad (17)$$

where n is any real number greater than two. The only difference between eq. (11) and eq. (17) is that the latter enforces the boundary condition with the help of the material law of eq. (15) which is imposed weakly on $\Omega_{\Gamma_{in}}$. In essence, the potential eq. (17) enforces the Lagrange multiplier to be the trace of the unknown in a least squares sense.

The variation of the potential given in eq. (17) leads to the following weak form of eqs. (14) to (16):

$$k(\nabla u, \nabla v)_{\Omega} - \langle \mathbf{n} \cdot \boldsymbol{\sigma}, v \rangle_{\Gamma} + \frac{1}{n}(\nabla v, \boldsymbol{\sigma})_{\Omega_{\Gamma_{in}}} - \frac{1}{n}k(\nabla v, \nabla u)_{\Omega_{\Gamma_{in}}} = \langle f, v \rangle_{\Omega} \quad (18)$$

$$-\frac{1}{nk}(\boldsymbol{\tau}, \boldsymbol{\sigma})_{\Omega_{\Gamma_{in}}} + \frac{1}{n}(\boldsymbol{\tau}, \nabla u)_{\Omega_{\Gamma_{in}}} - \langle \boldsymbol{\tau} \cdot \mathbf{n}, u \rangle_{\Gamma} = -\langle \boldsymbol{\tau} \cdot \mathbf{n}, \bar{u} \rangle_{\Gamma} \quad (19)$$

This formulation still contains the primal unknown u and the flux variable $\boldsymbol{\sigma}$. However, as we shall see later, $\boldsymbol{\sigma}$ can be condensed out at the discrete level in such a way that no additional unknowns will be necessary to impose the boundary conditions.

3.2.2. *Discretization in the context of the finite cell method.* The finite element subspaces are constructed to ensure that $V_h \subset H^1(\Omega_h)^d$, so V_h consists of continuous functions. Further, we construct $S_h \subset L^2(\Omega_h)^d$. Functions in S_h are considered to be discontinuous only within the elements cut by Γ and are zero elsewhere. The discretized weak form given in eqs. (18) and (19) now reads:

$$k(\nabla u_h, \nabla v_h)_{\Omega} - \langle \mathbf{n} \cdot \boldsymbol{\sigma}_h, v_h \rangle_{\Gamma} + \frac{1}{n}(\nabla v_h, \boldsymbol{\sigma}_h)_{\Omega_{\Gamma_{in}}} - \frac{1}{n}k(\nabla v_h, \nabla u_h)_{\Omega_{\Gamma_{in}}} = \langle f, v_h \rangle_{\Omega} \quad \forall v_h \in V_h \quad (20)$$

$$-\frac{1}{nk}(\boldsymbol{\tau}_h, \boldsymbol{\sigma}_h)_{\Omega_{\Gamma_{in}}} + \frac{1}{n}(\boldsymbol{\tau}_h, \nabla u_h)_{\Omega_{\Gamma_{in}}} - \langle \boldsymbol{\tau}_h \cdot \mathbf{n}, u_h \rangle_{\Gamma} = -\langle \boldsymbol{\tau}_h \cdot \mathbf{n}, \bar{u} \rangle_{\Gamma} \quad \forall \boldsymbol{\tau}_h \in S_h \quad (21)$$

At this point it is worthwhile to discuss the meaning of discontinuity in the context of the finite cell method. The primary goal is to simplify the implementation process as far as possible by reusing the shape functions from V_h for S_h . In the context of the finite cell method, the first term in eq. (20) reads

$$(\nabla u_h, \nabla v_h)_{\Omega_e}^{\alpha} = \sum_{c=1}^{n_c} \int_{\Omega_c} [\mathbf{L}v_h]^T \alpha \mathbf{C} [\mathbf{L}u_h] d\Omega \quad (22)$$

The following, alternative view is naturally possible:

$$(\nabla u_h, \nabla v_h)_{\Omega_e}^{\alpha} = \sum_{c=1}^{n_c} \int_{\Omega_c} [\mathbf{L}v_h \sqrt{\alpha}]^T \mathbf{C} [\mathbf{L}u_h \sqrt{\alpha}] d\Omega \quad (23)$$

Hereby, the shape and the weighting functions u and v are both multiplied by the square root of the indicator function α defined in eq. (2). We now have two views on the discontinuity of u and v :

variational eqn.	$k (\nabla u_h, \nabla v_h)_{\Omega, \Omega_{\Gamma in}}$	$-\frac{1}{nk} (\boldsymbol{\tau}_h, \boldsymbol{\sigma})_{\Omega_{\Gamma in}}$	$\frac{1}{n} (\boldsymbol{\tau}_h, \nabla u)_{\Omega_{\Gamma in}^I}$	$\frac{1}{n} (\nabla v_h, \boldsymbol{\sigma}_h)_{\Omega_{\Gamma in}}$
algebraic form	$\mathbf{K}_{uu} \mathbf{U}$	$\mathbf{K}_{\sigma\sigma} \boldsymbol{\Sigma}$	$\mathbf{K}_{\sigma u} \mathbf{U}$	$\mathbf{K}_{u\sigma} \boldsymbol{\Sigma}$
variational eqn.	$-\langle \boldsymbol{\sigma}_h \cdot \mathbf{n}, v_h \rangle_{\Gamma}$	$-\langle \boldsymbol{\tau}_h \cdot \mathbf{n}, u \rangle_{\Gamma}$	$-\langle \mathbf{n} \cdot \boldsymbol{\tau}_h, \bar{u} \rangle_{\Gamma}$	$\langle f, v_h \rangle_{\Omega}$
algebraic form	$\mathbf{G}_{u\sigma} \boldsymbol{\Sigma}$	$\mathbf{G}_{\sigma u} \mathbf{U}$	$\mathbf{g}_{\sigma \bar{u}}$	\mathbf{f}

Table I. Terms concerned with the internal part of each domain of the discrete variational and algebraic form of eqs. (20) and (21).

(a) we may view them to *smoothly extend* across Γ into the embedding domain under the condition that this extension does not contribute to the energy described by the bilinear form. Clearly, this view is expressed in eq. (22). (b) we might view the shape functions u_h and v_h to exhibit a jump to zero from whatever value that satisfies the conditions on Γ internal to Ω_{C3} . This view is represented in eq. (23). From the implementational point of view there is, thus, no need to define a new set of discontinuous shape functions for the discretization of either $\boldsymbol{\sigma}_h$ or $\boldsymbol{\tau}_h$. Instead, one may simply reuse the shape functions from the discretization of u_h and v_h for the discretization of $\boldsymbol{\sigma}_h$ and $\boldsymbol{\tau}_h$. Thus, we discretize u_h and v_h in a Bubnov-Galerkin sense as follows:

$$u_h = \mathbf{N}\mathbf{U} = \begin{bmatrix} N_{i=1} & \dots & N_{i=n} \end{bmatrix} \begin{bmatrix} \hat{u}_{i=1} \\ \vdots \\ \hat{u}_{i=n} \end{bmatrix} \quad (24)$$

$$v_h = \mathbf{N}\mathbf{V} = \begin{bmatrix} N_{i=1} & \dots & N_{i=n} \end{bmatrix} \begin{bmatrix} \hat{v}_{i=1} \\ \vdots \\ \hat{v}_{i=n} \end{bmatrix} \quad (25)$$

$$\boldsymbol{\sigma}_h = \mathbf{N}\boldsymbol{\Sigma} = \begin{bmatrix} N_{i=1} & \dots & N_{i=n} \\ N_{i=1} & \dots & N_{i=n} \end{bmatrix} \begin{bmatrix} \hat{\sigma}_{i=1x} & \hat{\sigma}_{i=1y} \\ \vdots & \vdots \\ \hat{\sigma}_{i=nx} & \hat{\sigma}_{i=ny} \end{bmatrix} \quad (26)$$

$$\boldsymbol{\tau}_h = \mathbf{n}\mathbf{N}\mathbf{T} = \begin{bmatrix} \hat{n}_1 \\ \hat{n}_2 \end{bmatrix} \begin{bmatrix} N_{i=1} & \dots & N_{i=n} \end{bmatrix} \begin{bmatrix} \hat{\tau}_{i=1x} & \hat{\tau}_{i=1y} \\ \vdots & \vdots \\ \hat{\tau}_{i=nx} & \hat{\tau}_{i=ny} \end{bmatrix} \quad (27)$$

for scalar-valued Poisson's problem. In eqs. (24) to (27) N_i denote the i^{th} shape function and $n = (p+1)^d$ for a tensor product space in d dimensions. The corresponding coefficients $\hat{u}_i, \hat{v}_i, \hat{\sigma}_{ix}, \hat{\sigma}_{iy}, \hat{\tau}_{ix}, \hat{\tau}_{iy}$ are organized in the respective matrices \mathbf{U} , \mathbf{V} , $\boldsymbol{\Sigma}$, and \mathbf{T} . We would like to emphasize that, in the following, we will use an equal order discretization for all fields and that, therefore, all N 's given in eqs. (24) to (27) are completely identical.

Note that the definition of $\boldsymbol{\tau}_h$ includes the multiplication by the normal vector \mathbf{n} . We apply the following definition for \mathbf{n} : \mathbf{n} at Γ_D is the outward pointing normal vector. Logically, then, inside $\Omega_{\Gamma in}$, \mathbf{n} is defined to deliver a smooth extension of Γ into $\Omega_{\Gamma in}$ which is obtained by taking \mathbf{n} to be the normal vector towards Γ from the considered point in $\Omega_{\Gamma in}^\dagger$.

3.2.3. *Static condensation and final formulation.* With the definitions given in table I the system given by eqs. (20) and (21) can be written in compact form as:

$$\begin{bmatrix} \left(1 - \frac{1}{n}\right) \mathbf{K}_{uu} & \mathbf{K}_{u\sigma} + \mathbf{G}_{u\sigma} \\ \mathbf{K}_{\sigma u} + \mathbf{G}_{\sigma u} & \mathbf{K}_{\sigma\sigma} \end{bmatrix} \begin{bmatrix} \mathbf{U} \\ \boldsymbol{\Sigma} \end{bmatrix} = \begin{bmatrix} \mathbf{f} \\ \mathbf{g}_{\sigma \bar{u}} \end{bmatrix} \quad (28)$$

[†]This definition is applied, for example, in the evaluation of $\mathbf{K}_{\sigma\sigma}$ in eqs. (35), (53) and (54).

variational eqn.	$-k (\mathbf{n} \cdot \nabla u, v)_\Gamma$	$\frac{\beta k}{h} (v_h, u_h)_\Gamma$	$-k (\mathbf{n} \nabla v_h, \bar{u})_\Gamma$	$\frac{\beta k}{h} (v_h, \bar{u})_\Gamma$
algebraic form	$\mathbf{G}_{uu} \mathbf{U}$	$\mathbf{G}_{uu}^\beta \mathbf{U}$	$\mathbf{g}_{u\bar{u}}$	$\mathbf{g}_{u\bar{u}}^\beta$

Table II. Nitsche's method: Linear and bilinear forms used in eq. (36).

The fluxes may be computed as follows:

$$\boldsymbol{\Sigma} = \mathbf{K}_{\sigma\sigma}^{-1} (-(\mathbf{K}_{\sigma u} + \mathbf{G}_{\sigma u}) \mathbf{U} + \mathbf{g}_{\sigma\bar{u}}) \quad (29)$$

$\mathbf{K}_{\sigma\sigma}$ is always positive definite and, therefore, invertible. Additionally, $\mathbf{K}_{\sigma\sigma}$ must only be computed on the elements cut by Γ . The fluxes may, thus, be condensed out of eq. (28) analytically on element level prior to computation. Equation (28), therefore, simplifies to:

$$\left[\left(1 - \frac{1}{n}\right) \mathbf{K}_{uu} - (\mathbf{G}_{u\sigma} + \mathbf{K}_{u\sigma}) \mathbf{K}_{\sigma\sigma}^{-1} (\mathbf{K}_{\sigma u} + \mathbf{G}_{\sigma u}) \right] \mathbf{U} = [\mathbf{f} - (\mathbf{G}_{u\sigma} + \mathbf{K}_{u\sigma}) \mathbf{K}_{\sigma\sigma}^{-1} \mathbf{g}_{\sigma\bar{u}}] \quad (30)$$

The system given in eq. (30) can be written in a simpler way with the help of the following identities:

$$\frac{1}{n} \mathbf{K}_{uu} \mathbf{U} = - [\mathbf{K}_{u\sigma} \mathbf{K}_{\sigma\sigma}^{-1} \mathbf{K}_{\sigma u}] \mathbf{U} \quad (31)$$

$$\mathbf{G}_{uu}^T \mathbf{U} = - [\mathbf{K}_{u\sigma} \mathbf{K}_{\sigma\sigma}^{-1} \mathbf{G}_{\sigma u}] \mathbf{U} \quad (32)$$

$$\mathbf{g}_{u\bar{u}} \mathbf{U} = - [\mathbf{K}_{u\sigma} \mathbf{K}_{\sigma\sigma}^{-1} \mathbf{g}_{\sigma\bar{u}}] \quad (33)$$

$$\mathbf{G}_{uu} \mathbf{U} = - [\mathbf{G}_{u\sigma} \mathbf{K}_{\sigma\sigma}^{-1} \mathbf{K}_{\sigma u}] \mathbf{U} \quad (34)$$

to give:

$$[\mathbf{K}_{uu} + \mathbf{G}_{uu} + \mathbf{G}_{uu}^T - \mathbf{G}_{u\sigma} \mathbf{K}_{\sigma\sigma}^{-1} \mathbf{G}_{\sigma u}] \mathbf{U} = [\mathbf{f} + \mathbf{g}_{u\bar{u}} - \mathbf{G}_{u\sigma} \mathbf{K}_{\sigma\sigma}^{-1} \mathbf{g}_{\sigma\bar{u}}] \quad (35)$$

Equation (35) is the final formulation of the method presented in [33]. For means of comparison, the discrete form of Nitsche's method is given in eq. (36)

$$[\mathbf{K}_{uu} + \mathbf{G}_{uu} + \mathbf{G}_{uu}^T + \mathbf{G}_{uu}^\beta] \mathbf{U} = [\mathbf{f} + \mathbf{g}_{u\bar{u}} + \mathbf{g}_{u\bar{u}}^\beta] \quad (36)$$

where the linear forms of $\mathbf{g}_{u\bar{u}}$ and $\mathbf{g}_{u\bar{u}}^\beta$ and the bilinear forms of \mathbf{G}_{uu}^T and \mathbf{G}_{uu}^β are given in table II. Now it can be clearly seen that eq. (35) and eq. (36) only differ in the last terms in the brackets. The new terms $\mathbf{G}_{u\sigma} \mathbf{K}_{\sigma\sigma}^{-1} \mathbf{G}_{\sigma u}$ and $\mathbf{G}_{u\sigma} \mathbf{K}_{\sigma\sigma}^{-1} \mathbf{g}_{\sigma\bar{u}}$ are now responsible for counterbalancing the Nitsche terms resulting from the identified Lagrange multiplier \mathbf{G}_{uu} and its transpose. Stability of eq. (35) was proven in the sense of an inf-sup condition in [33] for low order discretizations. We will provide examples to show that the system given by eq. (35) even remained positive definite.

It is interesting to note how the method given by eq. (35) avoids the estimation of a problem-dependent parameter. Let us consider the simple case in which $\Omega_{\Gamma_{in}}$ is small. Then, the magnitude of $\mathbf{K}_{\sigma\sigma}^{-1}$ is large and the term $\mathbf{G}_{u\sigma} \mathbf{K}_{\sigma\sigma}^{-1} \mathbf{G}_{\sigma u}$ automatically adds the needed stabilization. In this process, the dependency of the penalty term on a characteristic mesh size h and a specific p is respected quite naturally.

3.3. Two-dimensional Poisson's problem

In this section, we investigate the 2D Poisson's problem proposed in [70]. The same example was studied in the context of the finite cell method in [28] as well, and is depicted in fig. 3. It consists of a square domain with edge $a = 1$ and $k = 1$. Homogeneous Dirichlet boundary conditions are imposed on the left, right and upper boundary of the domain while $u(x, 0) = \sin(\pi x)$ is imposed on the lower boundary. We discretize the domain by 8×8 elements using both the p -version and the B-Spline version of the finite cell method and study pure p -refinements, keeping the discretization grid as it is. All boundary conditions are imposed weakly as presented above either by the Nitsche's

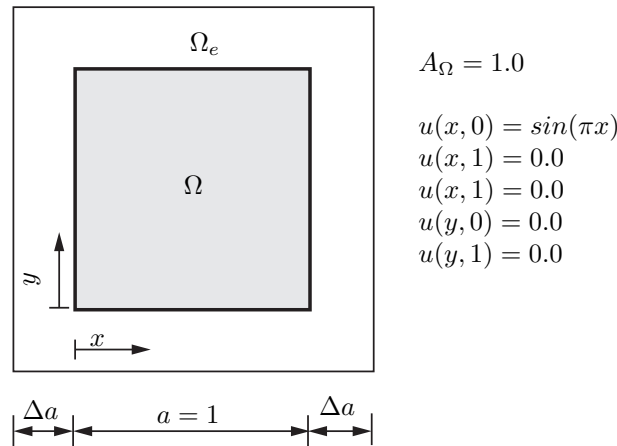


Figure 3. two-dimensional Poisson's problem as proposed in [70].

method or by eq. (35). For Nitsche's method, we estimate the penalty parameter for each polynomial degree as suggested in [69] and multiply it by two. We expect the solution to converge to the analytical one given in eq. (37) which has the strain energy given by eq. (38):

$$U_{ex}(x, y) = (\cosh(\pi y) - \coth(\pi) \sinh(\pi y)) \sin(\pi x) \quad (37)$$

$$U_{ex} = \frac{k}{4} \pi \coth(\pi) \quad (38)$$

With no singularities in the solution, we expect an exponential convergence in the energy norm under pure p -refinement.

We begin by studying the case where the discretization of the domain conforms to all boundaries. A first overall impression of the solution is given in fig. 4a, where the case $p = 3$ is considered. The exact solution is not depicted as the approximation is already so close to it. More insight is only gained by looking at the convergence in the strain energy norm, as depicted in fig. 4b. All methods converge exponentially as expected. With no even-odd behavior present, almost no difference in the p - or B-Spline version of the finite cell method is observable. This is not surprising as we are performing a pure p -extension on both meshes starting with $p=1$ and, therefore, stay C^0 continuous at inter-element level for both discretization schemes. The absolute difference of the analytic strain energy to its numeric approximation is in the order of 10^{-12} at $p = 8$. Figure 4c depicts the condition number of the stiffness matrix. It is of no surprise that the p -version has a clear advantage over B-Splines in terms of conditioning for this boundary conforming example. The minimum eigenvalues of the overall stiffness matrix including its weak boundary conditions are depicted in fig. 4d. They decrease to small values but always stay positive.

Additionally, we provide a convergence study concerning an h -extension in Figure 5. While Nitsche's method and the method given by Equation (35) possess the same slopes, we found the method given by Equation (35) to consistently provide a slightly better constant.

We now look at the performance of the boundary conditions for an embedded problem which we solve using the finite cell method. We use the same mesh consisting of a single patch with 8×8 elements for the B-Spline version or 8×8 p -elements. The discretization is plotted along with the integration mesh and the solution in fig. 6a. Note that there are now all three types of cells Ω_{C1} , Ω_{C2} , and Ω_{C3} and that the computations are carried out as described in section 2: the cells in Ω_{C1} are located completely outside the physical domain and are neither computed nor assembled. The

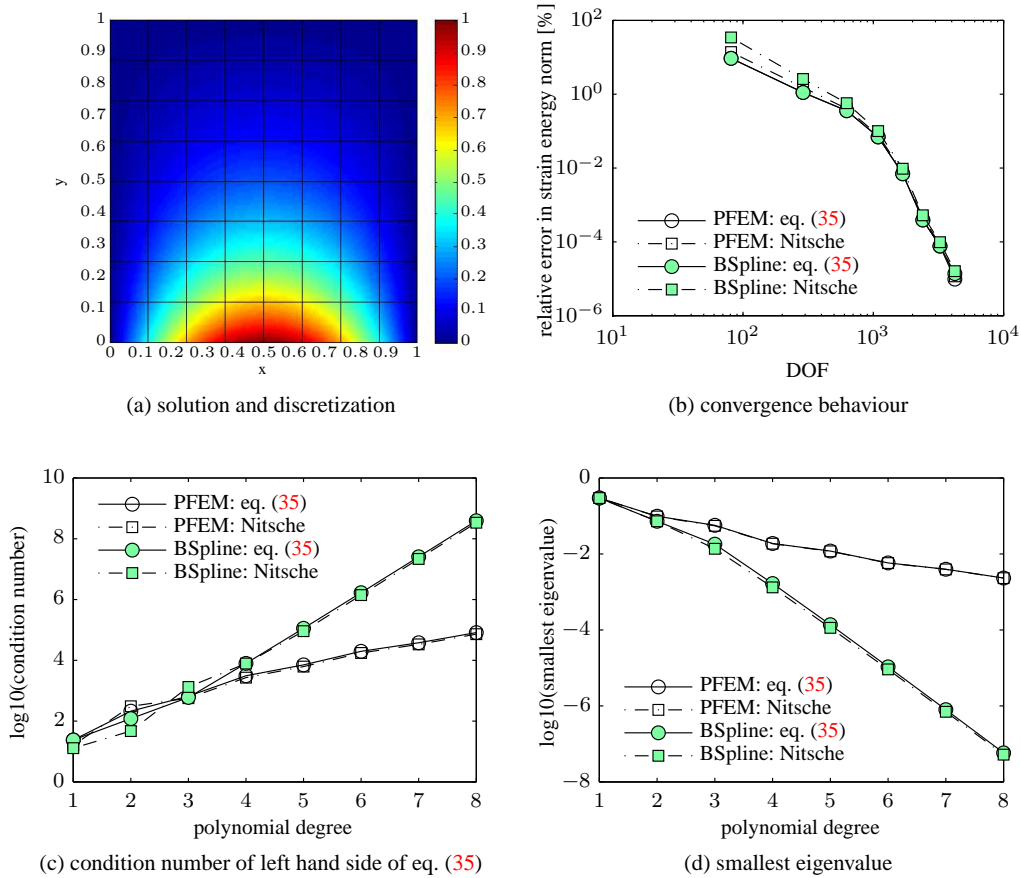


Figure 4. 2D $\Delta a = 0$ (conforming boundary conditions): 8×8 elements. Each marker symbolizes the result for $p=1 \dots 8$, respectively

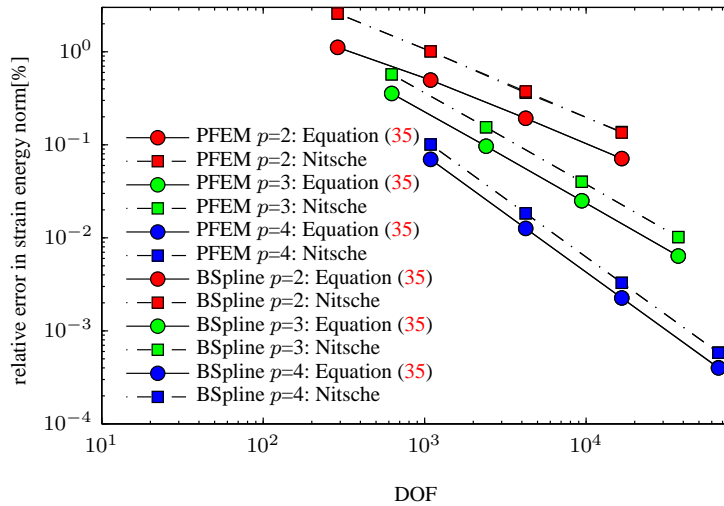
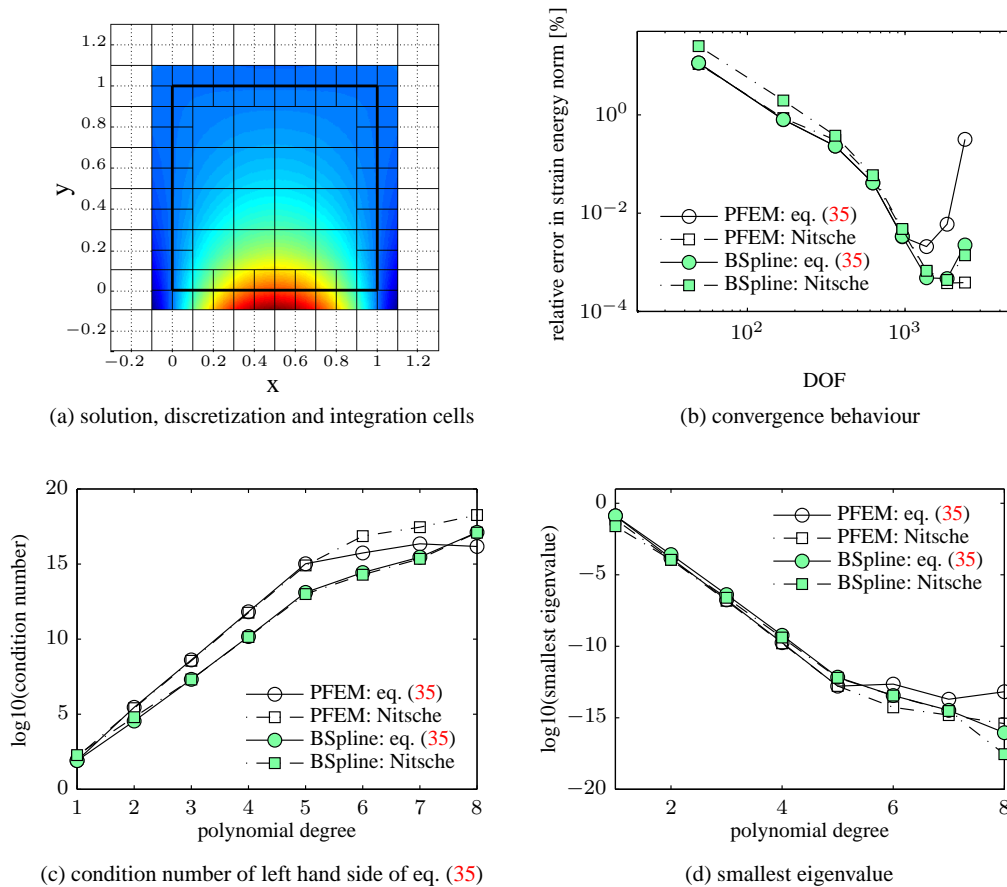


Figure 5. 2D $\Delta a = 0$ (conforming boundary conditions): h -refinement for different p . Each marker symbolizes the result for $8 \times 8, 16 \times 16, 32 \times 32$, and 64×64 elements, of degree $p = 2 \dots 4$, respectively

Figure 6. *non-conforming boundary*

cells in Ω_{C2} are computed as regular finite elements of high order. The cells in Ω_{C3} are adaptively integrated. As we chose a very favorable situation — namely that the boundary Γ cuts the cells in Ω_{C3} exactly in half — an adaptive integration on a quadtree is chosen, leading to an exact integration already at the first level of refinement. Figure 6a explicitly depicts the entire solution, including the integration tree and the unphysical solution in the fictitious domain $\Omega_{\Gamma_{out}}$ outside the boundary depicted by a thick black line. The physical solution is similar to the non-embedded case, only the color-scale is shifted due to the smooth extension. The convergence behavior as percent error in the energy norm is depicted in fig. 6b, whereby the reference solution is the strain energy computed analytically from the solution given in eq. (37). The convergence is of higher order until $p=6$, where it levels off and, in case of the method given by eq. (35) version, increases. The increase is non-physical and clearly shows the limits of the numerics to compute this situation. At this level, the condition number depicted in fig. 6c rises above 10^{15} which imposes difficulties for the solver used. This deserves a closer look. Apart from the inversion of the stiffness matrix \mathbf{K} , there are more places where matrices have to be inverted. For the Nitsche's version of imposing boundary conditions, one must solve the eigenvalue problem given in eq. (13) as accurately as possible. The eigenvalue problem is defined on the cut cells and therefore suffers from the conditioning problems as well.

The crucial point for the method given in eq. (35) is the inversion of $\mathbf{K}_{\sigma\sigma}$, also defined on the cut cells. It constitutes the core part of the lower bound for the stability term. In the current Matlab implementation of the method, the inversion of $\mathbf{K}_{\sigma\sigma}$ is most stable using the pseudo inverse instead of the inverse but it is still less accurate than the solution of the corresponding eigenvalue problem

necessary for Nitsche's method. It is worthwhile noting that $\mathbf{K}_{\sigma\sigma}$ is similar to a mass matrix and, therefore, neither basis is optimized to render optimal conditioning here. At $p=7$, the condition of $\mathbf{K}_{\sigma\sigma}$ reaches 10^{18} and a correct solution is no longer guaranteed. However, all eigenvalues of the overall stiffness matrix of the discrete system are very small but always remain positive as depicted in fig. 6d.

3.4. Curved boundary

We compute the example depicted in fig. 7. It consists of a Poisson problem solved on a quarter of an annulus, embedded in a Cartesian mesh. Two Dirichlet boundary conditions are set weakly at the inner and outer ring to 1 and 3, respectively. The analytical solution in polar coordinates is given by

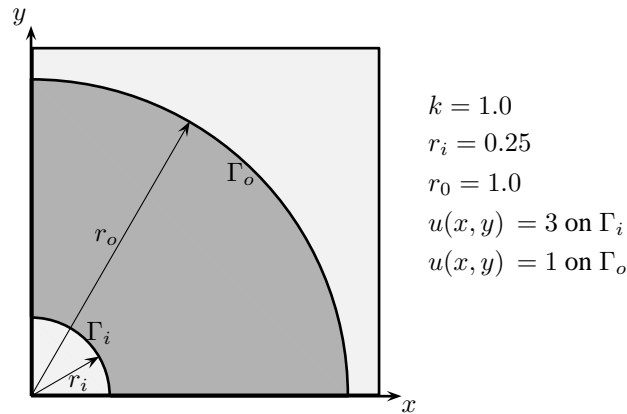


Figure 7. Quarter of an annulus.

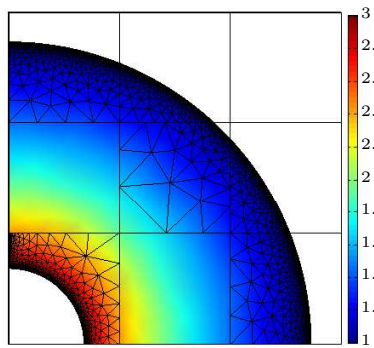
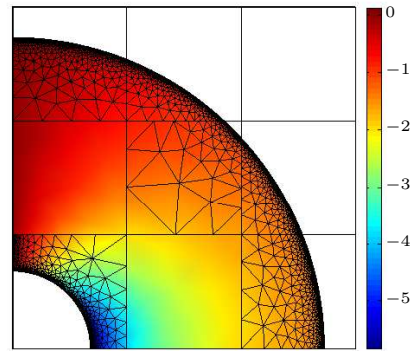
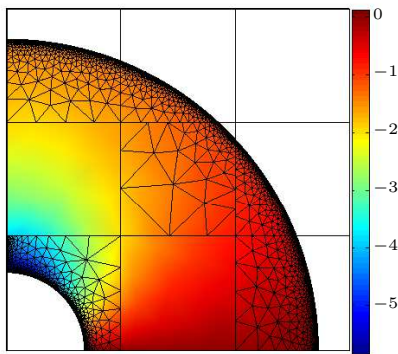
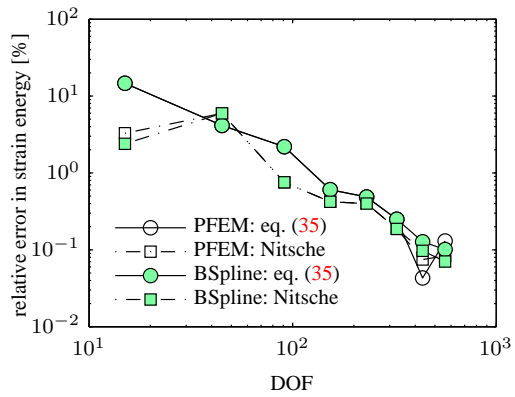
$$u(r) = 1 - \frac{\ln(r)}{\ln(2)} \quad (39)$$

where r denotes the radial distance. It possesses a strain energy of

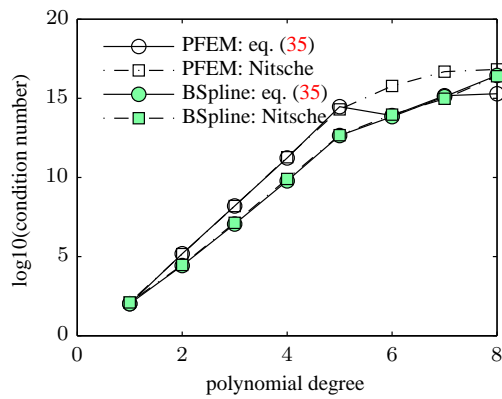
$$U_{ex} = \frac{\pi}{\ln(4)} \approx 2.266180071.$$

The results are plotted in fig. 8. In addition to the Cartesian mesh spanning the shape functions, we plot the integration mesh in figs. 8a to 8c. Note that, instead of the adaptive sub-cells we use sub-triangles here to integrate the cut-cells. This increases the accuracy of the integration and the solution, see e.g.[28]. The results are snapshots for $p = 3$ whereby we do not plot the smooth extension for the sake of clarity.

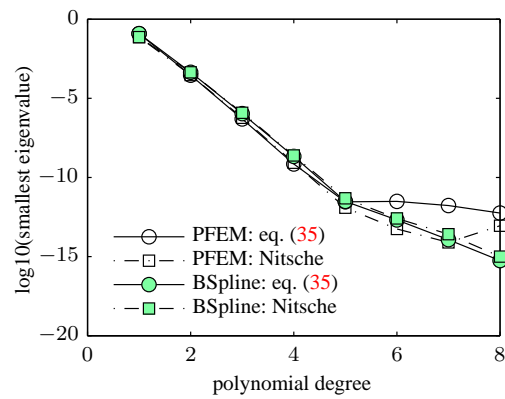
The error in the energy norm, the condition number and the smallest eigenvalues are depicted in figs. 8d to 8f, respectively. The new method practically delivers results of comparable accuracy and conditioning for non-conforming, curved boundaries.

(a) Numerical solution for $p = 3$, discretization, and integration domains(b) Numerical flux for $p = 3$ in x-direction(c) Numerical flux for $p = 3$ in y-direction

(d) convergence behaviour



(e) Condition number of left hand side of eq. (35)



(f) Smallest eigenvalue

Figure 8. Example: Quarter ring, results.

4. PARAMETER-FREE, WEAK COUPLING OF TWO DOMAINS

4.1. General setting

In the following, we consider two domains Ω^I and Ω^{II} , which are sewed together along the common boundary $\Gamma = \Gamma^I = \Gamma^{II}$. On this common boundary, we will demand the continuity of the displacements as well as the continuity of the fluxes in a weak form. The continuous view of this general setting is depicted in fig. 9. The coupling conditions of Ω^I and Ω^{II} at the common boundary

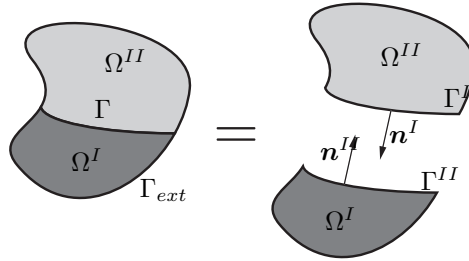


Figure 9. Two physical domains Ω^I and Ω^{II} weakly coupled at Γ .

Γ in strong form are:

$$u^I - u^{II} = 0 \quad \text{on } \Gamma \quad (40)$$

$$\boldsymbol{\sigma}^I \cdot \mathbf{n}^I - \boldsymbol{\sigma}^{II} \cdot \mathbf{n}^{II} = 0 \quad \text{on } \Gamma \quad (41)$$

4.2. Weak coupling using Nitsche's method

Nitsche's method applied to the coupled Poisson's system is given by:

$$k^I (\nabla u^I, \nabla v^I)_{\Omega^I} - \frac{1}{2} (\mathbf{n}^I \cdot k^I \nabla u^I - \mathbf{n}^{II} \cdot k^{II} \nabla u^{II}, v^I)_{\Gamma} - \frac{1}{2} (u^I, \mathbf{n}^I \cdot k^I \nabla v^I - \mathbf{n}^{II} \cdot k^{II} \nabla v^{II})_{\Gamma} + \beta (u^I - u^{II}, v^I)_{\Gamma} = \langle f, v^I \rangle_{\Omega^I} \quad (42)$$

$$k^{II} (\nabla u^{II}, \nabla v^{II})_{\Omega^{II}} - \frac{1}{2} (\mathbf{n}^{II} \cdot k^{II} \nabla u^{II} - \mathbf{n}^I \cdot k^I \nabla u^I, v^{II})_{\Gamma} - \frac{1}{2} (u^{II}, \mathbf{n}^{II} \cdot k^{II} \nabla v^{II} - \mathbf{n}^I \cdot k^I \nabla v^I)_{\Gamma} + \beta (u^{II} - u^I, v^{II})_{\Gamma} = \langle f, v^{II} \rangle_{\Omega^{II}} \quad (43)$$

where the weak boundary conditions on $\Gamma_{ext} = (\partial\Omega^I \cup \partial\Omega^{II}) \setminus \Gamma$ are not written down for the sake of simplicity. The use of Nitsche's method for coupling domains weakly has already extensively been studied, for example in [39]. We refer to [6], for an overview of its properties and to the introduction for further references. Again, an estimator for β is needed and we use the one provided by eq. (13), wherein now

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}^I & \mathbf{0} \\ \mathbf{0} & \mathbf{A}^{II} \end{bmatrix} \quad (44)$$

and likewise for \mathbf{B} .

4.3. New formulation for parameter-free, symmetry-preserving weak coupling

We first reconsider the weak form given in eqs. (20) and (21) and note that the parts of the equations on Ω and $\Omega_{\Gamma_{in}}$ only concern the interior Ω . They may thus be written independently

Discrete variational eqn.	$-\langle \boldsymbol{\sigma}^{II} \cdot \mathbf{n}^I, v_h^I \rangle_\Gamma$	$-\langle \boldsymbol{\sigma}^I \cdot \mathbf{n}^{II}, v_h^{II} \rangle_\Gamma$
Algebraic form	$\mathbf{G}_{u\sigma}^{I,II} \boldsymbol{\Sigma}^{II}$	$\mathbf{G}_{u\sigma}^{II,I} \boldsymbol{\Sigma}^I$
Discrete variational eqn.	$-\langle \boldsymbol{\tau}_h^I \cdot \mathbf{n}^{II}, u^{II} \rangle_\Gamma$	$-\langle \boldsymbol{\tau}_h^{II} \cdot \mathbf{n}^I, u^I \rangle_\Gamma$
Algebraic form	$\mathbf{G}_{\sigma u}^{I,II} \mathbf{U}^{II}$	$\mathbf{G}_{\sigma u}^{II,I} \mathbf{U}^I$

Table III. Coupling terms of the discrete variational and algebraic form eqs. (45) to (48).

of each other for each subdomain Ω^I and Ω^{II} . On the contrary, the coupling conditions at Γ i.e., eqs. (40) and (41), are concerned with both domains. For reasons of consistency, we equally distribute their contribution between both subdomains. Let us define the subspace for the unknowns in each subdomain as $V_h^I \subset H^1(\Omega_h^I)$ and $V_h^{II} \subset H^1(\Omega_h^{II})$. Similarly, we define $S_h^I \subset L^2(\Omega_h^I)$ and $S_h^{II} \subset L^2(\Omega_h^{II})$. Utilizing eqs. (40) and (41), the variational form of the problem consists of finding $u_h^I \in V_h^I$, $\boldsymbol{\sigma}_h^I \in S_h^I$, $u_h^{II} \in V_h^{II}$, $\boldsymbol{\sigma}_h^{II} \in S_h^{II}$ such that:

$$k (\nabla u_h^I, \nabla v_h^I)_{\Omega^I} - \frac{1}{2} \langle \boldsymbol{\sigma}_h^{II} \cdot \mathbf{n}^I + \boldsymbol{\sigma}_h^I \cdot \mathbf{n}^I, v_h^I \rangle_\Gamma + \frac{1}{n} (\nabla v_h^I, \boldsymbol{\sigma}_h^I)_{\Omega_{\Gamma_{in}}^I} + \quad (45)$$

$$- \frac{1}{n} k (\nabla v_h^I, \nabla u_h^I)_{\Omega_{\Gamma_{in}}^I} = \langle f, v_h \rangle_{\Omega^I} \quad \forall v_h^I \in V_h^I$$

$$- \frac{1}{nk} (\boldsymbol{\tau}_h^I, \boldsymbol{\sigma}_h^I)_{\Omega_{\Gamma_{in}}^I} + \frac{1}{n} (\boldsymbol{\tau}_h^I, \nabla u_h^I)_{\Omega_{\Gamma_{in}}^I} - \frac{1}{2} \langle \boldsymbol{\tau}_h^I \cdot \mathbf{n}^I, u_h^I - u_h^{II} \rangle_\Gamma = 0 \quad \forall \boldsymbol{\tau}_h^I \in S_h^I \quad (46)$$

$$k (\nabla u_h^{II}, \nabla v_h^{II})_{\Omega^{II}} - \frac{1}{2} \langle \boldsymbol{\sigma}_h^I \cdot \mathbf{n}^{II} + \boldsymbol{\sigma}_h^{II} \cdot \mathbf{n}^{II}, v_h^{II} \rangle_\Gamma + \frac{1}{n} (\nabla v_h^{II}, \boldsymbol{\sigma}_h^{II})_{\Omega_{\Gamma_{in}}^{II}} + \quad (47)$$

$$- \frac{1}{n} k (\nabla v_h^{II}, \nabla u_h^{II})_{\Omega_{\Gamma_{in}}^{II}} = \langle f, v_h^{II} \rangle_{\Omega^{II}} \quad \forall v_h^{II} \in V_h^{II}$$

$$- \frac{1}{nk} (\boldsymbol{\tau}_h, \boldsymbol{\sigma}_h)_{\Omega_{\Gamma_{in}}^{II}} + \frac{1}{n} (\boldsymbol{\tau}_h, \nabla u_h)_{\Omega_{\Gamma_{in}}^{II}} - \frac{1}{2} \langle \boldsymbol{\tau}_h^{II} \cdot \mathbf{n}^{II}, u_h^{II} - u_h^I \rangle_\Gamma = 0 \quad \forall \boldsymbol{\tau}_h^{II} \in S_h^{II} \quad (48)$$

where the coupling terms from the respective other domain are marked in blue. The first two equations correspond to the first subdomain, while the last two equations correspond to the second subdomain.

Let \mathbf{U} and $\boldsymbol{\Sigma}$ be the vectors of the unknown coefficients of u_h and $\boldsymbol{\sigma}_h$, respectively. For the algebraic version of eqs. (45) to (48) we utilize the notation given in table I for the terms only concerned with one domain. **They are identical to the case where only the boundary conditions were imposed with the difference that the force term $g_{\sigma\bar{u}}$ is missing. Its contribution now appears as additional terms in the stiffness matrix.** These new terms concerned with the sewing of the domains are collected in table III, where we have used the fact that $\mathbf{n}^I = -\mathbf{n}^{II}$. Further, we utilize the fact that $\mathbf{G}_{\sigma u}^{I,II} = \mathbf{G}_{u\sigma}^{II,I^T}$ and $\mathbf{G}_{\sigma u}^{II,I} = \mathbf{G}_{u\sigma}^{I,II^T}$. The coupled problem written in matrix form then reads:

$$\begin{bmatrix} (1 - \frac{1}{n}) \mathbf{K}_{uu}^I & \mathbf{K}_{u\sigma}^I + \frac{1}{2} \mathbf{G}_{u\sigma}^I & \mathbf{0} & \frac{1}{2} \mathbf{G}_{u\sigma}^{I,II} \\ \mathbf{K}_{\sigma u}^I + \frac{1}{2} \mathbf{G}_{\sigma u}^I & \mathbf{K}_{\sigma\sigma}^I & \frac{1}{2} \mathbf{G}_{\sigma u}^{I,II} & \mathbf{0} \\ \mathbf{0} & \frac{1}{2} \mathbf{G}_{u\sigma}^{II,I} & (1 - \frac{1}{n}) \mathbf{K}_{uu}^{II} & \mathbf{K}_{u\sigma}^{II} + \frac{1}{2} \mathbf{G}_{u\sigma}^{II} \\ \frac{1}{2} \mathbf{G}_{\sigma u}^{II,I} & \mathbf{0} & \mathbf{K}_{\sigma u}^{II} + \frac{1}{2} \mathbf{G}_{\sigma u}^{II} & \mathbf{K}_{\sigma\sigma}^{II} \end{bmatrix} \begin{bmatrix} \mathbf{U}^I \\ \boldsymbol{\Sigma}^I \\ \mathbf{U}^{II} \\ \boldsymbol{\Sigma}^{II} \end{bmatrix} = \begin{bmatrix} \mathbf{f}^I \\ \mathbf{0} \\ \mathbf{f}^{II} \\ \mathbf{0} \end{bmatrix} \quad (49)$$

4.4. Static condensation

We may compute the fluxes in eq. (49) as:

$$\boldsymbol{\Sigma}^I = (\mathbf{K}_{\sigma\sigma}^I)^{-1} \left(- \left(\mathbf{K}_{\sigma u}^I + \frac{1}{2} \mathbf{G}_{\sigma u}^I \right) \mathbf{U}^I - \frac{1}{2} \mathbf{G}_{\sigma u}^{I,II} \mathbf{U}^{II} \right) \quad (50)$$

variational eqn.	$-k^I \langle \nabla u_h^I \cdot \mathbf{n}^{II}, v_h^{II} \rangle_\Gamma$	$-k^{II} \langle \nabla u_h^{II} \cdot \mathbf{n}^I, v_h^I \rangle_\Gamma$
algebraic form	$\mathbf{G}_{uu}^{II,I} \mathbf{U}^I$	$\mathbf{G}_{uu}^{I,II} \mathbf{U}^{II}$
variational eqn.	$-k^I \langle \nabla v_h^I \cdot \mathbf{n}^{II}, u_h^{II} \rangle_\Gamma$	$-k^{II} \langle \nabla v_h^{II} \cdot \mathbf{n}^I, u_h^I \rangle_\Gamma$
algebraic form	$(\mathbf{G}_{uu}^{II,I})^T \mathbf{U}^{II}$	$(\mathbf{G}_{uu}^{I,II})^T \mathbf{U}^I$

Table IV. Coupling terms from Nitsche's method.

$$\Sigma^{II} = (\mathbf{K}_{\sigma\sigma}^{II})^{-1} \left(- \left(\mathbf{K}_{\sigma u}^{II} + \frac{1}{2} \mathbf{G}_{\sigma u}^{II} \right) \mathbf{U}^{II} - \frac{1}{2} \mathbf{G}_{\sigma u}^{II,I} \mathbf{U}^I \right) \quad (51)$$

and condense Σ^I and Σ^{II} out of the system eq. (49). The expression for subdomain I reads:

$$\begin{aligned} & \left(\left(1 - \frac{1}{n} \right) \mathbf{K}_{uu}^I - \left(\frac{1}{2} \mathbf{G}_{u\sigma}^I + \mathbf{K}_{u\sigma}^I \right) (\mathbf{K}_{\sigma\sigma}^I)^{-1} \left(\mathbf{K}_{\sigma u}^I + \frac{1}{2} \mathbf{G}_{\sigma u}^I \right) - \frac{1}{4} \mathbf{G}_{u\sigma}^{I,II} (\mathbf{K}_{\sigma\sigma}^{II})^{-1} \mathbf{G}_{\sigma u}^{II,I} \right) \mathbf{U}^I \\ & \left(- \left(\mathbf{K}_{u\sigma}^I + \frac{1}{2} \mathbf{G}_{u\sigma}^I \right) (\mathbf{K}_{\sigma\sigma}^I)^{-1} \frac{1}{2} \mathbf{G}_{\sigma u}^{II,I} - \frac{1}{2} \mathbf{G}_{u\sigma}^{I,II} (\mathbf{K}_{\sigma\sigma}^{II})^{-1} \left(\mathbf{K}_{\sigma u}^{II} + \frac{1}{2} \mathbf{G}_{\sigma u}^{II} \right) \right) \mathbf{U}^{II} \end{aligned} \quad (52)$$

We then use the definitions given in eqs. (31), (32) and (34) to simplify the terms only concerned with domain I. Expression (52) then reduces to:

$$\begin{aligned} & \left(\mathbf{K}_{uu}^I + \frac{1}{2} \left(\mathbf{G}_{uu}^I + (\mathbf{G}_{uu}^I)^T \right) - \frac{1}{4} \mathbf{G}_{u\sigma}^I (\mathbf{K}_{\sigma\sigma}^I)^{-1} \mathbf{G}_{\sigma u}^I - \frac{1}{4} \mathbf{G}_{u\sigma}^{I,II} (\mathbf{K}_{\sigma\sigma}^{II})^{-1} \mathbf{G}_{\sigma u}^{II,I} \right) \mathbf{U}^I \\ & \left(- \frac{1}{2} \left(\mathbf{K}_{u\sigma}^I + \frac{1}{2} \mathbf{G}_{u\sigma}^I \right) (\mathbf{K}_{\sigma\sigma}^I)^{-1} \mathbf{G}_{\sigma u}^{II,I} - \frac{1}{2} \mathbf{G}_{u\sigma}^{I,II} (\mathbf{K}_{\sigma\sigma}^{II})^{-1} \left(\mathbf{K}_{\sigma u}^{II} + \frac{1}{2} \mathbf{G}_{\sigma u}^{II} \right) \right) \mathbf{U}^{II} \end{aligned} \quad (53)$$

Likewise, for subdomain II we have:

$$\begin{aligned} & \left(- \frac{1}{2} \mathbf{G}_{u\sigma}^{II,I} (\mathbf{K}_{\sigma\sigma}^I)^{-1} \left(\mathbf{K}_{\sigma u}^I + \frac{1}{2} \mathbf{G}_{\sigma u}^I \right) - \frac{1}{2} \left(\mathbf{K}_{u\sigma}^{II} + \frac{1}{2} \mathbf{G}_{u\sigma}^{II} \right) (\mathbf{K}_{\sigma\sigma}^{II})^{-1} \mathbf{G}_{\sigma u}^{II,I} \right) \mathbf{U}^I \\ & \left(\mathbf{K}_{uu}^{II} - \frac{1}{4} \mathbf{G}_{u\sigma}^{II} (\mathbf{K}_{\sigma\sigma}^{II})^{-1} \mathbf{G}_{\sigma u}^{II} + \frac{1}{2} \left(\mathbf{G}_{uu}^{II} + (\mathbf{G}_{uu}^{II})^T \right) - \frac{1}{4} \mathbf{G}_{u\sigma}^{II,I} (\mathbf{K}_{\sigma\sigma}^I)^{-1} \mathbf{G}_{\sigma u}^{II,I} \right) \mathbf{U}^{II} \end{aligned} \quad (54)$$

Further, we introduce the following identities which are justified in [33]:

$$- \mathbf{K}_{u\sigma}^{II} (\mathbf{K}_{\sigma\sigma}^{II})^{-1} \mathbf{G}_{\sigma u}^{II,I} \mathbf{U}^I = (\mathbf{G}_{uu}^{I,II})^T \mathbf{U}^I \quad (55)$$

$$- \mathbf{G}_{u\sigma}^{II,I} (\mathbf{K}_{\sigma\sigma}^I)^{-1} \mathbf{K}_{\sigma u}^I \mathbf{U}^I = \mathbf{G}_{uu}^{II,I} \mathbf{U}^I \quad (56)$$

$$- \mathbf{K}_{u\sigma}^I (\mathbf{K}_{\sigma\sigma}^I)^{-1} \mathbf{G}_{\sigma u}^{II,I} \mathbf{U}^{II} = (\mathbf{G}_{uu}^{II,I})^T \mathbf{U}^{II} \quad (57)$$

$$- \mathbf{G}_{u\sigma}^{I,II} (\mathbf{K}_{\sigma\sigma}^{II})^{-1} \mathbf{K}_{\sigma u}^{II} \mathbf{U}^{II} = \mathbf{G}_{uu}^{I,II} \mathbf{U}^{II} \quad (58)$$

and correspond to terms arising from Nitsche's method as defined in table IV. Expression (52) then reads (terms in blue now mark the terms which differ from Nitsche's method, that is, the stabilization terms):

$$\begin{aligned} & \left(\mathbf{K}_{uu}^I + \frac{1}{2} \left(\mathbf{G}_{uu}^I + (\mathbf{G}_{uu}^I)^T \right) - \frac{1}{4} \mathbf{G}_{u\sigma}^I (\mathbf{K}_{\sigma\sigma}^I)^{-1} \mathbf{G}_{\sigma u}^I - \frac{1}{4} \mathbf{G}_{u\sigma}^{I,II} (\mathbf{K}_{\sigma\sigma}^{II})^{-1} \mathbf{G}_{\sigma u}^{II,I} \right) \mathbf{U}^I \\ & \left(\frac{1}{2} \left(\mathbf{G}_{uu}^{II,I} + (\mathbf{G}_{uu}^{II,I})^T \right) - \frac{1}{4} \mathbf{G}_{u\sigma}^I (\mathbf{K}_{\sigma\sigma}^I)^{-1} \mathbf{G}_{\sigma u}^{II,I} - \frac{1}{4} \mathbf{G}_{u\sigma}^{I,II} (\mathbf{K}_{\sigma\sigma}^{II})^{-1} \mathbf{G}_{\sigma u}^{II,I} \right) \mathbf{U}^{II} \end{aligned} \quad (59a)$$

Likewise, for subdomain II we have:

$$\left(\frac{1}{2} \left(\mathbf{G}_{uu}^{II,I} + (\mathbf{G}_{uu}^{II,I})^T \right) - \frac{1}{4} \mathbf{G}_{u\sigma}^{II,I} (\mathbf{K}_{\sigma\sigma}^I)^{-1} \mathbf{G}_{\sigma u}^I - \frac{1}{4} \mathbf{G}_{u\sigma}^{II} (\mathbf{K}_{\sigma\sigma}^{II})^{-1} \mathbf{G}_{\sigma u}^{II,I} \right) \mathbf{U}^I$$

$$\left(\mathbf{K}_{uu}^{II} + \frac{1}{2} \left(\mathbf{G}_{uu}^{II} + (\mathbf{G}_{uu}^{II})^T \right) - \frac{1}{4} \mathbf{G}_{u\sigma}^{II} (\mathbf{K}_{\sigma\sigma}^{II})^{-1} \mathbf{G}_{\sigma u}^{II} - \frac{1}{4} \mathbf{G}_{u\sigma}^{II,I} (\mathbf{K}_{\sigma\sigma}^I)^{-1} \mathbf{G}_{\sigma u}^{I,II} \right) \mathbf{U}^{II} \quad (59b)$$

It is interesting to note that the penalty terms have a contribution which arises from subdomain I and a contribution from subdomain II. They ensure the stability of the method independently of the geometry of the meshes in subdomain I and II, as proven in section 4.5.

4.5. Stability Analysis

In this subsection we prove that the formulation for weak coupling given by eqs. (59a) and (59b) is stable. To do so, we follow a strategy similar to the one presented in [33]. In order to show the stability of the method for weak coupling, we will consider the unknowns u^I to vanish on the external boundary Γ^{ext} of subdomain I, and the normal component of the stresses σ^{II} to be null in the external boundary in subdomain II. Further, to keep the proof as concise as possible, we consider only the case in which the material constants k are the same for both domains. Additionally, we assume the common boundary Γ^{ext} to be a straight line.

$$u^I = 0 \quad \text{on } \Gamma^{ext} \quad (60)$$

$$\sigma^{II} \cdot \mathbf{n}^{II} = 0 \quad \text{on } \Gamma^{ext} \quad (61)$$

In this setting, we redefine $V_h^I \subset H_0^1(\Omega_h^I)$, where the zero subscript denotes functions vanishing in the external boundary Γ^{ext} . We define the norm:

$$\begin{aligned} & \| [u^I, \sigma^I, u^{II}, \sigma^{II}] \|^2 = \\ & k \|\nabla u^I\|_{L^2(\Omega^I)}^2 + k \|\nabla u^{II}\|_{L^2(\Omega^{II})}^2 + \frac{k}{h} \|u^I - u^{II}\|_{L^2(\Gamma)}^2 + \frac{1}{k} \|\sigma^I\|_{L^2(\Omega_{\Gamma, in}^I)}^2 + \frac{1}{k} \|\sigma^{II}\|_{L^2(\Omega_{\Gamma, in}^{II})}^2 \end{aligned} \quad (62)$$

where h is the cell size. The bilinear form for the global problem defined on both subdomains is:

$$\begin{aligned} & B([u_h^I, \sigma_h^I, u_h^{II}, \sigma_h^{II}], [v_h^I, \tau_h^I, v_h^{II}, \tau_h^{II}]) = \\ & k (\nabla u_h^I, \nabla v_h^I)_{\Omega^I} - \frac{1}{2} \langle \sigma_h^{II} \cdot \mathbf{n}^I + \sigma_h^I \cdot \mathbf{n}^I, v_h^I \rangle_{\Gamma} + \frac{1}{n} (\nabla v_h^I, \sigma_h^I)_{\Omega_{\Gamma, in}^I} - \frac{1}{n} k (\nabla v_h^I, \nabla u_h^I)_{\Omega_{\Gamma, in}^I} + \\ & \quad - \frac{1}{nk} (\tau_h^I, \sigma_h^I)_{\Omega_{\Gamma, in}^I} + \frac{1}{n} (\tau_h^I, \nabla u_h^I)_{\Omega_{\Gamma, in}^I} - \frac{1}{2} \langle \tau_h^I \cdot \mathbf{n}^I, u_h^I - u_h^{II} \rangle_{\Gamma} + \\ & k (\nabla u_h^{II}, \nabla v_h^{II})_{\Omega^{II}} - \frac{1}{2} \langle \sigma_h^I \cdot \mathbf{n}^{II} + \sigma_h^{II} \cdot \mathbf{n}^{II}, v_h^{II} \rangle_{\Gamma} + \frac{1}{n} (\nabla v_h^{II}, \sigma_h^{II})_{\Omega_{\Gamma, in}^{II}} - \frac{1}{n} k (\nabla v_h^{II}, \nabla u_h^{II})_{\Omega_{\Gamma, in}^{II}} + \\ & \quad - \frac{1}{nk} (\tau_h^{II}, \sigma_h^{II})_{\Omega_{\Gamma, in}^{II}} + \frac{1}{n} (\tau_h^{II}, \nabla u_h^{II})_{\Omega_{\Gamma, in}^{II}} - \frac{1}{2} \langle \tau_h^{II} \cdot \mathbf{n}^{II}, u_h^{II} - u_h^I \rangle_{\Gamma} \end{aligned} \quad (63)$$

We suppose that V_h and S_h are such that

$$\forall v_h \in V_h \quad \exists \tau_h \in S_h \quad | \quad \delta_1 \|v_h\|_{L^2(\Gamma)}^2 \leq \langle \tau_h \cdot \mathbf{n}, v_h \rangle_{\Gamma} + \delta_0 h \|\nabla v_h\|^2 \quad (64)$$

$$\|\tau_h\|_{L^2(\Gamma)} = \|v_h\|_{L^2(\Gamma)}, \quad \|\tau_h\|^2 \leq \delta_2 h \|v_h\|_{L^2(\Gamma)}^2 \quad (65)$$

where $\delta_0, \delta_1, \delta_2$ are positive non-dimensional constants which depend on the geometry of the mesh. Conditions in eqs. (64) and (65) are an assumption of the formulation, but these conditions hold for the most common interpolation spaces. **As shown in [33], the inf-sup conditions hold for:**

- **Equal order interpolation spaces for V_h and S_h , including also high-order interpolation spaces.**
- **V_h made of piecewise linears and S_h made of piecewise constants.**

We will show the stability of eq. (63) by obtaining an inf-sup condition in the norm of eq. (62). Let us take $[v_h^I, \tau_h^I, v_h^{II}, \tau_h^{II}] = [u_h^I, -\sigma_h^I - \frac{\beta}{h} k \tilde{\tau}_h^I, u_h^{II}, -\sigma_h^{II} - \frac{\beta}{h} k \tilde{\tau}_h^{II}]$, where $\tilde{\tau}_h^I$ is the function in S_h^I which makes eqs. (64) and (65) hold for $u_h^I - u_h^I$, and $\tilde{\tau}_h^{II}$ is the function in S_h^{II} which makes eqs. (64) and (65) hold for $u_h^{II} - u_h^{II}$. β is a dimensionless constant to be defined. Using (64)-(65) and Young's inequality, we have:

$$\begin{aligned}
& B \left([u_h^I, \sigma_h^I, u_h^{II}, \sigma_h^{II}], [u_h^I, -\sigma_h^I - \frac{\beta}{h} k \tilde{\tau}_h^I, u_h^{II}, -\sigma_h^{II} - \frac{\beta}{h} k \tilde{\tau}_h^{II}] \right) = \\
& k \left(\nabla u_h^I, \nabla u_h^I \right)_{\Omega^I} - \frac{1}{2} \left(\sigma_h^I \cdot \mathbf{n}^I + \sigma_h^I \cdot \mathbf{n}^I, u_h^I \right)_{\Gamma} + \frac{1}{n} \left(\nabla u_h^I, \sigma_h^I \right)_{\Omega_{\Gamma_{in}}^I} - \frac{1}{n} k \left(\nabla u_h^I, \nabla u_h^I \right)_{\Omega_{\Gamma_{in}}^I} + \\
& + \frac{1}{nk} \left(\sigma_h^I, \sigma_h^I \right)_{\Omega_{\Gamma_{in}}^I} - \frac{1}{n} \left(\sigma_h^I, \nabla u_h^I \right)_{\Omega_{\Gamma_{in}}^I} + \frac{1}{2} \left(\sigma_h^I \cdot \mathbf{n}^I, u_h^I - u_h^I \right)_{\Gamma} + \\
& + \frac{\beta}{nh} \left(\tilde{\tau}_h^I, \sigma_h^I \right)_{\Omega_{\Gamma_{in}}^I} - \frac{\beta k}{nh} \left(\tilde{\tau}_h^I, \nabla u_h^I \right)_{\Omega_{\Gamma_{in}}^I} + \frac{\beta k}{2h} \left(\tilde{\tau}_h^I \cdot \mathbf{n}^I, u_h^I - u_h^I \right)_{\Gamma} + \\
& k \left(\nabla u_h^{II}, \nabla u_h^{II} \right)_{\Omega^{II}} - \frac{1}{2} \left(\sigma_h^{II} \cdot \mathbf{n}^{II} + \sigma_h^{II} \cdot \mathbf{n}^{II}, u_h^{II} \right)_{\Gamma} + \frac{1}{n} \left(\nabla u_h^{II}, \sigma_h^{II} \right)_{\Omega_{\Gamma_{in}}^{II}} - \frac{1}{n} k \left(\nabla u_h^{II}, \nabla u_h^{II} \right)_{\Omega_{\Gamma_{in}}^{II}} + \\
& + \frac{1}{nk} \left(\sigma_h^{II}, \sigma_h^{II} \right)_{\Omega_{\Gamma_{in}}^{II}} - \frac{1}{n} \left(\sigma_h^{II}, \nabla u_h^{II} \right)_{\Omega_{\Gamma_{in}}^{II}} + \frac{1}{2} \left(\sigma_h^{II} \cdot \mathbf{n}^{II}, u_h^{II} - u_h^I \right)_{\Gamma} + \\
& + \frac{\beta}{nh} \left(\tilde{\tau}_h^{II}, \sigma_h^{II} \right)_{\Omega_{\Gamma_{in}}^{II}} - \frac{\beta k}{nh} \left(\tilde{\tau}_h^{II}, \nabla u_h^{II} \right)_{\Omega_{\Gamma_{in}}^{II}} + \frac{\beta k}{2h} \left(\tilde{\tau}_h^{II} \cdot \mathbf{n}^{II}, u_h^{II} - u_h^I \right)_{\Gamma} \geq \\
& \left(1 - \frac{1}{n} \right) k \|\nabla u_h^I\|^2 + \frac{1}{nk} \|\sigma_h^I\|^2 + \\
& - \frac{\beta \delta_2^{1/2}}{nh^{1/2}} \|u_h^I - u_h^I\|_{L^2(\Gamma)} \|\sigma_h^I\| - \frac{\beta k \delta_2^{1/2}}{nh^{1/2}} \|u_h^I - u_h^I\|_{L^2(\Gamma)} \|\nabla u_h^I\| + \frac{\beta k \delta_1}{2h} \|u_h^I - u_h^I\|_{L^2(\Gamma)}^2 - \frac{\beta k \delta_0}{2} \|\nabla(u_h^I - u_h^I)\|^2 + \\
& \left(1 - \frac{1}{n} \right) k \|\nabla u_h^{II}\|^2 + \frac{1}{nk} \|\sigma_h^{II}\|^2 + \\
& - \frac{\beta \delta_2^{1/2}}{nh^{1/2}} \|u_h^{II} - u_h^I\|_{L^2(\Gamma)} \|\sigma_h^{II}\| - \frac{\beta k \delta_2^{1/2}}{nh^{1/2}} \|u_h^{II} - u_h^I\|_{L^2(\Gamma)} \|\nabla u_h^{II}\| + \frac{\beta k \delta_1}{2h} \|u_h^{II} - u_h^I\|_{L^2(\Gamma)}^2 - \frac{\beta k \delta_0}{2} \|\nabla(u_h^{II} - u_h^I)\|^2 \geq \\
& \left(1 - \frac{1}{n} \right) k \|\nabla u_h^I\|^2 + \frac{1}{nk} \|\sigma_h^I\|^2 + \left(1 - \frac{1}{n} \right) k \|\nabla u_h^{II}\|^2 + \frac{1}{nk} \|\sigma_h^{II}\|^2 + \frac{\beta k \delta_1}{h} \|u_h^I - u_h^I\|_{L^2(\Gamma)}^2 + \\
& - \frac{\delta_2^{1/2}}{2\gamma h^{1/2}} \frac{\beta \delta_2^{1/2}}{nh^{1/2}} \|u_h^I - u_h^I\|_{L^2(\Gamma)} - \frac{\gamma h^{1/2}}{2\delta_2^{1/2}} \frac{\beta \delta_2^{1/2}}{nh^{1/2}} \|\sigma_h^I\| - \frac{\delta_2^{1/2}}{2\gamma h^{1/2}} \frac{\beta \delta_2^{1/2}}{nh^{1/2}} \|u_h^I - u_h^I\|_{L^2(\Gamma)} - \frac{\gamma h^{1/2}}{2\delta_2^{1/2}} \frac{\beta \delta_2^{1/2}}{nh^{1/2}} \|\nabla u_h^I\| + \\
& - \frac{\delta_2^{1/2}}{2\gamma h^{1/2}} \frac{\beta \delta_2^{1/2}}{nh^{1/2}} \|u_h^{II} - u_h^I\|_{L^2(\Gamma)} - \frac{\gamma h^{1/2}}{2\delta_2^{1/2}} \frac{\beta \delta_2^{1/2}}{nh^{1/2}} \|\sigma_h^{II}\| - \frac{\delta_2^{1/2}}{2\gamma h^{1/2}} \frac{\beta \delta_2^{1/2}}{nh^{1/2}} \|u_h^{II} - u_h^I\|_{L^2(\Gamma)} - \frac{\gamma h^{1/2}}{2\delta_2^{1/2}} \frac{\beta \delta_2^{1/2}}{nh^{1/2}} \|\nabla u_h^{II}\| + \\
& - \beta k \delta_0 \|\nabla u_h^I\|^2 - \beta k \delta_0 \|\nabla u_h^{II}\|^2 \geq \\
& \left(1 - \frac{1}{n} - \beta \left(\frac{\gamma}{2n} + \delta_0 \right) \right) k \|\nabla u_h^I\|^2 + \frac{1}{n} \left(1 - \frac{\beta \gamma}{2} \right) \frac{1}{k} \|\sigma_h^I\|^2 + \\
& \left(1 - \frac{1}{n} - \beta \left(\frac{\gamma}{2n} + \delta_0 \right) \right) k \|\nabla u_h^{II}\|^2 + \frac{1}{n} \left(1 - \frac{\beta \gamma}{2} \right) \frac{1}{k} \|\sigma_h^{II}\|^2 + \\
& \beta \left(\delta_1 - \frac{2\delta_2}{\gamma n} \right) \frac{k}{h} \|u_h^I - u_h^I\|_{L^2(\Gamma)}
\end{aligned}$$

where γ is an arbitrary dimensionless constant. We now take

$$n > 1, \gamma > \frac{2\delta_2}{\delta_1 n}, \beta < \min \left(\frac{1 - \frac{1}{n}}{\left(\frac{\gamma}{2n} + \delta_0 \right)}, \frac{2}{\gamma} \right), \quad (66)$$

and we take into account the following inequality, where again (64)-(65) and Young's inequality have been used:

$$\begin{aligned}
& \left\| \left[u_h^I, -\sigma_h^I - \frac{\beta}{h} k \tilde{\tau}_h^I, u_h^{II}, -\sigma_h^{II} - \frac{\beta}{h} k \tilde{\tau}_h^{II} \right] \right\|^2 = \\
& k \|\nabla u_h^I\|_{L^2(\Omega^I)}^2 + k \|\nabla u_h^{II}\|_{L^2(\Omega^{II})}^2 + \frac{k}{h} \|u_h^I - u_h^{II}\|_{L^2(\Gamma)}^2 + \frac{1}{k} \|\sigma_h^I - \frac{\beta}{h} k \tilde{\tau}_h^I\|^2 + \frac{1}{k} \|\sigma_h^{II} - \frac{\beta}{h} k \tilde{\tau}_h^{II}\|^2 \leq \\
& k \|\nabla u_h^I\|_{L^2(\Omega^I)}^2 + k \|\nabla u_h^{II}\|_{L^2(\Omega^{II})}^2 + \frac{k}{h} \|u_h^I - u_h^{II}\|_{L^2(\Gamma)}^2 + \frac{2}{k} \|\sigma_h^I\|^2 + \frac{2k\beta^2}{h^2} \|\tilde{\tau}_h^I\|^2 + \frac{2}{k} \|\sigma_h^{II}\|^2 + \frac{2k\beta^2}{h^2} \|\tilde{\tau}_h^{II}\|^2 \leq \\
& k \|\nabla u_h^I\|_{L^2(\Omega^I)}^2 + k \|\nabla u_h^{II}\|_{L^2(\Omega^{II})}^2 + \frac{k}{h} (1 + 4\beta^2 \delta_2) \|u_h^I - u_h^{II}\|_{L^2(\Gamma)}^2 + \frac{2}{k} \|\sigma_h^I\|^2 + \frac{2}{k} \|\sigma_h^{II}\|^2 \leq \\
& \max(1 + 4\beta^2 \delta_2, 2) \left\| \left[u_h^I, \sigma_h^I, u_h^{II}, \sigma_h^{II} \right] \right\|^2
\end{aligned}$$

This gives us the following theorem::

Theorem. Suppose that eqs. (64) and (65) hold and $n > 1$. Then, the bilinear form eq. (63) satisfies that for all $[u_h^I, \sigma_h^I, u_h^{II}, \sigma_h^{II}]$ there exist $[v_h^I, \tau_h^I, v_h^{II}, \tau_h^{II}]$ and $\alpha > 0$ such that

$$B([u_h^I, \sigma_h^I, u_h^{II}, \sigma_h^{II}], [v_h^I, \tau_h^I, v_h^{II}, \tau_h^{II}]) \geq \alpha ||| [u_h^I, \sigma_h^I, u_h^{II}, \sigma_h^{II}] ||| ||| [v_h^I, \tau_h^I, v_h^{II}, \tau_h^{II}] |||$$

This theorem establishes the stability of the method for all interpolation spaces fulfilling (eq. (64))-(eq. (65)).

4.6. Examples

In this section we present various examples, where the performance of the new method is compared to Nitsche's method in the framework of both p - and B-spline version of the FCM.

4.6.1. Constant coefficients, non-conform discretizations in h and p . This introductory example is to provide a first insight into the performance of the weak coupling method given by eqs. (59a) and (59b). The starting point is the 2D Poisson's problem with a smooth solution already presented in section 3.3. This time, however, the problem is split into two domains Ω^I and Ω^{II} which are weakly coupled. The continuous situation is depicted in fig. 10a. We choose to discretize Ω^I with

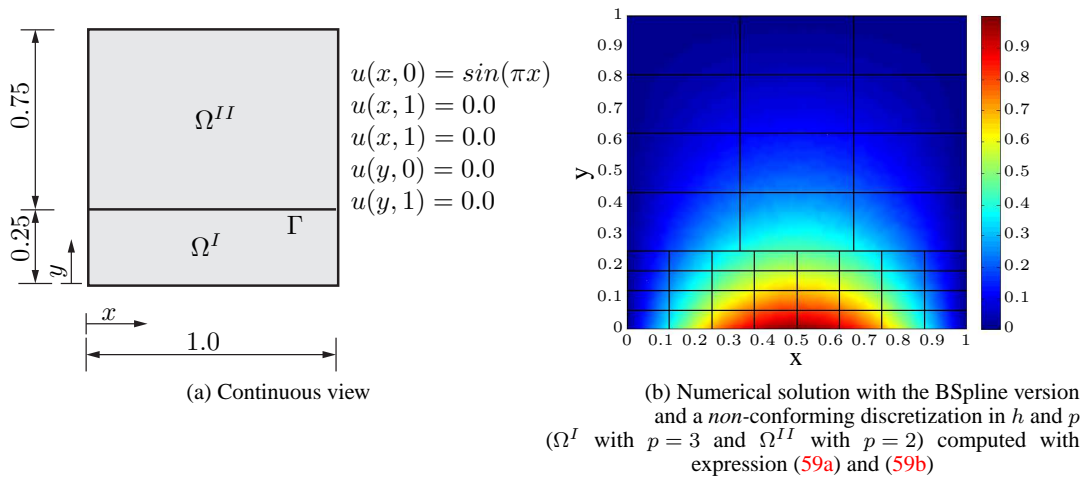


Figure 10. 2D Poisson's problem as in fig. 3 but consisting of two domains Ω^I and Ω^{II} , weakly coupled at Γ .

8×4 elements while Ω^{II} is discretized by 3×4 elements. The discretization is depicted in fig. 10b among with a snapshot of the solution. The *non-conformity* of the discretization on Γ manifests itself not only in the appearance of hanging nodes. Additionally, we discretize Ω^I with p and Ω^{II} with $p - 1$. The discretizations are, thus, *non-conforming* in h and p across the interface. However, in a first step, the domains do not overlap, i.e., they are coupled along Γ . The coupled solution depicted in fig. 10b does not show any (visible) kinks or jumps across Γ . This demonstrates that eq. (40) is satisfied.

The corresponding fluxes depicted for both directions x and y are depicted in fig. 11. They are smooth across Γ which demonstrates that eq. (41) is also satisfied.

Figure 12a depicts the convergence in the energy norm as we increase p from 2 to 9 for Ω^I and from 1 to 8 for Ω^{II} . We consider the p - and B-spline version of FCM, using both standard Nitsche and the method proposed here for patch coupling. The rate of convergence is exponential until numerical accuracy is reached and the curves for all methods are practically congruent to one another. The condition numbers are depicted in fig. 12b. They remain quite low for the p -version and are once again higher for the B-Spline version due to the sub-optimality of the BSpline basis

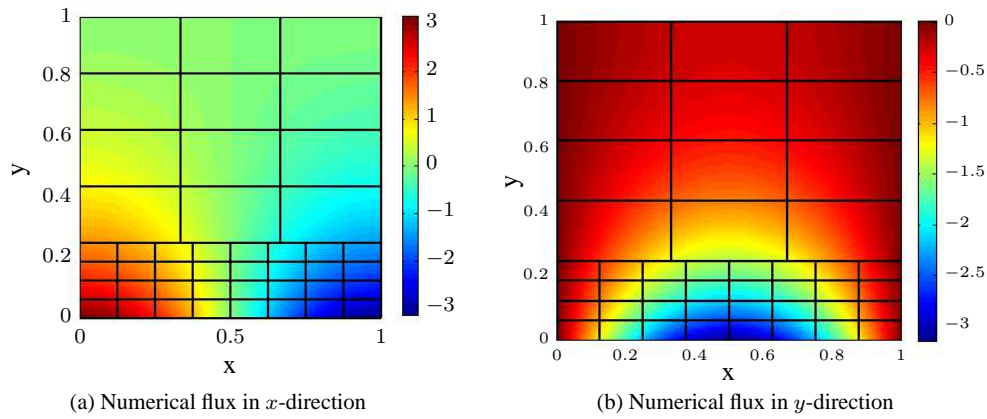
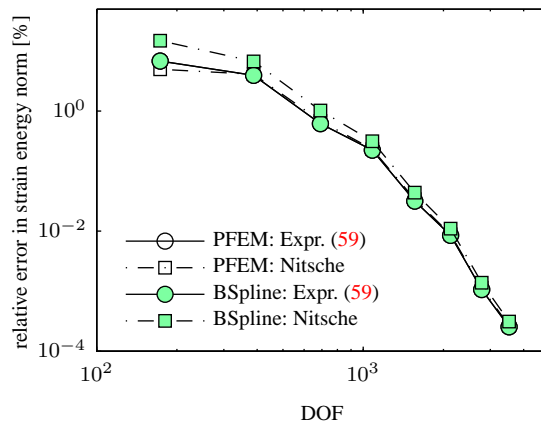
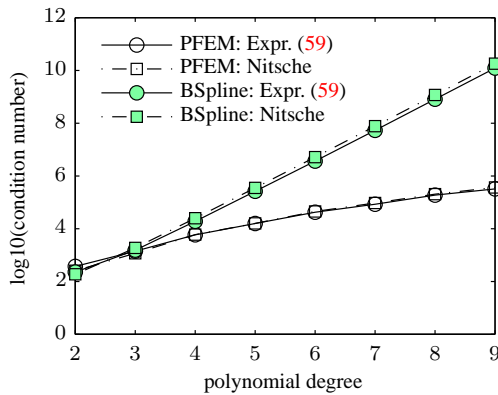


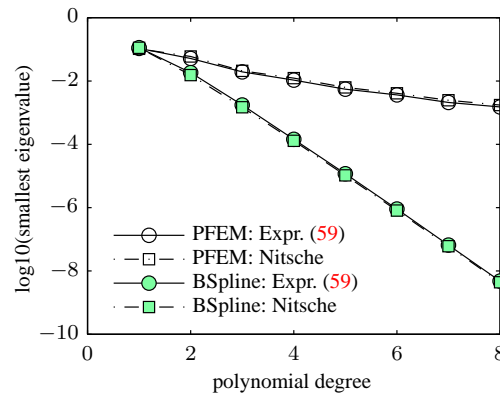
Figure 11. Flux corresponding to fig. 10b.



(a) convergence behaviour



(b) condition number of system stiffness matrix



(c) smallest eigenvalue

Figure 12. analysis corresponding to fig. 10b

with respect to the condition numbers. The plot of the smallest eigenvalue of the stiffness matrix fig. 12c again shows that all discrete eigenvalues remain positive.

Additionally, we provide a convergence study concerning an h -extension in Figure 13.

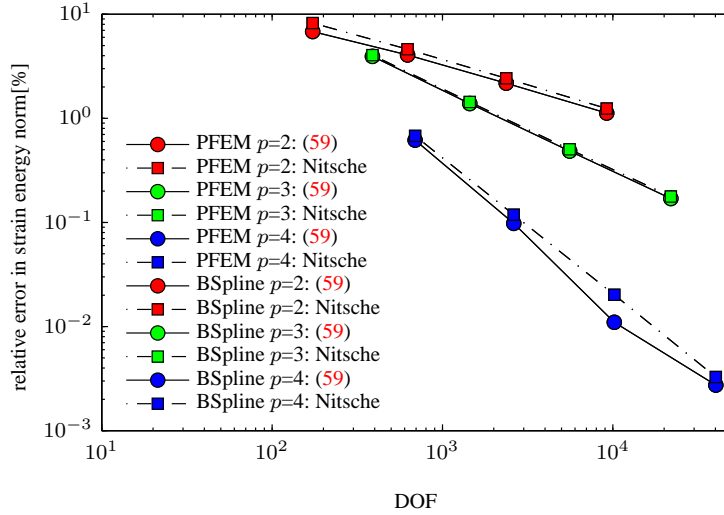


Figure 13. h -refinement for different p . Each marker symbolizes the result for 8×4 , 16×8 , 32×16 , and 64×32 elements, of degree $p = 2 \dots 4$ for the lower patch, respectively. The upper patch is discretized with 3×4 , 6×8 , 12×16 , and 24×32 elements of degree $p = 1 \dots 3$, respectively.

4.6.2. *Constant coefficients, overlapping discretizations.* We now change the discretization of the example given in section 4.6.1, (fig. 10a) such that the domains Ω^I and Ω^{II} are overlapping in the region $(0 \leq x < 1.0) \wedge (0.45 \leq y < 0.55)$. We weakly enforce the continuity (eqs. (40) and (41)) at the boundary Γ and explicitly note that Γ is now inclined and internal to the discretization of both domains. The continuous situation is depicted in fig. 14a. The solution is depicted in fig. 14b along

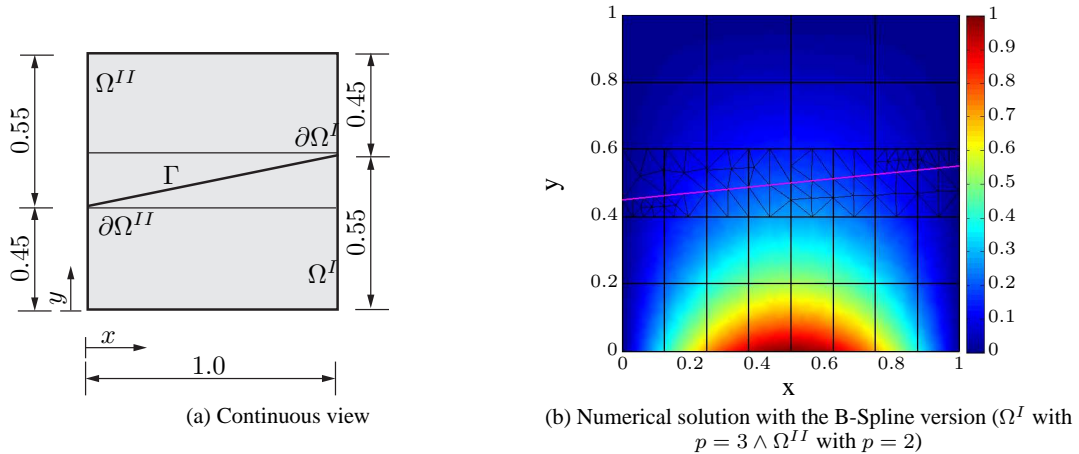


Figure 14. Constant coefficients, overlapping discretizations: Both computational domains Ω^I and Ω^{II} have an extension of $x \times y = 1.0 \times 0.55$. They overlap on $(0 \leq x < 1.0) \wedge (0.45 \leq y < 0.55)$.

with its Cartesian discretization and the triangular integration mesh. The integration is carried out on $\Omega_{\Gamma_{in}}$ only. Note that $\Omega_{\Gamma_{in}} \in \Omega^I$ lies below Γ and $\Omega_{\Gamma_{in}} \in \Omega^{II}$ lies above Γ . The obtained numerical solution is smooth, already at low polynomial orders. This is particularly eminent on the plot of the flux in y -direction in fig. 15a. The plot of the fluxes in x -direction is omitted as it provides no further insight. Again, we note an exponential convergence in fig. 15b before it levels off at a low strain

energy due to the bad conditioning depicted in fig. 15c. The corresponding lowest eigenvalues are depicted in fig. 15d.

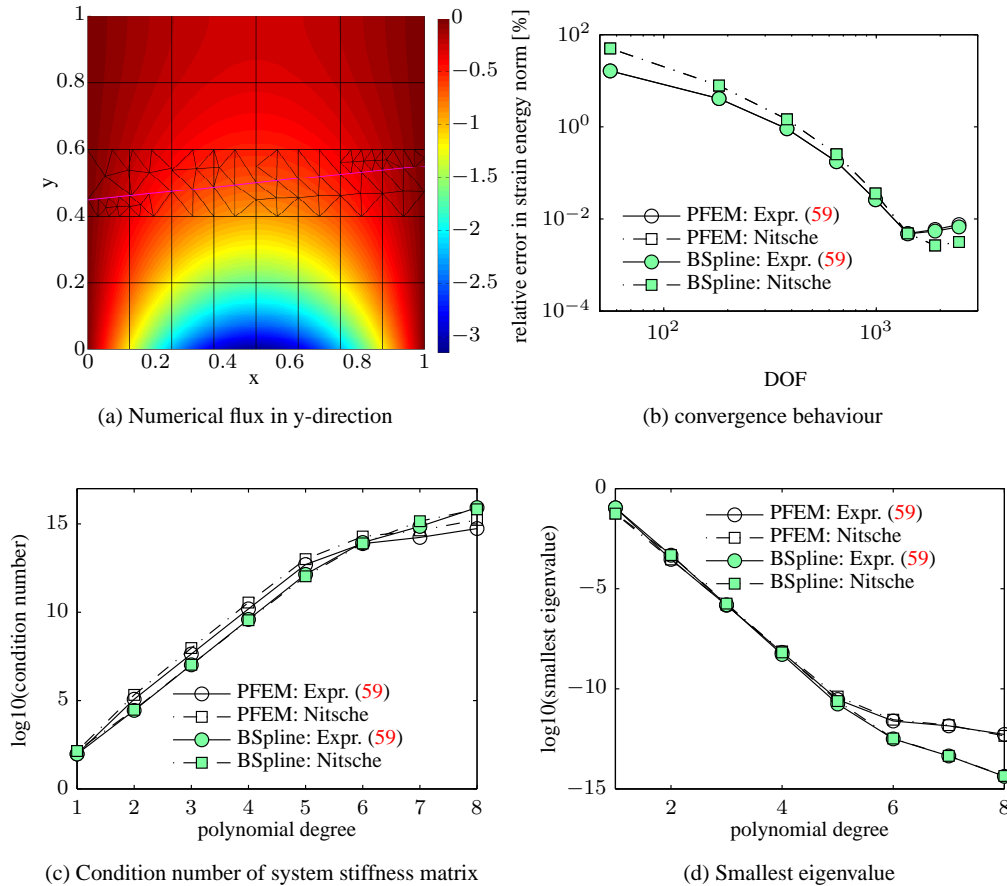


Figure 15. Analysis corresponding to fig. 14a.

4.6.3. Bimetal strip: discontinuous coefficients, non-conforming discretizations in h and p . This example serves to show that the proposed method has no difficulties in representing discontinuities in the derivatives of the solution across material interfaces which arise if the material coefficients jump as well. The first example with jumping coefficients is a bimetal strip on which the Laplace problem is solved. Its solution is depicted in fig. 16a. Dirichlet boundary conditions are applied at the lower end $u(x, 0) = 0$ and on the upper end $u(x, 1) = 1$. We split the domain into a lower part Ω^I in which the material coefficient is $k^I = 1 \forall x, y \in \Omega^I$ and an upper part Ω^{II} in which $k^{II} = 2 \forall x, y \in \Omega^{II}$. The interface between the two domains $\Gamma : y = 0.5$ is a straight line, parallel to the x -axis. We discretize Ω^I with 5×4 elements with $p = 1$ organized in one patch covering all of Ω^I and likewise Ω^{II} with 3×3 elements with $p = 1$. This results in hanging nodes at Γ .

We expect a linear solution in y and a constant solution in x with a kink at the interface Γ . Since the solution is already represented exactly at $p = 1$, convergence-studies are not carried out. Figure 16a depicts the numerical solution along with the discretization. The fluxes are expected to exhibit a jump at Γ and should remain constant within the domains. Figure 16b depicts a view of the fluxes in y -direction along with the chosen discretization. The exact solution is matched up to machine-precision, as expected.

4.6.4. Inclusion problem: discontinuous coefficients and embedded domain. The last example aims at demonstrating the performance of the method given by expression (59a) and (59b) in

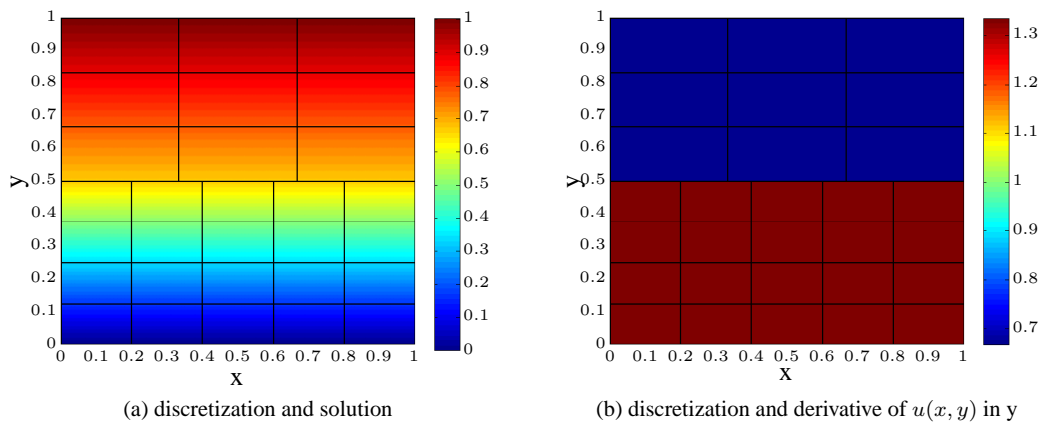


Figure 16. Bimetal strip: solution $u(x, y)$ and derivative of $u(x, y)$ in y along with discretization

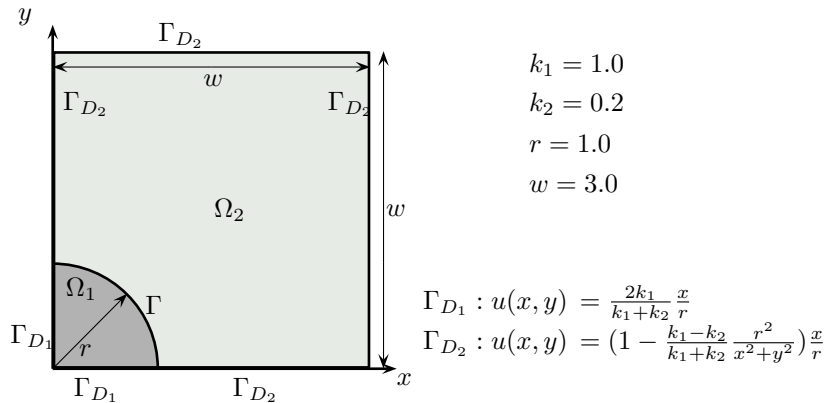


Figure 17. Example: circular inclusion

an embedded domain situation. The problem consists of an infinite medium, where the material coefficient is $k^I = 1$, with a circular inclusion, where the material coefficient is $k^{II} = 0.2$. A constant flux is applied along the x -direction at large distances from the inclusion. The analytical solution is given by

$$u(x, y) = \begin{cases} \frac{2k_1}{k_1+k_2} \frac{x}{r} & \text{if } x^2 + y^2 < r^2 \\ \left(1 - \frac{k_1-k_2}{k_1+k_2} \frac{r^2}{x^2+y^2}\right) \frac{x}{r} & \text{if } x^2 + y^2 \geq r^2 \end{cases} \quad (67)$$

where r denotes the radius of the inclusion. Due to its symmetry, only a quarter of the domain is modeled and the Dirichlet boundary conditions are applied to the boundaries Γ_{D_1} and Γ_{D_2} according to the analytical solution. The continuous view of the example is depicted in fig. 17. The results are depicted in figs. 18a to 18f. The performance of the new method carries over from the rest of the examples provided in this section. This example shows how well discontinuities in the derivatives are approximated, also in an embedded domain situation (see figs. 18b and 18c). To clear the view, we omit depicting the integration mesh and only show the Cartesian discretization on which the shape functions are spanned.

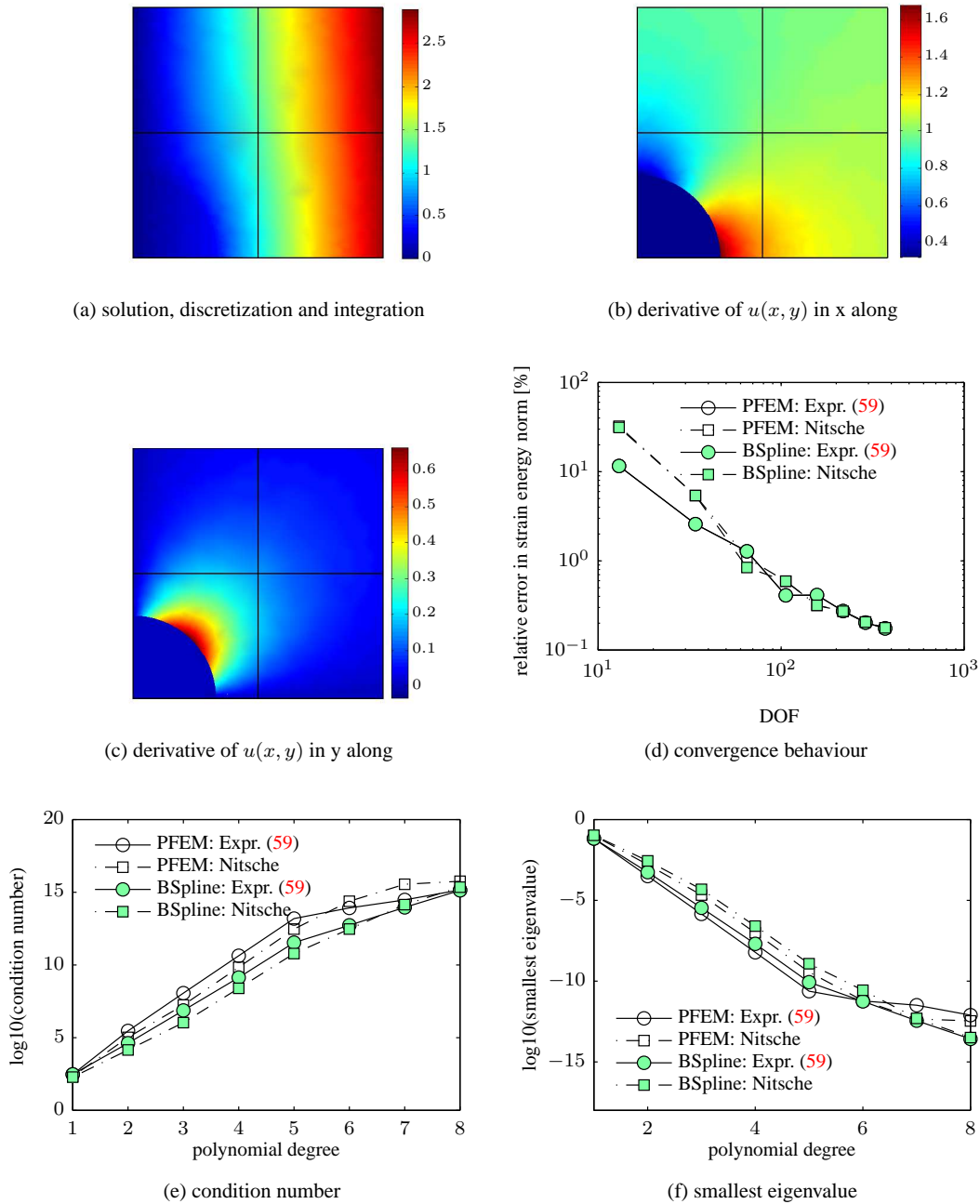


Figure 18. Circular inclusion, results.

5. SUMMARY, CONCLUSIONS AND SUGGESTIONS FOR FURTHER RESEARCH

Starting point of this paper was the formulation given in [33] for the application of boundary conditions which must not necessarily conform to the finite element mesh. It is attractive in the sense that there is no need to compute a stabilization parameter. We demonstrated, that this method is able to provide results of similar accuracy as the more classical Nitsche’s formulation which calls for the computation of stabilization parameters.

We have then used the fundamental ideas provided in [33] for the derivation of a new formulation for the weak coupling of domains. It is, as well, free from problem-dependent stabilization parameters which need estimation. The new, coupled formulation works well for discretizations of high order, too. It is not specific to the underlying discretization and preserves symmetry. It is thus well suited for the weak coupling of trimmed B-Spline patches and high-order embedded domain situations. We are able to prove stability and clearly point out the difference to Nitsche's method. While this are very nice features, we see some room for improvement as well.

The formulation carries the material matrix in the Nitsche terms as well as in the automatic stabilization terms. This poses no extra effort as long as geometric nonlinearities are not considered. In this case one would need the directional derivative of practically all terms involved in the entire formulation. If one sets out to perform these analytically, this will result in a complicated formulation. Automatic differentiation would probably be the best way to go in this case, especially if one intends to avoid redoing the derivatives for each material law employed. The same is true, of course, for Nitsche's method but we would like to point out that the new formulation does not alleviate this burden.

Recently, an interesting alternative has been proposed in [71], for which, however stability could not be proven. Nevertheless, the formulation presented therein seems to be stable and perform well, also in geometric nonlinear cases. It would be interesting to extend it to high-order embedded domain approaches.

The most important advantage of the new method presented in this article compared to the classical Nitsche's method is the avoidance of the solution of auxiliary eigenvalue problems for the estimation of the stability parameters. This is replaced by computing the inverse of a matrix similar to a mass matrix defined on the remaining part of the cut element. However, it is to be remarked that both problems are formulated on the cut elements and this leads to badly conditioned problems. Throughout the computations we noticed a very close correspondence between the results of Nitsche's method and the new method in the computed examples. It would be interesting to investigate if there is a direct mathematical connection. This insight might also help to overcome the conditioning problem on cut elements in a similar fashion as presented in [72], where yet another additional penalty term is added which acts on the jumps of the gradients over element faces in the interface zone to ensure that the conditioning of the matrix is independent of how the boundary cuts the mesh.

REFERENCES

1. Hughes TJR, Cottrell JA, Bazilevs Y. Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement. *Computer Methods in Applied Mechanics and Engineering* 2005; **194**:4135–4195.
2. Cottrell JA, Hughes TJR, Bazilevs Y. *Isogeometric analysis: Towards Indegration of CAD and FEM*. John Wiley & Sons, 2009.
3. Parvizian J, Düster A, Rank E. Finite cell method – h- and p-extension for embedded domain problems in solid mechanics. *Computational Mechanics* 2007; **41**:121–133.
4. Düster A, Parvizian J, Yang Z, Rank E. The finite cell method for three-dimensional problems of solid mechanics. *Computer Methods in Applied Mechanics and Engineering* 2008; **197**:3768–3782.
5. Rank E, Ruess M, Kollmannsberger S, Schillinger D, Düster A. Geometric modeling, isogeometric analysis and the finite cell method. *Computer Methods in Applied Mechanics and Engineering* 2012; **249-252**:104–115, doi: <http://dx.doi.org/10.1016/j.cma.2012.05.022>.
6. Hansbo P. Nitsche's method for interface problems in computational mechanics. *GAMM-Mitteilungen* 2005; **28**(2):183206.
7. Lew AJ, Buscaglia GC. A discontinuous-Galerkin-based immersed boundary method. *International Journal for Numerical Methods in Engineering* 2008; **76**(4):427–454, doi:10.1002/nme.2312. URL <http://doi.wiley.com/10.1002/nme.2312>.
8. Höllig K. *Finite Element Methods with B-Splines*. Frontiers in Applied Mathematics, SIAM Society for Industrial and Applied Mathematics, 2003.
9. Sanches R, Bornemann P, Cirak F. Immersed b-spline (i-spline) finite element method for geometrically complex domains. *Computer Methods in Applied Mechanics and Engineering* 2011; **200**(13-16):14321445, doi:10.1016/j.cma.2010.12.008. URL <http://linkinghub.elsevier.com/retrieve/pii/S0045782510003531>.
10. Rüberg T, Cirak F. Subdivision-stabilised immersed b-spline finite elements for moving boundary flows. *Computer Methods in Applied Mechanics and Engineering* 2012; **209-212**:266–283, doi:10.1016/j.cma.2011.10.007. URL <http://linkinghub.elsevier.com/retrieve/pii/S0045782511003197>.

11. Arroyo M, Ortiz M. Local maximum-entropy approximation schemes: a seamless bridge between finite elements and meshfree methods. *International Journal for Numerical Methods in Engineering* 2006; **65**(13):2167–2202, doi:10.1002/nme.1534. URL <http://doi.wiley.com/10.1002/nme.1534>.
12. Legrain G, Chevaugnon N, Drau K. High order x-FEM and levelsets for complex microstructures: Uncoupling geometry and approximation. *Computer Methods in Applied Mechanics and Engineering* 2012; **241-244**:172189, doi:10.1016/j.cma.2012.06.001. URL <http://linkinghub.elsevier.com/retrieve/pii/S0045782512001880>.
13. Duarte CA, Reno LG, Simone A. A high-order generalized FEM for through-the-thickness branched cracks. *International Journal for Numerical Methods in Engineering* 2007; **72**(3):325–351, doi:10.1002/nme.2012. URL <http://doi.wiley.com/10.1002/nme.2012>.
14. Joulaian M, Düster A. Local enrichment of the finite cell method for problems with material interfaces. *Computational Mechanics* 2013; doi:10.1007/s00466-013-0853-8. URL <http://link.springer.com/10.1007/s00466-013-0853-8>.
15. Babuška. The finite element method with penalty. *Mathematics of Computation* April 1973; **27**(122):221.
16. Zhu T, Atluri SN. A modified collocation method and a penalty formulation for enforcing the essential boundary conditions in the element free galerkin method. *Computational Mechanics* April 1998; **21**(3):211–222.
17. Babuška I. The Finite Element Method with Lagrangian Multipliers. *Numerische Mathematik* 1973; **20**:179–192.
18. Brezzi F. On the existence, uniqueness and approximation of saddle-point problems arising from lagrangian multipliers. *ESAIM: Mathematical Modelling and Numerical Analysis - Modlisation Mathematique et Analyse Numrique* 1974; **8**(R2):129–151.
19. Barbosa HJC, Hughes TJR. Boundary Lagrange multipliers in finite element methods: Error analysis in natural norms. *Numerische Mathematik* 1992; **62**(1):1–15.
20. Glowinski R, Pan TW, Periaux J. A fictitious domain method for dirichlet problem and applications. *Computer Methods in Applied Mechanics and Engineering* 1994; **111**(34):283–303.
21. Hansbo P, Lovadina C, Perugia I, Sangalli G. A Lagrange multiplier method for the finite element solution of elliptic interface problems using non-matching meshes. *Numerische Mathematik* 2005; **100**(1):91–115, doi: 10.1007/s00211-005-0587-4. URL <http://link.springer.com/10.1007/s00211-005-0587-4>.
22. Mourad HM, Dolbow J, Harari I. A bubble-stabilized finite element method for dirichlet constraints on embedded interfaces. *International Journal for Numerical Methods in Engineering* 2007; **69**(4):772–793, doi:10.1002/nme.1788. URL <http://doi.wiley.com/10.1002/nme.1788>.
23. Burman E, Hansbo P. Fictitious domain finite element methods using cut elements: I. a stabilized Lagrange multiplier method. *Computer Methods in Applied Mechanics and Engineering* 2010; **199**(4144):2680–2686.
24. Nitsche J. Über ein Variationsprinzip zur Lösung von Dirichlet-Problemen bei Verwendung von Teilräumen, die keinen Randbedingungen unterworfen sind. *Abhandlungen aus dem Mathematischen Seminar der Universität Hamburg* 1971; **36**(1):9–15.
25. Stenberg R. On some techniques for approximating boundary conditions in the finite element method. *Journal of Computational and Applied Mathematics* 1995; **63**(1-3):139148, doi:10.1016/0377-0427(95)00057-7.
26. Griebel M, Schweitzer M. A Particle-Partition of Unity Method Part v: Boundary Conditions. *Geometric Analysis and Nonlinear Partial Differential Equations*, Hildebrandt S, Karcher H (eds.). Springer Berlin Heidelberg, 2003; 519–542, doi:10.1007/978-3-642-55627-2.27. URL http://dx.doi.org/10.1007/978-3-642-55627-2_27.
27. Harari I, Shavelzon E. Embedded kinematic boundary conditions for thin plate bending by Nitsche's approach. *International Journal for Numerical Methods in Engineering* 2012; **92**(1):99–114, doi:10.1002/nme.4337. URL <http://doi.wiley.com/10.1002/nme.4337>.
28. Ruess M, Schillinger D, Bazilevs Y, Varduhn V, Rank E. Weakly enforced essential boundary conditions for NURBS-embedded and trimmed NURBS geometries on the basis of the finite cell method. *International Journal for Numerical Methods in Engineering* 2013; **95**(10):811–846, doi:10.1002/nme.4522. URL <http://doi.wiley.com/10.1002/nme.4522>.
29. Zander N, Kollmannsberger S, Ruess M, Yosibash Z, Rank E. The Finite Cell Method for Linear Thermoelasticity. *Computers & Mathematics with Applications* 2012; **64**(11):3527–3541, doi:10.1016/j.camwa.2012.09.002. URL <http://linkinghub.elsevier.com/retrieve/pii/S0898122112005688>.
30. Bazilevs Y, Hughes T. Weak imposition of Dirichlet boundary conditions in fluid mechanics. *Computers & Fluids* 2007; **36**(1):12–26, doi:10.1016/j.compfluid.2005.07.012.
31. Codina R, Baiges J. Approximate imposition of boundary conditions in immersed boundary methods. *International Journal for Numerical Methods in Engineering* 2009; **80**:13791405, doi:10.1002/nme.2662.
32. Gerstenberger A, Wall W. An embedded Dirichlet formulation for 3D continua. *International Journal for Numerical Methods in Engineering* 2010; **82**:537–563.
33. Baiges J, Codina R, Henke F, Shahmiri S, Wall WA. A symmetric method for weakly imposing Dirichlet boundary conditions in embedded finite element meshes. *International Journal for Numerical Methods in Engineering* 2012; **90**(5):636–658, doi:10.1002/nme.3339. URL <http://dx.doi.org/10.1002/nme.3339>.
34. Bernardi C, Maday Y, Patera A. A new non conforming approach to domain decomposition: The mortar element method. *Collège de France Seminar*, Brezis H, Lions JL (eds.). Pitman, 1994; 53–63.
35. Flemisch B, Puso M, Wohlmuth B. A new dual mortar method for curved interfaces:2d linear elasticity. *International Journal for Numerical Methods in Engineering* 2000; **63**:813–832.
36. Wohlmuth B. A mortar finite element method using dual spaces for the Lagrange multiplier. *SIAM Journal on Numerical Analysis* 2000; **38**:989–1012.
37. Puso MA. A 3d mortar method for solid mechanics. *International Journal for Numerical Methods in Engineering* 2004; **59**:315–336.
38. Lamichhane B, Stevenson R, Wohlmuth B. Higher order mortar finite element methods in 3d with dual Lagrange multiplier bases. *Numerische Mathematik* 2005; **102**:93–121.

39. Hansbo A, Hansbo P. An unfitted finite element method, based on Nitsche's method, for elliptic interface problems. *Computer Methods in Applied Mechanics and Engineering* 2002; **191**(47-48):5537 – 5552, doi:DOI:10.1016/S0045-7825(02)00524-8. URL <http://www.sciencedirect.com/science/article/pii/S0045782502005248>.
40. Becker R, Hansbo P, Stenberg R. A finite element method for domain decomposition with non-matching grids. *ESAIM: Mathematical Modelling and Numerical Analysis* November 2003; **37**(2):209–225.
41. Hansbo A, Hansbo P, Larson MG. A finite element method on composite grids based on Nitsche's method. *ESAIM: Mathematical Modelling and Numerical Analysis* 2003; **37**(03):495–514.
42. Hansbo A, Hansbo P. A finite element method for the simulation of strong and weak discontinuities in solid mechanics. *Computer Methods in Applied Mechanics and Engineering* 2004; **193**(33-35):3523–3540.
43. Sanders JD, Laursen T, Puso M. A Nitsche embedded mesh method. *Computational Mechanics* September 2011; **49**(2):243257.
44. Ruess M, Schillinger D, Özcan A, Rank E. Weak coupling for isogeometric analysis of non-matching and trimmed multi-patch geometries. *Computer Methods in Applied Mechanics and Engineering* 2014; :46–71.
45. Wriggers P, Zavarise G. A formulation for frictionless contact problems using a weak form introduced by Nitsche. *Computational Mechanics* Februar 2008; **41**(3):407–420.
46. Hansbo P, Hermansson J. Nitsche's method for coupling non-matching meshes in fluid-structure vibration problems. *Computational Mechanics* September 2003; **32**(1-2):134–139.
47. Szabó B, Babuška I. *Finite element analysis*. John Wiley & Sons, 1991.
48. Parvizian J, Düster A, Rank E. Topology optimization using the finite cell method. *Optimization and Engineering* 2011; :1–22URL <http://dx.doi.org/10.1007/s11081-011-9159-x>, 10.1007/s11081-011-9159-x.
49. Schillinger D, Ruess M, Zander N, Bazilevs Y, Düster A, Rank E. Small and large deformation analysis with the p - and B -spline versions of the finite cell method. *Computational Mechanics* 2012; **50**:445–478, doi:DOI: 10.1007/s00466-012-0684-z.
50. Schillinger D, Düster A, Rank E. The hp - d -adaptive finite cell method for geometrically nonlinear problems of solid mechanics. *International Journal for Numerical Methods in Engineering* 2012; **89**:1171–1202, doi: DOI10.1002/nme.3289.
51. Schillinger D, Dedè L, Scott MA, Evans JA, Borden MJ, Rank E, Hughes TJ. An isogeometric design-through-analysis methodology based on adaptive hierarchical refinement of NURBS, immersed boundary methods, and T-spline CAD surfaces. *Computer Methods in Applied Mechanics and Engineering* 2012; **249–252**:116–150, doi: 10.1016/j.cma.2012.03.017.
52. Schillinger D. The p - and b -spline versions of the geometrically nonlinear finite cell method and hierarchical refinement strategies for adaptive isogeometric and embedded domain analysis. Dissertation, Technische Universität München 2012.
53. Yang Z, Kollmannsberger S, Düster A, Ruess M, Burgkart R, Garcia E, Rank E. Computational Steering for Orthopaedics. *Computing and Visualization in Science* 2012; **14**:207–216.
54. Yang Z, Ruess M, Kollmannsberger S, Düster A, Rank E. An efficient integration technique for the voxel-based Finite Cell Method. *International Journal for Numerical Methods in Engineering* 2012; **91**:457–471.
55. Ruess M, Tal D, Trabelsi N, Yosibash Z, Rank E. The finite cell method for bone simulations: verification and validation. *Biomechanics and modeling in mechanobiology* 2012; **11**:425–437.
56. Düster A, Sehlhorst HG, Rank E. Numerical homogenization of heterogeneous and cellular materials utilizing the finite cell method. *Computational Mechanics* 2012; **50**:413–431, doi:doi:10.1007/s00466-012-0681-2.
57. Abedian A, Parvizian J, Düster A, Rank E. The finite cell method for the j_2 flow theory of plasticity. *Finite Elements in Analysis and Design* 2013; **69**:37–47, doi:10.1016/j.finel.2013.01.006.
58. Duczek S, Joulaian M, Düster A, Gabbert U. Modeling of ultrasonic guided waves using the finite and spectral cell method. *International Journal for Numerical Methods in Engineering* 2014, accepted; .
59. Cai Q, Kollmannsberger S, Mundani R, Rank E. The finite cell method for spatially varying dispersions in coupled multispecies reactive transport problems. *Proceedings of the International Conference on Computational Methods for Coupled Problems in Science and Engineering*, Kos Island, Greece, 2011.
60. Cai Q, Kollmannsberger S, Mundani R, Rank E. The finite cell method for solute transport problems in porous media. *Proceedings of the International Conference on Finite Elements in Flow Problems*, Garching, Germany, 2011.
61. Rank E, Kollmannsberger S, Sorger C, Düster A. Shell Finite Cell Method: A High Order Fictitious Domain Approach for Thin-Walled Structures. *Computer Methods in Applied Mechanics and Engineering* 2011; **200**:3200–3209, doi:DOI:10.1016/j.cma.2011.06.005. URL <http://www.sciencedirect.com/science/article/pii/S0045782511002234>.
62. Zander N, Bog S, Elhaddad M, Espinoza R, Hu H, Joly A, Wu C, Zerbe P, Düster A, Kollmannsberger S, et al. FCMLab: A Finite Cell Research Toolbox for MATLAB. *Advances in Engineering Software* 2014; doi: 10.1016/j.advengsoft.2014.04.004.
63. Gamma E, Helm R, Johnson R. *Design patterns: Elements of reusable object-oriented software*. Addison-Wesley, 1995.
64. Schillinger D, Rank E. An unfitted hp adaptive finite element method based on hierarchical b -splines for interface problems of complex geometry. *Computer Methods in Applied Mechanics and Engineering* 2011; **200**:3358–3380.
65. Schillinger D, Düster A, Rank E. The hp - d -adaptive finite cell method for geometrically nonlinear problems of solid mechanics. *International Journal for Numerical Methods in Engineering* 2012; **89**:1171–1202, doi: DOI10.1002/nme.3289.
66. Schillinger D, Kollmannsberger S, Mundani R, Rank E. The Finite Cell Method for Geometrically Nonlinear Problems of Solid Mechanics. *The World Congress on Computational Mechanics*, Sidney, Australia, 2010, doi: 10.1088/1757-899X/10/1/012170.
67. Abedian A, Parvizian J, Düster A, Khademyzadeh H, Rank E. Performance of different integration schemes in facing discontinuities in the Finite Cell Method. *International Journal of Computational Methods* 2013;

- 10(03):1350002–1–1350002–24.
68. Gordon W, Hall C. Transfinite element methods: Blending function interpolation over arbitrary curved element domains. *Numerische Mathematik* 1973; **21**:109–129.
 69. Embar A, Dolbow J, Harari I. Imposing Dirichlet boundary conditions with Nitsche’s method and spline-based finite elements. *International Journal for Numerical Methods in Engineering* 2010; **83**(7):877–898.
 70. Fernández-Méndez S, Huerta A. Imposing essential boundary conditions in mesh-free methods. *Computer Methods in Applied Mechanics and Engineering* 2004; **193**(12-14):1257 – 1275.
 71. Sanders J, Puso MA. An embedded mesh method for treating overlapping finite element meshes. *International Journal for Numerical Methods in Engineering* 2012; **91**(3):289–305.
 72. Burman E, Hansbo P. Fictitious domain finite element methods using cut elements: II. a stabilized Nitsche method. *Applied Numerical Mathematics* 2012; **62**(4):328–341, doi:10.1016/j.apnum.2011.01.008. URL <http://linkinghub.elsevier.com/retrieve/pii/S0168927411000298>.