Available online at www.sciencedirect.com
ScienceDirect
Comput. Methods Appl. Mech. Engrg. 392 (2022) 114630
Computer methods in applied mechanics and engineering
www.elsevier.com/locate/cma

A generalized multigrid method for solving contact problems in Lagrange multiplier based unfitted Finite Element method

Hardik Kothari*, Rolf Krause

Euler Institute, Università della Svizzera Italiana, Lugano, Switzerland

Received 30 June 2021; received in revised form 12 January 2022; accepted 13 January 2022
Available online 7 February 2022

Abstract

Internal interfaces in a domain could exist as a material defect or they can appear due to propagations of cracks. Discretization of such geometries and solution of the contact problem on the internal interfaces can be computationally challenging. We employ an unfitted Finite Element (FE) framework for the discretization of the domains and develop a tailored, globally convergent, and efficient multigrid method for solving contact problems on the internal interfaces. In the unfitted FE methods, structured background meshes are used and only the underlying finite element spaces are modified to incorporate the discontinuities. The non-penetration conditions on the embedded interfaces of the domains are discretized using the method of Lagrange multipliers. We reformulate the arising variational inequality problem as a quadratic minimization problem with linear inequality constraints. Our multigrid method can solve such problems by employing a tailored multilevel hierarchy of the FE spaces and a novel approach for tackling the discretized non-penetration conditions. We employ pseudo-L^2 projection-based transfer operators to construct a hierarchy of nested FE spaces from the hierarchy of non-nested meshes. The essential component of our multigrid method is a technique that decouples the linear constraints using an orthogonal transformation. The decoupled constraints are handled by a modified variant of the projected Gauss–Seidel method, which we employ as a smoother in the multigrid method. These components of the multigrid method allow us to enforce linear constraints locally and ensure the global convergence. We will demonstrate the robustness, efficiency, and level independent convergence property of the proposed method for Signorini's problem and two-body contact problems.

Keywords: XFEM; Contact problem; Multigrid methods; Unfitted finite element methods; L^2-projections

1. Introduction

Contact problems are virtually ubiquitous in the field of mechanics and engineering. An accurate and reliable simulation of the contact problem is important in many engineering applications. From the numerical modeling perspective, contact problems are challenging to solve as the contact boundary is unknown a priori. Hence, a special type of iterative scheme is needed to solve such problems, as the contact zone has to be identified during the solution process. In this work, we present contact problems in the unfitted finite element (FE) framework and introduce the problem in terms of a variational inequality. Here, we consider frictionless contact problems, where we neglect

* Corresponding author.
E-mail addresses: hardik.kothari@usi.ch (H. Kothari), rolf.krause@usi.ch (R. Krause).

https://doi.org/10.1016/j.cma.2022.114630

the tangential forces on the contact interfaces. The main contribution of this work is a novel generalized multigrid method that is developed for solving the arising quadratic minimization problem with linear inequality constraints.

In the last two decades, unfitted FE methods have seen a rise in popularity and multiple frameworks for handling unfitted geometries have emerged. Unlike traditional FE methods, these unfitted FE methods do not require a fitted mesh that describes the computational domain explicitly. The unfitted FE methods, generally, require a background mesh that encapsulates the computational domain, and the FE spaces associated with the background meshes are modified to capture the information of the domain. These methods are ideal for solving problems with complex computational domains, interface problems with discontinuous coefficients, or moving interfaces. As the background mesh and the computational domain are created independently, the interfaces/boundaries of the domain are, generally, embedded in the background mesh. For this reason, it becomes essential to enforce boundary conditions or interface conditions in a weak sense. To this end, the penalty method, the method of Lagrange multipliers, and Nitsche's method are used to impose the interface/boundary conditions. In practice, Nitsche's method is significantly more popular than the method of Lagrange multipliers and the penalty method. This is due to the fact that the penalty method is variationally inconsistent and it does not produce optimal convergence rates of the discretization error in absence of a sufficiently large penalty parameter. Nitsche's method can be regarded as a variationally consistent penalty method and due to its robustness, it is widely used in the unfitted FE methods. Nitsche's method also requires a penalty/stabilization parameter, where the parameter has to be chosen such that the coercivity of the bilinear form is ensured. The Lagrange multipliers give rise to mixed FE formulations, the linear systems of equations stemming from this discretization scheme have a saddle point structure and the method is not stable if the FE spaces do not satisfy discrete inf–sup condition. The eXtended finite element method (XFEM) was introduced as a partition of unity method to enrich the underlying FE spaces to tackle the problems in fracture mechanics with crack propagation [1,2]. A similar unfitted method, which utilized Nitsche's method for enforcing the interface condition, was introduced in [3,4]. This method evolved into CutFEM [5], where Nitsche's method is used to enforce the boundary/interface conditions and a ghost penalty type term is used for stabilization [6].

The modeling of contact problems in the context of fitted FE methods has been studied from both numerical and theoretical points of view in detail, for example, in [7–9]. In the unfitted FE framework, the initial work regarding the contact problem was carried out by Dolbow et al. in the context of XFEM to tackle the frictional sliding contact on the crack faces [10]. In the work [10], the frictional contact constraints on the crack faces were handled using the Large Time Increment (LaTIn) method [11]. In order to solve the contact problems in the unfitted FE framework, multiple approaches have been considered. The penalty formulation for solving the contact between the open crack faces was utilized in a few works [12–14]. A formal theoretical framework of Nitsche's method for solving the contact problem was later given by several authors in fitted FE framework [15–18] and unfitted FE framework for fictitious domain methods [19]. Recently, Nitsche-based contact problems in an unfitted FE framework have been considered in the context of large deformation [20]. Nitsche's method for enforcing the contact conditions gives rise to a non-smooth energy functional and to solve such problems, the generalized Newton's method was used as a solution strategy in [21]. In the context of the CutFEM solver, a LaTIn-based solution scheme was proposed for solving the contact problems, where the contact conditions are handled with Nitsche's method [22,23]. Lagrange multiplier based approaches for contact problems in unfitted methods have also been pursued in several works [24–26]. In these approaches, multiple techniques for constructing a stable multiplier space have been considered.

As noted earlier, the method of Lagrange multipliers gives rise to mixed FE formulations, and the stability of the mixed formulation is ensured only if the discrete inf–sup condition is satisfied. In the unfitted FE framework, it is shown that the most convenient approaches to construct the multiplier spaces give rise to instabilities [27,28]. To circumvent the strict requirement of satisfying the inf–sup condition, a different approach was introduced by Barbosa and Hughes [29]. In the Barbosa–Hughes approach, the restriction over the choices for FE spaces is dropped and the stability of the formulation is ensured using a stabilization term, which penalizes the jump between the multiplier and its physical interpretation. This approach was extended by Haslinger and Renard to the fictitious domain method in the unfitted FE framework [30], where an additional ghost penalty type stabilization was also used for cut elements. A different type of stabilization method was introduced by Burman and Hansbo, where the multiplier is chosen as a piecewise constant function and the stability of the saddle-point formulation is achieved by penalizing the jump of the multiplier over the element faces [31]. Alternatively, different augmented Lagrangian methods have also been considered, where Nitsche's method is derived by explicitly interpreting the multiplier in terms of primal

variable [32]. Several other approaches are also considered in the literature to create FE spaces that satisfy the discrete inf–sup condition, where a primal space with bubble-stabilized basis functions is considered [33,34]. Béchet et al. developed a stable Lagrange multiplier space based on a vital vertex algorithm [24], which was later extended by Hautefeuille et al. [35]. This method does not require any stabilization terms or modification of the primal space, only the multiplier space is designed carefully such that it satisfies the inf–sup condition and ensures optimal convergence of discretization error.

In the unfitted FE framework, the background mesh and the computational domains are independent entities, hence the elements associated with the background mesh are allowed to intersect arbitrarily. Due to this reason, the linear system of equations arising from the unfitted FE discretization can be highly ill-conditioned. Recently, many efforts have been made to develop multigrid solution strategies in context of the unfitted finite element methods [36–39]. For example, in [37], a multigrid method is developed for solving an elliptic interface problem with high contrast coefficients. While in [38], a parallel geometric multigrid method is developed for solving elliptic problems in the Finite Cell framework. Additionally, the system of equations arising from the discretization of contact conditions using the method of Lagrange multipliers can be formulated as a quadratic minimization problem with linear inequality constraints. Due to this reason, the standard multigrid methods cannot be employed to solve the contact problems. In context of fitted FE methods, the geometric multigrid methods for solving contact problems have been developed in [40–42] and the algebraic multigrid methods have been proposed in [43,44].

In our work, contrary to the more popular Nitsche's method, the method of Lagrange multipliers is used to enforce the non-penetration condition on the interface. We employ the vital vertex algorithm for constructing a stable Lagrange multiplier space. In addition, the ghost penalty stabilization term is used to control the gradients of the function on the cut elements, which in turn provides an upper bound on the condition number of the system matrix. Here, as we have a mixed FE formulation, we need to solve a primal–dual constrained optimization problem. We have circumvented this by developing a tailored multigrid method, which solves a constrained quadratic optimization problem with linear inequality constraints, where we are required to solve the problem using the primal formulation. Our generalized multigrid method utilizes the pseudo-L^2-projection to compute the transfer operator which was originally proposed in [39] and utilized in [45]. The generalized multigrid method is motivated by the monotone multigrid method proposed in [46], which was developed for solving a quadratic minimization problem with point-wise constraints arising from the variational inequalities. In the monotone multigrid method, the energy functional is minimized successively such that each iterate satisfies the constraints. An important component of the monotone multigrid is projected Gauss–Seidel (PGS) smoother, which can simultaneously minimize the energy functional and project the current iterate onto a feasible set in each local iteration. For the linear constraints, which are represented by a linear combination of several variables, the traditional PGS method is unusable. To overcome this difficulty, we introduce an orthogonal transformation of the linear constraints and propose a novel variant of the PGS method that can handle such constraints. The overall methodology proposed in our work, the particular combination of the discretization and the multigrid method, is computationally less expensive than employing a variant of Nitsche's method.

In the context of the contact problems in the unfitted methods, similar to our discretization scheme, Nitsche's method also employs the ghost penalty stabilization term [19,22]. Additionally, in the context of fitted FE methods for solving the contact problems, it has been shown that coercivity of the symmetric variant of bilinear form can only be ensured if a sufficiently large stabilization method is used [15]. While for the non-symmetric variant the coercivity of the bilinear form is ensured for any non-zero stabilization parameter [16]. The ghost penalty term added to the bilinear form of a symmetric variant of Nitsche's method provides additional coercivity, however, the stabilization parameter has to be chosen carefully to ensure the coercivity. One can employ generalized eigenvalue problems to estimate the stabilization parameter or a lifting operator scheme to remove the dependence of the stabilization parameter on the coercivity [45]. Thus, (symmetric) Nitsche's method also requires the ghost penalty term, and then parameter selection for the stabilization parameter has to be carried out. Additionally, the non-penetration constraints have to be replaced with an approximate nonlinear function which has to be penalized/enforced in Nitsche's method [47]. Due to this reason, Nitsche's method remains in the primal formulation but it loses consistency as small penetration may occur if the penalty parameter is not chosen carefully. A larger value of the penalty parameter improves the approximation of non-penetration conditions, but it can cause the resulting linear system of the equation to become severely ill-conditioned. The addition of such a term to the underlying Nitsche's formulation causes it to become nonlinear, and variants of the Newton method have to be employed to solve such problems [21]. At each

Newton iteration, we have to reassemble the tangent system of Nitsche's formulation corresponding to the current active set. Therefore, by employing the proposed combination of the discretization and solution method, we have to solve a system with n-unknowns only one time with a multigrid method, while for Nitsche's method we would need to solve a system with n-unknowns several times and during each iteration, we have to reassemble the bilinear and linear forms.

The outline of this paper is given as follows. We introduce the two-body contact problem and discuss the unfitted FE discretization in detail in Section 2. In Section 3, we introduce our generalized multigrid method and explain each component of the multigrid method in detail. We discuss an orthogonalization strategy to decouple the linear constraints and introduce a modified PGS method to tackle the decoupled constraints. Lastly, we present the results of numerical experiments in Section 4. We study the discretization error and the performance of the multigrid method with respect to several parameters. We show the robustness of the proposed generalized multigrid method for Signorini's problem and the two-body contact problem and also compare the performance of our multigrid method with other solution strategies.

2. Two-body contact problem in an unfitted FE framework

In this section, we introduce the two-body contact problem within the unfitted finite element framework. Here, we assume that the contact between the two bodies takes place on an embedded interface. The Dirichlet boundary and Neumann boundaries can also be assumed to be embedded, but in order to simplify the presentation of the problem, these boundaries are assumed to be fitted with the background mesh.

2.1. Problem description

We assume two elastic bodies $\Omega^1, \Omega^2 \in \mathbb{R}^d$, $d \in \{2, 3\}$, with Lipschitz continuous boundaries Γ^1, Γ^2. The bodies are assumed to be subjected to volume forces $\boldsymbol{f}^i : \Omega^i \to \mathbb{R}^d$ and traction/surface forces on the Neumann boundary $\boldsymbol{t}_N^i : \Gamma_N^i \to \mathbb{R}^d$. Both bodies undergo deformation due to the influence of these external forces. A material point $\boldsymbol{X} \in \Omega$ in the undeformed state moves to the location $\boldsymbol{X} + \boldsymbol{u}$ after the deformation. Here, the vector-valued quantity $\boldsymbol{u} : \Omega \to \mathbb{R}^d$ denotes the displacement field of the material point $\boldsymbol{X}$, denoted as $\boldsymbol{u} := \boldsymbol{u}(\boldsymbol{X})$. The boundary Γ is decomposed into three parts: the Dirichlet boundary Γ_D, the Neumann boundary Γ_N, and apriori unknown contact boundary Γ_c.

In elastostatics, the displacement field $\boldsymbol{u} := (\boldsymbol{u}^1, \boldsymbol{u}^2)$ can be given as a solution of the following boundary value problem:

$$\begin{aligned} -\nabla \cdot \boldsymbol{\sigma}^i(\boldsymbol{u}^i) &= \boldsymbol{f}^i \qquad \text{in } \Omega^i, \\ \boldsymbol{u}^i &= \boldsymbol{u}_D^i \qquad \text{on } \Gamma_D^i, \\ \boldsymbol{\sigma}^i(\boldsymbol{u}^i) \cdot \boldsymbol{n}^i &= \boldsymbol{t}_N^i \qquad \text{on } \Gamma_N^i, \end{aligned} \tag{1}$$

where $\boldsymbol{n}^i$ denotes the outward normal on the Neumann boundary Γ_N^i. We note that unlike Dirichlet and Neumann boundaries the contact boundary is shared between both bodies, $\Gamma_c = \Gamma_c^1 = \Gamma_c^2$. These parts of the boundaries are assumed to be disjoint and the contact boundary is assumed to have a positive measure, i.e., $\text{meas}_{d-1}(\Gamma_c) > 0$. In (1), we denote the Cauchy stress tensor as $\boldsymbol{\sigma} := \boldsymbol{\sigma}(\boldsymbol{u})$. Here, we assume the bodies Ω^1, Ω^2 to be linear elastic, where the constitutive law is provided by Hooke's law

$$\boldsymbol{\sigma}^i(\boldsymbol{u}^i) = 2\mu^i \boldsymbol{\varepsilon}(\boldsymbol{u}^i) + \lambda^i \text{tr}(\boldsymbol{\varepsilon}(\boldsymbol{u}^i))\boldsymbol{I} \qquad \text{for } i \in \{1, 2\},$$

where λ^i and μ^i are the Lamé parameters, $\text{tr}(\cdot)$ denotes the trace operator, $\boldsymbol{I}$ is second order identity tensor, and the linearized strain tensor $\boldsymbol{\varepsilon} := \boldsymbol{\varepsilon}(\boldsymbol{u})$ is defined as $\boldsymbol{\varepsilon}(\boldsymbol{u}) := \frac{1}{2}\left(\nabla \boldsymbol{u} + (\nabla \boldsymbol{u})^{\mathsf{T}}\right)$.

We assume that a gap function $g_c : \Gamma_c \to \mathbb{R}^+$ is given between two bodies in the direction of outward normal $\boldsymbol{n}$, where the outward normal is defined as $\boldsymbol{n} = \boldsymbol{n}^1 = -\boldsymbol{n}^2$. The point-wise gap in the displacement fields for both domains is defined as

$$[\![\boldsymbol{u} \cdot \boldsymbol{n}]\!] := \boldsymbol{u}^1 \cdot \boldsymbol{n}^1 + \boldsymbol{u}^2 \cdot \boldsymbol{n}^2 = (\boldsymbol{u}^1 - \boldsymbol{u}^2) \cdot \boldsymbol{n}^1.$$

The non-penetration condition on the possible contact boundary Γ_c is given as in (2a). The contact pressure or stress developed in the normal direction on Γ_c is compressive (2b). We also decompose the traction vector at the

contact boundary into the normal and tangential components, given as $\boldsymbol{\sigma n} = \sigma_n \cdot \boldsymbol{n} + \boldsymbol{\sigma}_t$, where $\sigma_n = \boldsymbol{n} \cdot \boldsymbol{\sigma n}$. The third contact condition is given as complementarity condition, as in (2c), which ensures that the gap between two bodies is zero in presence of non-zero contact pressure and the contact pressure is zero in absence of contact. As we are considering the frictionless contact problem, the body is allowed to move freely in the tangential direction and the induced tangential stresses are given as in (2d). The frictionless linearized contact conditions are given as follows:

$$[\![\boldsymbol{u} \cdot \boldsymbol{n}]\!] - g_c \leqslant 0 \quad \text{on } \Gamma_c, \tag{2a}$$

$$\sigma_n \leqslant 0 \quad \text{on } \Gamma_c, \tag{2b}$$

$$([\![\boldsymbol{u} \cdot \boldsymbol{n}]\!] - g_c)\sigma_n = 0 \quad \text{on } \Gamma_c, \tag{2c}$$

$$\boldsymbol{\sigma}_t = 0 \quad \text{on } \Gamma_c. \tag{2d}$$

In contact mechanics these conditions are known as Hertz–Signorini–Moreau conditions for frictionless contact, while in optimization literature they are known as Karush–Kuhn–Tucker (KKT) conditions of the constraints.

Remark. In this work, we also consider Signorini's contact problem in the unfitted FE framework. In Signorini's problem, a contact between a linear elastic body and a rigid foundation is considered and the gap function is computed as a distance from apriori unknown contact boundary to the rigid foundation. The non-penetration condition for this problem is given as, $\boldsymbol{u} \cdot \boldsymbol{n} - g_c \leqslant 0$ on Γ_c.

2.2. An unfitted FE discretization

In this section, we discuss the discretization of the two-body contact problem. For simplicity, we assume that only the contact boundary is not fitted with the mesh, while Dirichlet and Neumann boundaries are fitted.

We assume a shape regular, quasi-uniform, conforming quadrilateral mesh $\widetilde{\mathcal{T}}_h$ of a polygonal domain $\widetilde{\Omega}$. The polygonal domain $\widetilde{\Omega}$ is define as $\widetilde{\Omega} \supseteq \Omega = \Omega^1 \cup \Omega^2$. The contact boundary Γ_c is assumed to be resolved sufficiently well by the mesh $\widetilde{\mathcal{T}}_h$ and the curvature of the boundary is assumed to be bounded. Let h_K be the diameter of the element K, and mesh size is defined as $h = \max_{K \in \widetilde{\mathcal{T}}_h} h_K$. We define an active mesh, which is strictly intersected by the domains Ω^1, Ω^2 as

$$\mathcal{T}_h^i = \{K \in \widetilde{\mathcal{T}}_h : K \cap \Omega^i \neq \emptyset\}, \quad \text{for } i \in \{1, 2\}.$$

For simplicity, we define a domain $\Omega_h^i = \cup_{K \in \mathcal{T}_h^i} K$, where $\Omega^i \subset \Omega_h^i$ for $i \in \{1, 2\}$. The active meshes exclude all elements that are neither intersected by the boundary Γ_c nor are in the interior of the domain. We define a set of elements that are intersected by the contact boundary Γ_c as

$$\mathcal{T}_{h,\Gamma_c} = \{K \in \widetilde{\mathcal{T}}_h : K \cap \Gamma_c \neq \emptyset\}.$$

For all elements $K \in \mathcal{T}_{h,\Gamma_c}$, let $K_\Omega := K \cap \Omega$ be part of K in domain Ω. The elements $K \in \mathcal{T}_h \setminus \mathcal{T}_{h,\Gamma_c}$ are strictly in the interior of domain Ω. For all $K \in \mathcal{T}_{h,\Gamma_c}$, let $\Gamma_K := \Gamma_c \cap K$ be part of Γ_c in K.

We define a continuous FE space over the mesh $\widetilde{\mathcal{T}}_h$ as

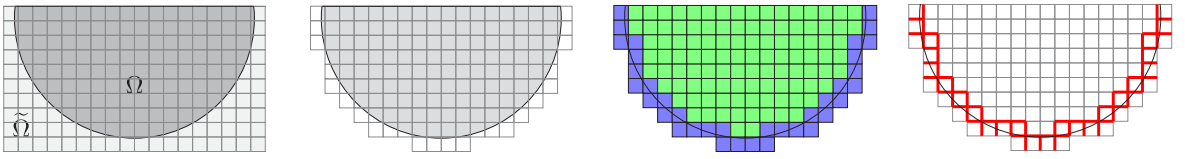
$$\widetilde{\boldsymbol{\mathcal{V}}}_h = \{\boldsymbol{v} \in [H^1(\widetilde{\mathcal{T}}_h)]^d : \boldsymbol{v}|_K \in \mathcal{Q}_1(K), \boldsymbol{v}|_{(\partial\widetilde{\mathcal{T}}_h)_D} = 0, \forall K \in \widetilde{\mathcal{T}}_h\}, \tag{3}$$

where $\mathcal{Q}_1$ denotes the space of piecewise bilinear functions. Following the original XFEM literature [2], we define a characteristic function of a computational domain Ω^i for $i \in \{1, 2\}$, as

$$\chi_{\Omega^i} : \mathbb{R}^d \to \mathbb{R}, \qquad \chi_{\Omega^i}(\boldsymbol{X}) = \begin{cases} 1 & \forall \boldsymbol{X} \in \overline{\Omega}^i, \\ 0 & \text{otherwise.} \end{cases} \tag{4}$$

The function space $\widetilde{\boldsymbol{\mathcal{V}}}_h$ is spanned by nodal basis functions $\widetilde{\Phi}_h = \{\widetilde{\phi}_h^p\}_{p \in \widetilde{\mathcal{N}_h}}$, where $\widetilde{\mathcal{N}_h}$ denotes the set of nodes of the background mesh $\widetilde{\mathcal{T}}_h$. The characteristic function is used to restrict the support of the finite element space $\widetilde{\boldsymbol{\mathcal{V}}}_h$ to the respective domain Ω^i thus $\boldsymbol{\mathcal{V}}_h^i = \chi_{\Omega^i}(\boldsymbol{X})\widetilde{\boldsymbol{\mathcal{V}}}_h$. We seek the approximation $\boldsymbol{u}_h = (\boldsymbol{u}_h^1 \oplus \boldsymbol{u}_h^2)$ in space $\boldsymbol{\mathcal{V}}_h = \boldsymbol{\mathcal{V}}_h^1 \oplus \boldsymbol{\mathcal{V}}_h^2$. We define a set of nodes on an active mesh $\mathcal{T}_h^i$ associated with a domain Ω^i as

$$\mathcal{N}_h^i := \{p \in \widetilde{\mathcal{N}_h} : \text{supp}(\widetilde{\phi}_h^p) \cap \Omega^i \neq \emptyset\} \quad \text{for } i \in \{1, 2\}.$$



(a) A quadrilateral mesh $\widetilde{\mathcal{T}}_h$ of a domain $\widetilde{\Omega}$ is used as a background mesh, where a semicircular domain $\Omega \subset \widetilde{\Omega}$.

(b) The mesh $\mathcal{T}_h$ is strictly intersected by the domain Ω.

(c) The mesh $\mathcal{T}_{h,\Gamma_c}$ is shaded in blue, while the interior mesh $\mathcal{T}_h \backslash \mathcal{T}_{h,\Gamma_c}$ is shaded in green.

(d) The set of faces $\mathcal{G}_{h,\Gamma_c}$ is shown in the red.

Fig. 1. An example of a domain Ω with a background mesh $\widetilde{\mathcal{T}}_h$.

We now define a "cut" basis function associated with a node p as

$$\phi_h^p = \chi_{\Omega^i}(\boldsymbol{X})\widetilde{\phi}_h^p \qquad \forall p \in \mathcal{N}_h^i,\ i \in \{1,2\}.$$

The function space $\boldsymbol{\mathcal{V}}_h^i$ is spanned by nodal basis functions $\Phi_h^i = \{\phi_h^p\}_{p\in\mathcal{N}_h^i}$. We define the FE space $\boldsymbol{\mathcal{V}}_h = \text{span}\{\Phi_h\}$, where $\Phi_h = \Phi_h^1 \cup \Phi_h^2$, and the set of nodes associated with the mesh $\mathcal{T}_h$ is given by $\mathcal{N}_h = \mathcal{N}_h^1 \cup \mathcal{N}_h^2$.

Remark. For Signorini's problem, the characteristic function (4) is defined only on the domain Ω^1. An example of a domain Ω embedded in the background mesh $\widetilde{\mathcal{T}}_h$, the active mesh $\mathcal{T}_h$, the interior mesh, and the cut mesh are shown in Fig. 1.

2.3. Variational formulation

The variational formulation of the two-body contact problem using the principle of virtual work is given as:

$$\text{find } \boldsymbol{u}_h \in \boldsymbol{\mathcal{K}}_h \text{ such that,} \quad a(\boldsymbol{u}_h, \boldsymbol{v}_h - \boldsymbol{u}_h) \geqslant F(\boldsymbol{v}_h - \boldsymbol{u}_h) \quad \forall \boldsymbol{v}_h \in \boldsymbol{\mathcal{K}}_h, \tag{5}$$

where $a(\cdot,\cdot) : \boldsymbol{\mathcal{V}}_h \times \boldsymbol{\mathcal{V}}_h \to \mathbb{R}$ is a symmetric, continuous and coercive bilinear form, and $F(\cdot) : \boldsymbol{\mathcal{V}}_h \to \mathbb{R}$ denotes continuous and bounded linear form. The bilinear and the linear forms are defined as

$$\begin{aligned} a(\boldsymbol{u}_h, \boldsymbol{v}_h) &= \sum_{i=1}^{2} \int_{\Omega^i} \boldsymbol{\sigma}^i(\boldsymbol{u}_h^i) : \boldsymbol{\varepsilon}(\boldsymbol{v}_h^i) d\Omega = \sum_{i=1}^{2} \left(\int_{\Omega^i} 2\mu^i \boldsymbol{\varepsilon}(\boldsymbol{u}_h^i) : \boldsymbol{\varepsilon}(\boldsymbol{v}_h^i) d\Omega + \int_{\Omega^i} \lambda^i \text{tr}(\boldsymbol{\varepsilon}(\boldsymbol{u}_h^i))\text{tr}(\boldsymbol{\varepsilon}(\boldsymbol{v}_h^i)) d\Omega \right), \\ F(\boldsymbol{v}_h) &= \sum_{i=1}^{2} \left(\int_{\Omega^i} \boldsymbol{f}^i \boldsymbol{v}_h^i d\Omega + \int_{\Gamma_N} \boldsymbol{t}_N^i \boldsymbol{v}_h^i d\Gamma \right). \end{aligned} \tag{6}$$

Here, we employ the method of Lagrange multipliers to enforce the non-penetration contact conditions (2a), where the multiplier space is constructed by employing the vital vertex algorithm [24,35]. This is due to the fact that the Lagrange multiplier formulation does not require modification of the primal formulation, and the contact condition can be handled by the multipliers implicitly. Whereas, Nitsche's formulation for the contact problem is more complex as we have to handle the non-penetration conditions in the primal formulation.

We introduce the multiplier space $\mathcal{M}_h \subseteq H^{-\frac{1}{2}}(\Gamma_c)$ and define the bilinear form $b(\cdot,\cdot) : \mathcal{M}_h \times \boldsymbol{\mathcal{V}}_h \to \mathbb{R}$, given as

$$b(\mu_h, \boldsymbol{u}_h) := \sum_{K\in\mathcal{T}_{h,\Gamma_c}} \int_{\Gamma_K} \mu_h [\![\boldsymbol{u}_h \cdot \boldsymbol{n}]\!]\, d\Gamma \qquad \forall \mu_h \in \mathcal{M}_h, \forall \boldsymbol{u}_h \in \boldsymbol{\mathcal{V}}_h, \tag{7}$$

and the linear form $G(\cdot) : \mathcal{M}_h \to \mathbb{R}$, given as

$$G(\mu_h) := \sum_{K\in\mathcal{T}_{h,\Gamma_c}} \int_{\Gamma_K} \mu_h g_c\, d\Gamma \qquad \forall \mu_h \in \mathcal{M}_h.$$

Finally, we can define a space of admissible displacements that satisfy the contact conditions as

$$\boldsymbol{\mathcal{K}}_h := \{\boldsymbol{v}_h \in \boldsymbol{\mathcal{V}}_h : b(\mu_h, \boldsymbol{v}_h) \leqslant G(\mu_h), \forall \mu_h \in \mathcal{M}_h\}. \tag{8}$$

The space of admissible displacements $\mathcal{K}_h$ is a closed convex subset of the FE space $\mathcal{V}_h$. Due to the inequality condition in (5), the contact problem is inherently nonlinear.

2.3.1. Ghost penalty stabilization

In unfitted FE methods, a background mesh can intersect with a boundary/interface of the computational domain of arbitrary shape, hence the elements are allowed to be cut arbitrarily by the boundary/interface. In general, this flexibility can result in disproportionally cut elements, which might not be shape regular anymore. For this reason, the bound on the gradient of a function can become arbitrarily weak for the unfortunately cut elements. By adding a ghost penalty term [6], we regain control over the gradients of the function on cut elements with very small support, and by extension, we can overcome the issue of ill-conditioning. This stabilization term has to be chosen in such a way that it provides sufficient stability and stays weakly consistent with the original formulation for smooth functions. We define a set of faces $\mathcal{G}^i_{h,\Gamma_c}$ for each subdomain Ω^i as

$$\mathcal{G}^i_{h,\Gamma_c} = \{G \subset \partial K \mid K \in \mathcal{T}^i_{h,\Gamma_c},\ \partial K \cap \partial \mathcal{T}^i_h = \emptyset\} \quad \text{for } i \in \{1,2\}.$$

An example of the set of faces $\mathcal{G}_{h,\Gamma_c}$ in the context of Signorini's problem can be seen in Fig. 1(d). The ghost penalty term is enforced on the faces $G \in \mathcal{G}_{h,\Gamma_c}$, and it is defined as

$$j(\boldsymbol{u}_h, \boldsymbol{v}_h) = \sum_{i=\{1,2\}} \sum_{G \in \mathcal{G}^i_{h,\Gamma_c}} \int_G \epsilon_G h_G (2\mu^i + \lambda^i) [\![\nabla \mathcal{E}_h \boldsymbol{u}_h \cdot \boldsymbol{n}_G]\!] [\![\nabla \mathcal{E}_h \boldsymbol{v}_h \cdot \boldsymbol{n}_G]\!]\, dG, \tag{9}$$

where h_G is the diameter of the face G, $\boldsymbol{n}_G$ denotes a unit normal to face G, ϵ_G is a positive constant and λ^i, μ^i denote the Lamé parameters associated with the either domain [48]. Here, $\mathcal{E}_h$ denotes the canonical extension of the function from the domain to the background mesh, which is defined as $\mathcal{E}_h : \mathcal{V}_h|_{K_\Omega} \to \widetilde{\mathcal{V}}_h|_K$. This ghost penalty term is enforced in the normal derivatives of the displacement field. A different approach is also considered in [22], where the ghost penalty term is enforced in the normal derivatives of the stress field.

Now, we modify the variational formulation of the two-body contact problem by adding the ghost penalty stabilization term. The updated variational problem is defined as:

$$\text{find } \boldsymbol{u}_h \in \mathcal{K}_h \text{ such that} \quad a_j(\boldsymbol{u}_h, \boldsymbol{v}_h - \boldsymbol{u}_h) \geqslant F(\boldsymbol{v}_h - \boldsymbol{u}_h) \quad \forall \boldsymbol{v} \in \mathcal{K}_h, \tag{10}$$

with the bilinear form $a_j(\boldsymbol{u}_h, \boldsymbol{v}_h) := a(\boldsymbol{u}_h, \boldsymbol{v}_h) + j(\boldsymbol{u}_h, \boldsymbol{v}_h)$.

2.3.2. Discretization of non-penetration condition

We have used the method of Lagrange multipliers to discretize and enforce the non-penetration contact condition. To achieve optimal convergence rates of the discretization method, the choice of the FE spaces for primal variable $\boldsymbol{u}_h \in \mathcal{V}_h$ and the dual variable $\lambda_h \in \mathcal{M}_h$ is crucial. In the unfitted FE framework, the most convenient options for $\mathcal{V}_h$ and $\mathcal{M}_h$ are very rarely stable. As, the method of multipliers is stable only if the following discrete inf–sup condition is satisfied

$$\inf_{\mu_h \in \mathcal{M}_h} \sup_{\boldsymbol{v}_h \in \mathcal{V}_h} \frac{b(\mu_h, \boldsymbol{v}_h)}{\|\mu_h\|_{H^{-\frac{1}{2}}(\Gamma_h),h} |||\boldsymbol{v}_h|||_a} \geqslant \beta > 0, \tag{11}$$

where the constant β does not dependent on mesh-size h. Also, the mesh dependent interface norm is defined as $\|\mu_h\|^2_{H^{-\frac{1}{2}}(\Gamma_h),h} := \sum_{K \in \mathcal{T}_{h,\Gamma}} h \|\mu_h\|^2_{L^2(\Gamma_K)}$ and $|||\boldsymbol{v}_h|||^2_a := a(\boldsymbol{v}_h, \boldsymbol{v}_h) + j(\boldsymbol{v}_h, \boldsymbol{v}_h)$ denotes a broken energy-norm. If the inf–sup condition (11) is not satisfied, it can give rise to spurious modes in the discrete Lagrange multiplier space. The effect of the spurious modes in the solution can be observed as locking phenomena on the interface. In the unfitted FE framework, it is not trivial to create an optimal multiplier space, as the interface is not resolved by the background mesh.

We employ the vital vertex algorithm to create a stable multiplier space, where the primal space is kept the same and a coarser multiplier space is chosen. Such an approach of constructing a coarser multiplier space can be found in several works in the context of unfitted FE framework [24,49]. Béchet et al. developed a stable multiplier space based on a vital vertex algorithm [24]. This method does not require any stabilization term for the Lagrange multiplier and the primal space $\mathcal{V}_h$ does not require any modification. Only the multiplier space $\mathcal{M}_h$ is designed carefully such that it satisfies the inf–sup condition.

We define a skeleton associated with the mesh $\mathcal{T}_{h,\Gamma_c}$, as $\mathcal{S}_{h,\Gamma_c} := \cup_{K\in\mathcal{T}_{h,\Gamma_c}} \partial K$. The set of vertices is defined by points that result from the intersection interface Γ_c with the skeleton $\mathcal{S}_{h,\Gamma_c}$, given as $\mathcal{V}_{h,\Gamma_c} := \mathcal{S}_{h,\Gamma_c} \cap \Gamma_c$. The set of vertices $\mathcal{V}_{h,\Gamma_c}$ is later decomposed into a set of vital vertices $\mathcal{V}^V_{h,\Gamma_c}$ and a set of non-vital vertices $\mathcal{V}^N_{h,\Gamma_c}$. This decomposition is carried out based on two main rules: a vertex is declared to be vital if it is not connected to another vital vertex; a non-vital vertex has to be connected to at least one vital vertex. The dimension of the multiplier space $\mathcal{M}_h$ is given as $|\mathcal{V}^V_{h,\Gamma_c}|$. A set of nodes $\mathcal{N}_{h,\Gamma_c}$ includes all nodes that are endpoints of the cut-edges, given as $\mathcal{N}_{h,\Gamma_c} := \{q \in \mathcal{N}_h : \phi^q_h|_{\Gamma_c} \neq 0\}$. The set $\mathcal{N}_{h,\Gamma_c}$ is later divided into a set of active nodes $\mathcal{N}^A_{h,\Gamma_c}$ and inactive nodes $\mathcal{N}^I_{h,\Gamma_c}$. Here, the set of active nodes $\mathcal{N}^A_{h,\Gamma_c}$ are defined as the endpoints of the edges on which the vital vertices are located and the inactive nodes are given as $\mathcal{N}^I_{h,\Gamma_c} = \mathcal{N}_{h,\Gamma_c} \setminus \mathcal{N}^A_{h,\Gamma_c}$. For each vital vertex $p \in \mathcal{V}^V_{h,\Gamma_c}$, we define an associated basis function ψ^p_h. These basis functions are defined as linear combination of trace of higher dimensional basis function ϕ^q_h on interface Γ_c, where $q \in \mathcal{N}_{h,\Gamma_c}$, given as

$$\psi^p_h := \sum_{q\in\mathcal{N}_{h,\Gamma_c}} w_{pq}\phi^q_h|_{\Gamma_c},$$

where w_{pq} are coefficients of the linear combination chosen such that $\psi^p_h(q) = \delta_{pq}$ for all $q \in \mathcal{V}^V_{h,\Gamma_c}$.

In Fig. 2, we can see a set of all vertices (Fig. 2(a)), a set of vital vertices $\mathcal{V}_{h,\Gamma_c}$ (Fig. 2(b)), a set of all nodes associated with the cut elements $\mathcal{N}_{h,\Gamma_c}$ (Fig. 2(c)), and an identified set of active nodes $\mathcal{N}^A_{h,\Gamma_c}$ and inactive nodes $\mathcal{N}^I_{h,\Gamma_c}$ (Fig. 2(d)) for Signorini's problem.

2.4. Local basis transformation

The non-penetration constraints in the contact problem are given by the relative displacement of the bodies in the normal direction. Thus, the constraint at any node is given by the coupling of the degrees of freedom (DoFs) on the node. To create the constraint matrix such that the non-penetration condition is enforced only on one DoF per node, we transform the system into a new basis.

Let $\{\boldsymbol{E}_i\}_{i=1,\dots,d}$ be the Euclidean basis of $\mathbb{R}^d$ and $\boldsymbol{n}_p$ be the outward unit normal on the node p. On each node $p \in \mathcal{N}_{h,\Gamma_c}$, we define a new basis $\boldsymbol{e}_1(p) = \boldsymbol{n}_p$ and also change $\{\boldsymbol{e}_i\}_{i=2,\dots,d}$ such that this redefined basis is also orthonormal, while for all $q \in \mathcal{N}_h \setminus \mathcal{N}_{h,\Gamma_c}$ the definition of the Euclidean basis remains the same. This approach was introduced for Signorini's problems [50] and has been effectively applied to multi-body contact problems [41,42,51]. The transformed basis is constructed using a local Householder transformation on $\mathbb{R}^d$, given as

$$\boldsymbol{O}_{pp} = \boldsymbol{I} - 2(\boldsymbol{w}_p \otimes \boldsymbol{w}_p) \qquad \forall p \in \mathcal{N}_{h,\Gamma_c},$$

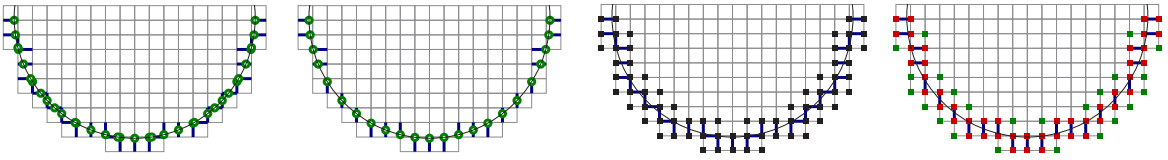
where the vector $\boldsymbol{w}_p$ is computed by $\boldsymbol{w}_p = (\boldsymbol{n}_p - \boldsymbol{E}_1)/\|\boldsymbol{n}_p - \boldsymbol{E}_1\|_2$. Now, due to the Householder transformation for all $p \in \mathcal{N}_{h,\Gamma_c}$ we can uniquely define local unit vectors as $\boldsymbol{e}_i(p) = \boldsymbol{O}_{pp}\boldsymbol{E}_i(p)$. While, we define $\boldsymbol{O}_{qq} = \boldsymbol{I}$ for all $q \in \mathcal{N}_h \setminus \mathcal{N}_{h,\Gamma_c}$, which ensure that the basis system on those nodes remain unchanged, i.e., $\boldsymbol{e}_i(q) = \boldsymbol{E}_i(q)$. Thus, by using these local transformation matrices, we can construct a global matrix $\boldsymbol{O} \in \mathbb{R}^{nd\times nd}$ where $nd = |\mathcal{N}_h| \cdot d$ and $\boldsymbol{O} = \oplus_{p\in\mathcal{N}_h} \boldsymbol{O}_{pp}$, which is an orthonormal matrix with the properties, $\boldsymbol{O}\boldsymbol{O}^\mathsf{T} = \boldsymbol{O}^\mathsf{T}\boldsymbol{O} = \boldsymbol{I}$ and $\boldsymbol{O} = \boldsymbol{O}^\mathsf{T}$.

This transformation decouples and locally modifies the constraints and it is only applicable in the normal direction. The bilinear form (7) for the two-body contact problem can be reformulated as

$$\bar{b}(\mu_h, \boldsymbol{u}_h) := \sum_{K\in\mathcal{T}_{h,\Gamma_c}} \int_{\Gamma_K} \mu_h[\![\boldsymbol{u}_h \cdot \boldsymbol{E}_1]\!]\, d\Gamma_K \quad \forall \mu_h \in \mathcal{M}_h, \forall \boldsymbol{u}_h \in \boldsymbol{\mathcal{V}}_h. \tag{12}$$

In the next section, we discuss the algebraic formulation of the contact problem (10) and also discuss the effect of the local basis transformation algebraically.

Remark. In unfitted methods, the interface Γ_c is defined as the zero-isoline of level set of a function $\Lambda(\boldsymbol{X})$. The gradient of such level set function Λ is used to define unit normal at any point in the domain Ω, given as $\boldsymbol{n} = \frac{\nabla\Lambda}{|\nabla\Lambda|}$ [52]. Thus, even if a node $p \in \mathcal{N}_{h,\Gamma_c}$ is not on the interface Γ_c, we can compute unit normals on any nodes in $\mathcal{N}_{h,\Gamma_c}$ using the above formula.



(a) Set of vertices $\mathcal{V}_{h,\Gamma_c}$ are marked with a green circle, all cut edges which are intersected by the boundary are marked with blue line.

(b) Vital vertices $\mathcal{V}^V_{g,\Gamma_c}$ are marked with a green circle, the cut edges on which the vital vertices lie are marked with blue line.

(c) The nodes $\mathcal{N}_{h,\Gamma_c}$ that belong to the sub-mesh $\mathcal{T}_{h,\Gamma_c}$ are marked with black squares

(d) All active nodes $\mathcal{N}^A_{h,\Gamma_c}$ are marked with red squares, and the inactive nodes $\mathcal{N}^I_{h,\Gamma_c}$ are marked with green squares

Fig. 2. Different type of nodes characterized by the vital-vertex algorithm.

2.5. Algebraic formulation

The abstract variational problem for the contact problem can be reformulated as an optimization problem with inequality constraints, given as

$$\begin{aligned} &\min_{\boldsymbol{u}_h \in \boldsymbol{\mathcal{V}}_h} \mathcal{J}(\boldsymbol{u}) = \frac{1}{2} a_j(\boldsymbol{u}_h, \boldsymbol{u}_h) - F(\boldsymbol{u}_h) \\ &\text{subject to} \quad b(\mu_h, \boldsymbol{u}_h) \leqslant G(\mu_h) \quad \forall \mu_h \in \mathcal{M}_h. \end{aligned} \tag{13}$$

The above minimization problem can be written in an algebraic formulation using the local basis transformation introduced in the previous section. We denote local entries of the stiffness matrix and the right-hand side as

$$A_{pq} = (a_j(\phi_h^p \boldsymbol{E}_i, \phi_h^q \boldsymbol{E}_k))_{i,k=1,\dots,d}, \quad b_p = (F(\phi_h^p \boldsymbol{E}_i))_{i=1,\dots,d}. \tag{14}$$

The global stiffness matrix and the right-hand side vector can be assembled as

$$\boldsymbol{A} = (A_{pq})_{p,q \in \mathcal{N}_h}, \quad \boldsymbol{b} = (b_p)_{p \in \mathcal{N}_h}.$$

The bilinear form $b(\cdot,\cdot)$ can be decomposed into two parts, $b^1(\cdot,\cdot) : \mathcal{M}_h \times \boldsymbol{\mathcal{V}}_h^1 \to \mathbb{R}$ and $b^2(\cdot,\cdot) : \mathcal{M}_h \times \boldsymbol{\mathcal{V}}_h^2 \to \mathbb{R}$ associated with domain Ω^1 and Ω^2, respectively, written as

$$b(\mu_h, \boldsymbol{u}_h) = b^1(\mu_h, \boldsymbol{u}_h^1) - b^2(\mu_h, \boldsymbol{u}_h^2),$$

where

$$b^1(\mu_h, \boldsymbol{u}_h^1) = \sum_{K \in \mathcal{T}_{h,\Gamma_c}} \int_{\Gamma_K} \mu_h(\boldsymbol{u}_h^1 \cdot \boldsymbol{n})\, d\Gamma \quad \text{and} \quad b^2(\mu_h, \boldsymbol{u}_h^2) = \sum_{K \in \mathcal{T}_{h,\Gamma_c}} \int_{\Gamma_K} \mu_h(\boldsymbol{u}_h^2 \cdot \boldsymbol{n})\, d\Gamma_K.$$

This decomposition allows us to write the local entries of the constraint matrices as

$$B^1_{rk} = (b^1(\psi_h^r, \phi_h^k \boldsymbol{E}_i))_{i=1,\dots,d}, \qquad B^2_{rl} = (b^2(\psi_h^r, \phi_h^l \boldsymbol{E}_i))_{i=1,\dots,d}, \qquad g_r = G(\psi_h^r), \tag{15}$$

where ψ_h^r, ϕ_h^k and ϕ_h^l denote the basis functions associated with nodes r, k, l in the FE spaces $\mathcal{M}_h$, $\boldsymbol{\mathcal{V}}_h^1$ and $\boldsymbol{\mathcal{V}}_h^2$, respectively. Thus, the entries of the constraint matrix $\boldsymbol{B}$ and the gap vector $\boldsymbol{g}$ are given as

$$\boldsymbol{B}^1 = (B^1_{rk})_{r \in \mathcal{V}_{h,\Gamma_c}, k \in \mathcal{N}_h^1}, \quad \boldsymbol{B}^2 = (B^2_{rl})_{r \in \mathcal{V}_{h,\Gamma_c}, l \in \mathcal{N}_h^2}, \quad \boldsymbol{g} = (g_r)_{r \in \mathcal{V}_{h,\Gamma_c}},$$

where $\mathcal{N}_h^1$ and $\mathcal{N}_h^2$ denote the set of nodes of the active meshes associated with each body, respectively, and $\mathcal{V}_{h,\Gamma_c}$ denotes the set of vital vertices. The matrix $\boldsymbol{B}$ can be constructed as $\boldsymbol{B} = \boldsymbol{B}^1 - \boldsymbol{B}^2$. Now, we can write the algebraic formulation of the constraint minimization problem (13) as

$$\begin{aligned} &\min_{\boldsymbol{x} \in \mathbb{R}^{nd}} J(\boldsymbol{x}) = \frac{1}{2} \boldsymbol{x}^\top \boldsymbol{A} \boldsymbol{x} - \boldsymbol{x}^\top \boldsymbol{b} \\ &\text{subject to} \quad \boldsymbol{B}\boldsymbol{x} \leqslant \boldsymbol{g}, \end{aligned} \tag{16}$$

where $\boldsymbol{x}, \boldsymbol{b} \in \mathbb{R}^{nd}$, $\boldsymbol{A} \in \mathbb{R}^{nd \times nd}$, $\boldsymbol{B} \in \mathbb{R}^{m \times nd}$, $\boldsymbol{g} \in \mathbb{R}^m$, $m \ll nd$ and $\operatorname{rank}(\boldsymbol{B}) = m$. Here, $\boldsymbol{x}$ denotes the unknown displacements, and $\boldsymbol{g}$ denotes the gap between two bodies on the contact boundary.

As we have changed the definition of the Euclidean basis (in Section 2.4), we have to also modify the problem algebraically. The (symmetric) matrix $\boldsymbol{O}$ can be used to transform the variables into the new basis as $\bar{\boldsymbol{x}} = \boldsymbol{O}\boldsymbol{x}$ and

$\boldsymbol{x} = \boldsymbol{O}\bar{\boldsymbol{x}}$. Similarly, the stiffness matrix $\boldsymbol{A}$ and the right hand side $\boldsymbol{b}$ in the new basis are given as $\overline{\boldsymbol{A}} = \boldsymbol{O}\boldsymbol{A}\boldsymbol{O}$ and $\bar{\boldsymbol{b}} = \boldsymbol{O}\boldsymbol{b}$. The constraint matrix can be written in the new basis as $\overline{\boldsymbol{B}} = \boldsymbol{B}\boldsymbol{O}$, the matrix $\boldsymbol{B}$ is obtained by discretization of (7) and the matrix $\overline{\boldsymbol{B}}$ is constructed by discretization of (12). The algebraic formulation of the contact problem (16) in the new basis system is given as following minimization problem:

$$\begin{aligned} &\min_{\bar{\boldsymbol{x}}\in\mathbb{R}^{nd}} J(\bar{\boldsymbol{x}}) = \frac{1}{2}\bar{\boldsymbol{x}}^{\mathsf{T}}\overline{\boldsymbol{A}}\bar{\boldsymbol{x}} - \bar{\boldsymbol{x}}^{\mathsf{T}}\bar{\boldsymbol{b}} \\ &\text{subject to} \quad \overline{\boldsymbol{B}}\bar{\boldsymbol{x}} \leqslant \boldsymbol{g}. \end{aligned} \tag{17}$$

Remark. The local basis transformation for this problem can be carried out directly during the assembly process. We can compute the stiffness matrix, constraint matrix and the right-hand side with locally transformed basis by directly utilizing the new basis $\boldsymbol{e}_i$ instead of the Euclidean basis $\boldsymbol{E}_i$ in (14) and (15).

3. A generalized multigrid method

In this section, we introduce a new generalized multigrid method for solving a quadratic minimization problem with linear constraints (17). This multigrid method is motivated by the monotone multigrid method [46,53], which was originally developed to solve a quadratic minimization problem with point-wise inequality constraints. Here, we present an extension of this method for solving a quadratic minimization problem with linear inequality constraints.

The monotone multigrid method is an iterative method, where within each iteration the energy functional is minimized successively such that the current iterate satisfies the constraints. This task is carried out using the projected Gauss–Seidel (PGS) method, which simultaneously minimizes the energy functional and projects the current iterate onto a feasible set. The traditional PGS method is unable to tackle the linearly constrained minimization problem, which represents a linear combination of several variables. To overcome this difficulty, we introduce an orthogonal transformation and a variant of the PGS method that can handle the linear inequality constraints. In addition, in this multigrid method, we employ the transfer operators constructed using the pseudo-L^2-projections. Here, we introduce the necessary ingredients used in our generalized multigrid method. The multigrid method is an ideal iterative method for solving many large-scale linear systems of equations that arise from the discretization of elliptic differential equations [54]. This method obtains optimal convergence rates by exploiting discretizations with different mesh sizes. The multigrid method is considered to have optimal complexity as its convergence rate is bounded from above and it does not depend on the size of the problem. Even though the convergence rate of the multigrid method does not depend on the problem size, the number of arithmetic operations grows proportionally with the problem size. Hence, the complexity of the multigrid method is given as $\mathcal{O}(n)$. The robustness of the multigrid method depends on a sophisticated combination of smoothing iterations and coarse-level corrections. These components are complementary to each other and reduce the error in a different part of the spectrum. We define levels as $\ell \in \{0, \ldots, L\}$, where $\ell = 0$ denote the coarsest level and $\ell = L$ denotes the finest level. The standard multigrid method to solve a linear system $\boldsymbol{A}_L\boldsymbol{x}_L = \boldsymbol{b}_L$ is described in Algorithm 1, where ν_1, ν_2 are the number of pre-smoothing and post-smoothing steps, respectively. The values of $\gamma = 1$ and $\gamma = 2$, in the multigrid algorithm transform the method to a $V(\nu_1, \nu_2)$-cycle and $W(\nu_1, \nu_2)$-cycle, respectively. The matrix $\boldsymbol{T}^{\ell}_{\ell-1}$ denotes a prolongation operator, and its adjoint denotes a restriction operator.

3.1. Transfer operators for unfitted FEM

It is well-known that the efficiency of the multigrid method depends heavily on the underlying hierarchy of meshes and FE spaces. In the multigrid method, the multilevel decomposition of the FE space is performed in such a way that the FE spaces associated with coarser levels are subspaces of the FE space associated with the finest level. This does not necessarily hold for a hierarchy of FE spaces in the unfitted FE framework. Here, we briefly present a strategy for constructing a nested hierarchy of FE spaces from a hierarchy of non-nested meshes.

We define a sequence of quadrilateral meshes of polygonal domain $\widetilde{\Omega}$, denoted as $\{\widetilde{\mathcal{T}}_\ell\}_{\ell=0,\ldots,L}$. We associate the original background mesh on which the problem is defined as the mesh on the finest level, give as, $\widetilde{\mathcal{T}}_L := \widetilde{\mathcal{T}}_h$. The sequence of meshes is created by choosing a mesh on the coarsest level $\widetilde{\mathcal{T}}_0$ and uniformly refining this mesh. Now, we can associate FE spaces $\widetilde{\mathcal{V}}_\ell$ to the meshes on each level, in the same way as given in (3). If the meshes $\{\widetilde{\mathcal{T}}_\ell\}_{\ell=0,\ldots,L}$ are nested then the associate FE spaces are also subspaces of the FE space on finest level, given as

Algorithm 1: Standard multigrid cycle

Input : $(\boldsymbol{A}_\ell)_{\ell=0,\dots,L}, \boldsymbol{b}_L, \boldsymbol{x}_L, L, \nu_1, \nu_2, (\boldsymbol{T}^\ell_{\ell-1})_{\ell=1,\dots,L}, \gamma$
Output: $\boldsymbol{x}_L$

Function: MG($\boldsymbol{A}_\ell, \boldsymbol{b}_\ell, \boldsymbol{x}_\ell, \ell, \nu_1, \nu_2, \boldsymbol{T}^\ell_{\ell-1}, \gamma$)
if $\ell \neq 0$ **then**
  $\boldsymbol{x}_\ell \leftarrow \boldsymbol{x}_\ell +$ Smoother($\boldsymbol{A}_\ell, \boldsymbol{b}_\ell, \boldsymbol{x}_\ell, \nu_1$); ▷ pre-smoothing
  $\boldsymbol{r}_{\ell-1} \leftarrow (\boldsymbol{T}^\ell_{\ell-1})^\mathsf{T}(\boldsymbol{b}_\ell - \boldsymbol{A}_\ell \boldsymbol{x}_\ell)$; ▷ restriction
  $\boldsymbol{A}_{\ell-1} \leftarrow (\boldsymbol{T}^\ell_{\ell-1})^\mathsf{T} \boldsymbol{A}_\ell \boldsymbol{T}^\ell_{\ell-1}$; ▷ Galerkin projection
  $\boldsymbol{c}_{\ell-1} \leftarrow \boldsymbol{0}$; ▷ initialize coarse level correction
  for $i = 1, \dots, \gamma$ **do**
    $\boldsymbol{c}_{\ell-1} \leftarrow \boldsymbol{c}_{\ell-1} +$ MG($\boldsymbol{A}_{\ell-1}, \boldsymbol{r}_{\ell-1}, \ell-1, \nu_1, \nu_2, \boldsymbol{T}^{\ell-1}_{\ell-2}, \gamma$); ▷ coarse level cycle
  $\boldsymbol{x}_\ell \leftarrow \boldsymbol{x}_\ell + \boldsymbol{T}^\ell_{\ell-1} \boldsymbol{c}_{\ell-1}$; ▷ prolongation
  $\boldsymbol{x}_\ell \leftarrow \boldsymbol{x}_\ell +$ Smoother($\boldsymbol{A}_\ell, \boldsymbol{b}_\ell, \boldsymbol{x}_\ell, \nu_2$); ▷ post-smoothing
else
  $\boldsymbol{c}_0 \leftarrow \boldsymbol{A}_0^{-1} \boldsymbol{r}_0$; ▷ direct solver

$\widetilde{\mathcal{V}}_{\ell-1} \subset \widetilde{\mathcal{V}}_\ell$, for all $\ell \in \{1, \dots, L\}$. In order to create a hierarchy of meshes for unfitted FEM, the background meshes are enriched, decomposed and they are associated with either of the domains.

In Fig. 3, we can see that even though the background meshes are nested, the active meshes are not necessarily nested. The nestedness of the active meshes depends heavily on the embedded interfaces. Now, utilizing the characteristic function (4), we restrict the support of the FE spaces $\widetilde{\mathcal{V}}_\ell$ to the domains, given as $\mathcal{V}^1_\ell$ and $\mathcal{V}^2_\ell$, respectively. Hence, enriched FE spaces associated with the domain Ω are also not nested, i.e., $\mathcal{V}^i_{\ell-1} \not\subset \mathcal{V}^i_\ell$, for $i \in \{1, 2\}$ and $\ell \in \{1, \dots, L\}$. To create a hierarchy of nested FE spaces from the hierarchy of non-nested meshes, we adopt the variational transfer approach introduced for the unfitted FEM [39,55]. We define a prolongation operator which projects quantities from a FE space associated with a coarse level to a FE space associated with a fine level, thus as

$$\boldsymbol{\Pi}^\ell_{\ell-1,i} : \mathcal{V}^i_{\ell-1} \to \mathcal{V}^i_\ell, \quad \forall\ \ell \in \{1, \dots, L\},\ i \in \{1, 2\},$$

such that $\boldsymbol{\Pi}^\ell_{\ell-1,i} \mathcal{V}^i_{\ell-1} \subset \mathcal{V}^i_\ell$. By employing this prolongation operator, a FE space associated with an active mesh $\mathcal{T}^i_\ell$ is constructed by the composition of a sequence of prolongation operators,

$$\mathcal{X}^i_\ell := \boldsymbol{\Pi}^L_{L-1,i} \cdots \boldsymbol{\Pi}^{\ell+1}_{\ell,i} \mathcal{V}^i_\ell, \quad \forall\ \ell \in \{1, \dots, L-1\},\ i \in \{1, 2\}.$$

We borrow the definition of the FE space on the finest level as $\mathcal{X}^i_L := \mathcal{V}^i_L$. The nested FE spaces are given as

$$\mathcal{X}^i_0 \subset \mathcal{X}^i_1 \subset \cdots \subset \mathcal{X}^i_{\ell-1} \subset \mathcal{X}^i_\ell \subset \mathcal{X}^i_{\ell+1} \subset \cdots \subset \mathcal{X}^i_{L-1} \subset \mathcal{X}^i_L, \quad \text{for } i \in \{1, 2\}.$$

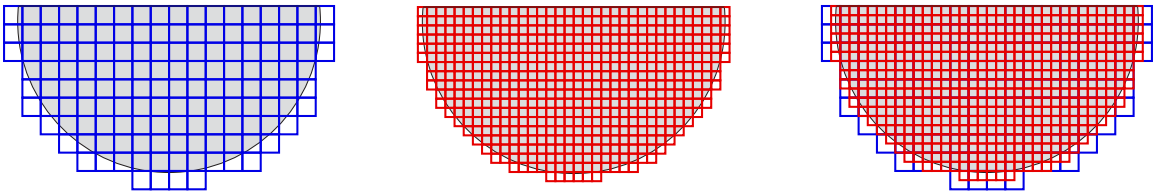
Following the previous section, we can construct the prolongation operator for the domain Ω as a direct sum of the prolongation operators on each domain, i.e., $\boldsymbol{\Pi}^\ell_{\ell-1} := \boldsymbol{\Pi}^\ell_{\ell-1,1} \oplus \boldsymbol{\Pi}^\ell_{\ell-1,2}$, for all $\ell \in \{1, \dots, L\}$. Thus, we can create enriched FE spaces $\mathcal{X}_\ell := \mathcal{X}^1_\ell \oplus \mathcal{X}^2_\ell$ associated with each level ℓ. In addition, we can create a hierarchy of nested FE spaces $\{\mathcal{X}_\ell\}_{\ell=0,\dots,L}$ for the domain Ω by using the prolongation operator $\boldsymbol{\Pi}^\ell_{\ell-1}$.

We compute the algebraic representation $\boldsymbol{T}^\ell_{\ell-1}$ of the prolongation operator $\boldsymbol{\Pi}^\ell_{\ell-1}$ using the pseudo-L^2-projection [39,56]. We define the projection operator $\boldsymbol{\Pi}^\ell_{\ell-1} : \mathcal{V}_{\ell-1} \to \mathcal{V}_\ell$, given by

$$\int_\Omega \boldsymbol{\Pi}^\ell_{\ell-1} \boldsymbol{v}_{\ell-1} \boldsymbol{\mu}_\ell \, d\Omega = \int_\Omega \boldsymbol{v}_{\ell-1} \boldsymbol{\mu}_\ell \, d\Omega \qquad \forall \boldsymbol{\mu}_\ell \in \mathcal{W}_\ell, \tag{18}$$

where $\mathcal{W}_\ell$ is a space of Lagrange multipliers. To construct the algebraic form of the transfer operator, we enforce weak equality condition (18) in a discrete framework, using $\boldsymbol{w}_\ell = \boldsymbol{\Pi}^\ell_{\ell-1} \boldsymbol{v}_{\ell-1}$. We rewrite the weak equality condition as

$$\int_\Omega \boldsymbol{w}_\ell \boldsymbol{\mu}_\ell \, d\Omega = \int_\Omega \boldsymbol{v}_{\ell-1} \boldsymbol{\mu}_\ell \, d\Omega \quad \forall \boldsymbol{\mu} \in \mathcal{W}_\ell.$$



(a) Quadrilateral mesh $\mathcal{T}^i_{\ell-1}$. (b) Quadrilateral mesh $\mathcal{T}^i_{\ell}$. (c) Superimposition of $\mathcal{T}^i_{\ell}$ on $\mathcal{T}^i_{\ell-1}$.

Fig. 3. 2D Quadrilateral meshes on different levels encapsulating the domain Ω^i, (domain Ω^i is shaded in gray).

We define basis of FE spaces $\boldsymbol{\mathcal{V}}_\ell$ and $\boldsymbol{\mathcal{V}}_{\ell-1}$ as $\{\boldsymbol{\phi}^i_\ell\}_{i\in\mathcal{N}_\ell}$ and $\{\boldsymbol{\phi}^i_{\ell-1}\}_{i\in\mathcal{N}_{\ell-1}}$, respectively. The basis of multiplier space $\boldsymbol{\mathcal{W}}_\ell$ is given as $\{\theta^k_\ell\}_{k\in\mathcal{N}_\mu}$. The set of nodes of the FE spaces $\boldsymbol{\mathcal{V}}_{\ell-1}$, $\boldsymbol{\mathcal{V}}_\ell$ and $\boldsymbol{\mathcal{W}}_\ell$ are defined as $\mathcal{N}_{\ell-1}$, $\mathcal{N}_\ell$ and $\mathcal{N}_\mu$ respectively. Now, we introduce the definition of the basis functions of the respective FE space in above equation

$$\sum_{i\in\mathcal{N}_\ell} \boldsymbol{w}^i_\ell \int_\Omega \boldsymbol{\phi}^i_\ell \theta^k_\ell \, d\Omega = \sum_{j\in\mathcal{N}_{\ell-1}} \boldsymbol{v}^j_{\ell-1} \int_\Omega \boldsymbol{\phi}^j_{\ell-1}\theta^k_\ell \, d\Omega \qquad \forall k \in \mathcal{N}_\mu. \tag{19}$$

The formulation (19) in matrix–vector form is given as $\boldsymbol{M}\boldsymbol{w}_\ell = \boldsymbol{N}\boldsymbol{v}_{\ell-1}$. Here, matrix $\boldsymbol{M}$ is a square matrix which is defined on a fine level with entries $M_{ik} = \int_\Omega \boldsymbol{\phi}^i_\ell \theta^k_\ell \, d\Omega$. While, the matrix $\boldsymbol{N}$ is a rectangular matrix, defined between a fine and a coarse space, with entries $N_{jk} = \int_\Omega \boldsymbol{\phi}^j_{\ell-1}\theta^k_\ell \, d\Omega$. Hence, the formula to compute the discrete transfer operator $\boldsymbol{T}^\ell_{\ell-1}$ can be expressed as

$$\boldsymbol{w}_\ell = \boldsymbol{M}^{-1}\boldsymbol{N}\boldsymbol{v}_{\ell-1} = \boldsymbol{T}^\ell_{\ell-1}\boldsymbol{v}_{\ell-1}. \tag{20}$$

If the multiplier space $\boldsymbol{\mathcal{W}}_h$ is considered to be the same as the FE space $\boldsymbol{\mathcal{V}}_\ell$, it gives rise to a mass matrix $M_{ik} = \int_\Omega \boldsymbol{\phi}^i_\ell \boldsymbol{\phi}^k_\ell \, d\Omega$. In this case, we would need to invert a matrix that is of the same size as the stiffness matrix. Additionally, the inverse of such matrix $\boldsymbol{M}$ would be dense, and using such mass matrix would result in a dense transfer operator.

In order to avoid inverting the mass matrix $\boldsymbol{M}$ and to create a sparse representation of the transfer operator, we employ a different definition of the multiplier space [56–58]. To this end, we choose discontinuous piecewise linear or bilinear basis functions ψ_ℓ such that $\boldsymbol{\mathcal{W}}_\ell := \text{span}\{\psi^i_\ell\}_{i\in\mathcal{N}_\mu}$. The basis functions ψ_ℓ are defined to be biorthogonal to the standard Lagrange basis with respect to L^2-inner product, given by

$$\int_\Omega \boldsymbol{\phi}^p_\ell \psi^q_\ell \, d\Omega = \delta_{pq} \int_\Omega \boldsymbol{\phi}^p_\ell \, d\Omega \qquad \forall p \in \mathcal{N}_\ell, \; q \in \mathcal{N}_\mu, \tag{21}$$

where δ_{pq} denotes the Kronecker delta. The transfer operator constructed using this definition of the multiplier space is termed as a pseudo-L^2-projection operator. Now, we can compute the entries of the matrix $\boldsymbol{M}$ and matrix $\boldsymbol{N}$ are computed as $M_{ik} = \delta_{ik}\int_\Omega \boldsymbol{\phi}^i_\ell \, d\Omega$ and $N_{jk} = \int_\Omega \boldsymbol{\phi}^j_{\ell-1}\psi^k_\ell \, d\Omega$, respectively. As we can notice, due to the definition of the biorthogonal basis the matrix $\boldsymbol{M}$ becomes a diagonal matrix, which is trivial to invert. Also the inverse of the matrix $\boldsymbol{M}$ is a sparse matrix and the transfer operator $\boldsymbol{T}^\ell_{\ell-1}$ also has a sparse structure. We employ transfer operators computed using the pseudo-L^2-projection in our multigrid method.

Remark. In practice, we need to compute the pseudo-L^2-projection only on cut-elements. As for the elements that are not intersected by the interfaces, we can employ the standard interpolation operator as the meshes are nested in the interior of the domain. This can be efficiently carried out by constructing interpolation operators associated with each subdomain and then replacing the entries (in the operator) associated with the cut-elements by appropriate entries of the pseudo-L^2-projection operator.

3.2. Orthogonal transformation

In this section, we introduce the orthogonal transformation for the contact problem (17). This transformation is necessary to decouple the linear constraints, which in turn allows us to utilize the modified PGS method. In order to decouple the constraints, we perform a QR decomposition of the constraint matrix $\overline{\boldsymbol{B}}^\mathsf{T}$

$$\overline{\boldsymbol{B}}^\mathsf{T} = \boldsymbol{Q}\boldsymbol{R} \quad \text{and} \quad \overline{\boldsymbol{B}} = \boldsymbol{R}^\mathsf{T}\boldsymbol{Q}^\mathsf{T},$$

where $\boldsymbol{Q} \in \mathbb{R}^{nd\times nd}$ is an orthonormal matrix. Thus, we have $\boldsymbol{Q}\boldsymbol{Q}^{\mathsf{T}} = \boldsymbol{Q}^{\mathsf{T}}\boldsymbol{Q} = \boldsymbol{I}$, where $\boldsymbol{I} \in \mathbb{R}^{nd\times nd}$, represents the identity matrix. The decomposition of the matrix $\boldsymbol{R} \in \mathbb{R}^{nd\times m}$ is given by $\boldsymbol{R} = [\boldsymbol{R}_1 \ \boldsymbol{O}_1]^{\mathsf{T}}$, where $\boldsymbol{R}_1 \in \mathbb{R}^{m\times m}$ is an upper triangular matrix and $\boldsymbol{O}_1 \in \mathbb{R}^{(nd-m)\times m}$ is a matrix with all zero entries. The matrix $\boldsymbol{Q}$ simply provides a change of basis, and on this new basis system, the representation of the constraint is modified. It is clear from the structure of the new constraint matrix $\boldsymbol{R}_1$ that in the modified basis system the constraints are sequentially dependent on the previous linear constraint. The matrix $\boldsymbol{Q}$ is used to define the variables in the new basis system, given as $\widehat{\boldsymbol{x}} = \boldsymbol{Q}^{\mathsf{T}}\bar{\boldsymbol{x}}$ and $\bar{\boldsymbol{x}} = \boldsymbol{Q}\widehat{\boldsymbol{x}}$. Moreover, we can observe that $\boldsymbol{Q}^{\mathsf{T}}\overline{\boldsymbol{B}}^{\mathsf{T}} = \boldsymbol{R}$ and $\overline{\boldsymbol{B}}\,\boldsymbol{Q} = \boldsymbol{R}^{\mathsf{T}}$. By incorporating the transformed matrices and vectors, we can reformulate the constrained minimization problem (17) as follows:

$$\begin{aligned} &\min_{\widehat{\boldsymbol{x}}\in\mathbb{R}^{nd}} J(\widehat{\boldsymbol{x}}) = \frac{1}{2}\widehat{\boldsymbol{x}}^{\mathsf{T}}\widehat{\boldsymbol{A}}\widehat{\boldsymbol{x}} - \widehat{\boldsymbol{x}}^{\mathsf{T}}\widehat{\boldsymbol{b}} \\ &\text{subject to} \quad \boldsymbol{R}^{\mathsf{T}}\widehat{\boldsymbol{x}} \leqslant \boldsymbol{g}, \end{aligned} \tag{22}$$

where $\widehat{\boldsymbol{A}} = \boldsymbol{Q}^{\mathsf{T}}\overline{\boldsymbol{A}}\,\boldsymbol{Q}$ and $\widehat{\boldsymbol{b}} = \boldsymbol{Q}^{\mathsf{T}}\bar{\boldsymbol{b}}$. The constraints of the above optimization problem can be written algebraically as,

$$\begin{pmatrix} \boldsymbol{R}_{11} & 0 & 0 & \cdots & 0 & 0 & \cdots & 0 \\ \boldsymbol{R}_{11} & \boldsymbol{R}_{22} & 0 & \cdots & 0 & 0 & \cdots & 0 \\ \boldsymbol{R}_{13} & \boldsymbol{R}_{23} & \boldsymbol{R}_{33} & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & & \ddots & \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{R}_{1m} & \boldsymbol{R}_{2m} & \boldsymbol{R}_{3m} & \cdots & \boldsymbol{R}_{mm} & 0 & \cdots & 0 \end{pmatrix} \begin{pmatrix} \widehat{\boldsymbol{x}}_1 \\ \widehat{\boldsymbol{x}}_2 \\ \widehat{\boldsymbol{x}}_3 \\ \vdots \\ \widehat{\boldsymbol{x}}_m \\ \widehat{\boldsymbol{x}}_{m+1} \\ \vdots \\ \widehat{\boldsymbol{x}}_{nd} \end{pmatrix} \leqslant \begin{pmatrix} \boldsymbol{g}_1 \\ \boldsymbol{g}_2 \\ \boldsymbol{g}_3 \\ \vdots \\ \boldsymbol{g}_m \end{pmatrix}. \tag{23}$$

As $\boldsymbol{Q}$ is an orthonormal matrix, the spectral properties of the $\widehat{\boldsymbol{A}}$ and $\boldsymbol{A}$ are equivalent. But, the sparsity pattern of the original matrix $\overline{\boldsymbol{A}}$ and its rotated variant $\widehat{\boldsymbol{A}}$ are quite different. In practice, the matrix $\widehat{\boldsymbol{A}}$ is denser than the original matrix, which in turn increases the computational cost of the matrix–vector products in the algorithm. The new constraint matrix $\boldsymbol{R}^{\mathsf{T}}$ has a lower triangular structure, which can be handled easily by forward substitution. It is important to note that, this type of constraint can be handled easily by the PGS method, due to its inherent sequential nature.

Now, we define a constrained subspace or a feasible set as

$$\widehat{\mathbb{K}} = \{\widehat{\boldsymbol{x}} \in \mathbb{R}^{nd} : \boldsymbol{R}^{\mathsf{T}}\widehat{\boldsymbol{x}} \leqslant \boldsymbol{g}\}.$$

We pose our problem as an energy minimization problem in the following algebraic formulation:

$$\text{find } \widehat{\boldsymbol{x}} \in \widehat{\mathbb{K}} \text{ such that} \qquad J(\widehat{\boldsymbol{x}}) \leqslant J(\widehat{\boldsymbol{y}}) \qquad \forall \widehat{\boldsymbol{y}} \in \widehat{\mathbb{K}}. \tag{24}$$

3.3. Modified projected Gauss–Seidel method

Here, we introduce a modified PGS method for solving the problem (24). The Gauss–Seidel method minimizes the energy functional $J(\cdot)$ in each local iteration step. The energy minimization takes place in the direction of the nodal basis functions that span the FE space. The Gauss–Seidel method can be written as a subspace correction method, where the subspace decomposition is achieved by a direct splitting of the underlying FE space into one-dimensional subspaces spanned by the nodal basis functions. The PGS method is used widely to solve various forms of obstacle problems, and it is globally convergent [40,46]. We remark that decoupling of the constraints with respect to the nodal basis function is essential for the global convergence of the PGS method [46,59]. The original linear contact condition, $\overline{\boldsymbol{B}}\bar{\boldsymbol{x}} \leqslant \boldsymbol{g}$, does not satisfy this property, as the constraints are represented by the linear combination of basis functions. The QR decomposition allows us to decouple the constraints by expressing them in new basis as $\boldsymbol{R}^{\mathsf{T}}\widehat{\boldsymbol{x}} \leqslant \boldsymbol{g}$. In order to discuss this method in generic way, we introduce abstract upper bound $\boldsymbol{ub} \in \mathbb{R}^m$ and lower bound $\boldsymbol{lb} \in \mathbb{R}^m$. In the context of the contact problem, the lower bound and upper bound are defined as $\boldsymbol{lb} = \{-\infty\}$ and $\boldsymbol{ub} = \boldsymbol{g}$, respectively. In addition, we define an active set as a set of all DoFs where the constraints are binding, thus as

$$\mathscr{A} := \{p : (\boldsymbol{R}^{\mathsf{T}}\widehat{\boldsymbol{x}})_p = \boldsymbol{g}_p\}.$$

Algorithm 2: Modified Projected Gauss-Seidel method

Input : $\widehat{\boldsymbol{A}}, \widehat{\boldsymbol{b}}, \boldsymbol{R}, \widehat{\boldsymbol{x}}^{(0)}, \boldsymbol{lb}, \boldsymbol{ub}, \nu_*$
Output: $\widehat{\boldsymbol{x}}^{(\nu_*)}, \mathscr{A}$

Function: Projected GS($\widehat{\boldsymbol{A}}, \widehat{\boldsymbol{b}}, \boldsymbol{R}, \widehat{\boldsymbol{x}}^{(0)}, \boldsymbol{lb}, \boldsymbol{ub}, \nu_*$)
for $k = 1, 2, \ldots, \nu_*$ **do**
  $\mathscr{A} \leftarrow \emptyset$; ▷ initialize empty active set
  for $i = 1, 2, \ldots, n$ **do**
    $\widehat{\boldsymbol{x}}_i^{(k)} = \frac{1}{\widehat{\boldsymbol{A}}_{ii}} (\widehat{\boldsymbol{b}}_i - \sum_{j<i} \widehat{\boldsymbol{A}}_{ij} \widehat{\boldsymbol{x}}_j^{(k)} - \sum_{j>i} \widehat{\boldsymbol{A}}_{ij} \boldsymbol{x}_j^{(k-1)})$; ▷ update the iterate
    if $i \leqslant m$ **then**
      $lb_t = \frac{1}{\boldsymbol{R}_{ii}} (\boldsymbol{lb}_i - \sum_{j=1}^{i-1} \boldsymbol{R}_{ji} \widehat{\boldsymbol{x}}_j^{(k)})$; $ub_t = \frac{1}{\boldsymbol{R}_{ii}} (\boldsymbol{ub}_i - \sum_{j=1}^{i-1} \boldsymbol{R}_{ji} \widehat{\boldsymbol{x}}_j^{(k)})$; ▷ updated local bounds
      if $\widehat{\boldsymbol{x}}_i^{(k)} < lb_t$ ***or*** $ub_t < \widehat{\boldsymbol{x}}_i^{(k)}$ **then**
        $\widehat{\boldsymbol{x}}_i^{(k)} = \max(lb_t, \min(\widehat{\boldsymbol{x}}_i^{(k)}, ub_t))$; ▷ project onto feasible set
        $\mathscr{A} \leftarrow \mathscr{A} \cup \{i\}$; ▷ add current index to the active set

The matrix $\boldsymbol{R}^{\mathsf{T}}$ is a lower triangular matrix, which allows us to write the constraints as a linear combination of the current nodal basis function and previously constrained basis. This key idea allows us to use the PGS method to solve the problem (24).

The iterative process is given as follows. For a given kth iterate $\widehat{\boldsymbol{x}}^{(k)} \in \widehat{\mathbb{K}}$, we compute a sequence of local intermediate iterates, $\boldsymbol{z}^{(0)}, \boldsymbol{z}^{(1)}, \ldots, \boldsymbol{z}^{(nd)}$. We begin with the first local iterate $\boldsymbol{z}^{(0)} := \widehat{\boldsymbol{x}}^{(k)}$, and the next local iterates are given by $\boldsymbol{z}^{(i)} = \boldsymbol{z}^{(i-1)} + \boldsymbol{c}^{(i)}$, for $i = 1, \ldots, nd$. Once all local intermediate iterates are computed, a new global iterate of the Gauss–Seidel step is given by $\widehat{\boldsymbol{x}}^{(k+1)} := \boldsymbol{z}^{(n)}$. The corrections $\boldsymbol{c}^{(i)}$ are obtained as the unique solution of the following local subproblems, given as,

$$\text{find } \boldsymbol{c}^{(i)} \in \mathbb{D}^{(i)} \text{ such that} \quad J(\boldsymbol{z}^{(i-1)} + \boldsymbol{c}^{(i)}) \leqslant J(\boldsymbol{z}^{(i-1)} + \boldsymbol{y}) \qquad \forall \boldsymbol{y} \in \mathbb{D}^{(i)},$$

with closed, convex set $\mathbb{D}^{(i)}$, defined for abstract upper bound $\boldsymbol{ub}$ and lower bound $\boldsymbol{lb}$ as

$$\mathbb{D}^{(i)} = \{\boldsymbol{c}^{(i)} \in \mathbb{R}^n : \boldsymbol{lb} - \boldsymbol{R}^{\mathsf{T}} \boldsymbol{z}^{(i-1)} \leqslant \boldsymbol{R}^{\mathsf{T}} \boldsymbol{c}^{(i)} \leqslant \boldsymbol{ub} - \boldsymbol{R}^{\mathsf{T}} \boldsymbol{z}^{(i-1)}\}. \tag{25}$$

Each intermediate step ensures that the iterate does not violate constraints. If the current iterate violates the constraints, it is projected to the admissible space. The PGS method for a generic linear inequality constrained minimization problem is summarized in Algorithm 2.

Thus, we have a globally convergent PGS method that can be used to solve the problem (24). But the convergence rate of the Gauss–Seidel method deteriorates as the size of the problem increases. Hence, we employ the modified PGS method as a smoother in our multigrid method.

Remark. In Algorithm 2, we have assumed that the values of diagonal entries of the matrix $\boldsymbol{R}_1$ (where $\boldsymbol{R} = [\boldsymbol{R}_1 \ \boldsymbol{O}_1]^{\mathsf{T}}$) are positive, which does not hold in general. It is necessary to pay attention to the sign of diagonal entries of the matrix $\boldsymbol{R}_1$, as the sign of the diagonal values may change the inequality bounds.

3.4. Multigrid method

In this section, we summarize the generalized multigrid method, which includes all the components introduced in the previous section. In particular, we have a sequence of non-nested finite element spaces $\{\boldsymbol{\mathcal{V}}_\ell\}_{\ell=0,\ldots,L}$ associated with the hierarchy of meshes $\{\mathcal{T}_\ell\}_{\ell=0,\ldots,L}$. Following Section 3.1, we have the transfer operators $\{\boldsymbol{\Pi}_{\ell-1}^{\ell}\}_{\ell=1,\ldots,L}$ which are computed using the pseudo-L^2-projections. By means of these transfer operators, we create a hierarchy of nested finite element spaces $\{\boldsymbol{\mathcal{X}}_\ell\}_{\ell=0,\ldots,L}$ from the hierarchy of background meshes. The prolongation matrices associated with the transfer operators are given as $\{\boldsymbol{T}_{\ell-1}^{\ell}\}_{\ell=1,\ldots,L}$.

The orthogonal transformation of the matrix $\overline{\boldsymbol{B}}^{\mathsf{T}}$ plays a vital role in our multigrid method. We recall, on the finest level the definition of the FE space is kept the same, as $\boldsymbol{\mathcal{X}}_L = \boldsymbol{\mathcal{V}}_L$, and hence also the nodal basis functions

Algorithm 3: Generalized Multigrid algorithm

Input : $\boldsymbol{A}_L, \boldsymbol{b}_L, \boldsymbol{x}_L, L, \nu_1, \nu_2, (\boldsymbol{T}^{\ell}_{\ell-1})_{\ell=1,\dots,L}, \boldsymbol{B}, \boldsymbol{O}, \boldsymbol{lb}, \boldsymbol{ub}, \gamma$
Output: $\boldsymbol{x}_L \leftarrow \boldsymbol{O}\bar{\boldsymbol{x}}_L$,$\bar{\boldsymbol{x}}_L \leftarrow \boldsymbol{Q}\widehat{\boldsymbol{x}}_L$

Function: GMG($\boldsymbol{A}_L, \boldsymbol{b}_L, \boldsymbol{x}_L, L, \nu_1, \nu_2, (\boldsymbol{T}^{\ell}_{\ell-1})_{\ell=1,\dots,L}, \boldsymbol{B}, \boldsymbol{O}, \boldsymbol{lb}, \boldsymbol{ub}, \gamma$)
$\overline{\boldsymbol{T}}^{L}_{L-1} \leftarrow \boldsymbol{O}\boldsymbol{T}^{L}_{L-1}$; $\overline{\boldsymbol{A}}_L \leftarrow \boldsymbol{O}\boldsymbol{A}_L\boldsymbol{O}$; $\bar{\boldsymbol{b}}_L \leftarrow \boldsymbol{O}\boldsymbol{b}_L$; $\bar{\boldsymbol{x}}_L \leftarrow \boldsymbol{O}\boldsymbol{x}_L$; $\overline{\boldsymbol{B}} \leftarrow \boldsymbol{B}\boldsymbol{O}$; ▷ local basis transformation
$\boldsymbol{Q}, \boldsymbol{R} \leftarrow$ QR Transformation($\overline{\boldsymbol{B}}^{\mathsf{T}}$); ▷ QR decomposition
$\widehat{\boldsymbol{T}}^{L}_{L-1} \leftarrow \boldsymbol{Q}^{\mathsf{T}}\overline{\boldsymbol{T}}^{L}_{L-1}$; $\widehat{\boldsymbol{A}}_L \leftarrow \boldsymbol{Q}^{\mathsf{T}}\overline{\boldsymbol{A}}_L\boldsymbol{Q}$; $\widehat{\boldsymbol{b}}_L \leftarrow \boldsymbol{Q}^{\mathsf{T}}\bar{\boldsymbol{b}}_L$; $\widehat{\boldsymbol{x}}_L \leftarrow \boldsymbol{Q}^{\mathsf{T}}\bar{\boldsymbol{x}}_L$; ▷ orthogonal rotation
while *not converged* **do**
  $\widehat{\boldsymbol{x}}_L, \mathscr{A}_L \leftarrow \widehat{\boldsymbol{x}}_L +$ Projected GS($\widehat{\boldsymbol{A}}_L, \widehat{\boldsymbol{b}}_L, \boldsymbol{R}^{\mathsf{T}}, \widehat{\boldsymbol{x}}_L, \boldsymbol{lb}, \boldsymbol{ub}, \nu_1$); ▷ ν_1 pre-smoothing steps
  $\widehat{\boldsymbol{r}}_L \leftarrow \widehat{\boldsymbol{b}}_L - \widehat{\boldsymbol{A}}_L\widehat{\boldsymbol{x}}_L$; ▷ residual
  $\widehat{\boldsymbol{r}}_{trc} \leftarrow$ trc($\widehat{\boldsymbol{r}}_L, \mathscr{A}_L$); $\widehat{\boldsymbol{A}}_{trc} \leftarrow$ trc($\widehat{\boldsymbol{A}}_L, \mathscr{A}_L$); ▷ truncation
  $\boldsymbol{r}_{L-1} \leftarrow (\widehat{\boldsymbol{T}}^{L}_{L-1})^{\mathsf{T}}\widehat{\boldsymbol{r}}_{trc}$; ▷ restriction
  $\boldsymbol{A}_{L-1} \leftarrow (\widehat{\boldsymbol{T}}^{L}_{L-1})^{\mathsf{T}}\widehat{\boldsymbol{A}}_{trc}\widehat{\boldsymbol{T}}^{L}_{L-1}$; ▷ Galerkin projection
  $\boldsymbol{c}_{L-1} \leftarrow \boldsymbol{0}$; ▷ initialize coarse level correction
  for $i = 1, \dots, \gamma$ **do**
    $\boldsymbol{c}_{L-1} \leftarrow \boldsymbol{c}_{L-1} +$ MG($\boldsymbol{A}_{L-1}, \boldsymbol{r}_{L-1}, L-1, \nu_1, \nu_2, \boldsymbol{T}^{L-1}_{L-2}, \gamma$); ▷ coarse level cycle
  $\widehat{\boldsymbol{c}}_L \leftarrow \widehat{\boldsymbol{T}}^{L}_{L-1}\boldsymbol{c}_{L-1}$; ▷ prolongation
  $\widehat{\boldsymbol{c}}_{\text{trc}} \leftarrow$ trc($\widehat{\boldsymbol{c}}_L, \mathscr{A}_L$); ▷ truncation
  $\widehat{\boldsymbol{x}}_L \leftarrow \widehat{\boldsymbol{x}}_L + \widehat{\boldsymbol{c}}_{\text{trc}}$; ▷ update iterate
  $\widehat{\boldsymbol{x}}_L, \mathscr{A}_L \leftarrow \widehat{\boldsymbol{x}}_L +$ Projected GS($\widehat{\boldsymbol{A}}_L, \widehat{\boldsymbol{b}}_L, \boldsymbol{R}^{\mathsf{T}}, \widehat{\boldsymbol{x}}_L, \boldsymbol{lb}, \boldsymbol{ub}, \nu_2$); ▷ ν_2 post-smoothing steps

defined on these FE spaces are given as $\zeta^p_L = \phi^p_L$, for all $p \in \mathscr{N}_L$. These nodal basis functions are modified or rotated after the orthogonal transformation, which can be written as

$$\widehat{\zeta}^q_L \boldsymbol{e}_i(q) := \sum_{p \in \mathscr{N}_L} \boldsymbol{Q}_{pq}\zeta^p_L \boldsymbol{e}_i(p) \qquad \forall q \in \mathscr{N}_L.$$

The transfer operators are computed using the nodal basis functions that span the FE space on a coarse level and a fine level. With the modified nodal basis functions on the finest level, it becomes essential to compute the transfer operator associated with the finest level such that the vector and the matrix quantities are projected on a FE space spanned by the modified basis system. Thus, the prolongation matrix $\boldsymbol{T}^{L}_{L-1}$ is also modified in two stages. The first stage is necessary because of the local basis transformation, which is carried out locally to modify the basis system by means of the Householder rotation matrix $\boldsymbol{O}$ such that contact conditions are only applicable in the normal direction. The second transformation is carried out using the orthogonal transformation matrix $\boldsymbol{Q}$, which is computed to decouple the linear contact constraints. The updated transfer operator is defined as

$$\widehat{\boldsymbol{T}}^{L}_{L-1} = \boldsymbol{Q}^{\mathsf{T}}\overline{\boldsymbol{T}}^{L}_{L-1} = \boldsymbol{Q}^{\mathsf{T}}\boldsymbol{O}\boldsymbol{T}^{L}_{L-1}.$$

Now, the nodal basis function associated with the FE space $\boldsymbol{\mathcal{X}}_{L-1}$ are given as

$$\widehat{\zeta}^q_{L-1}\boldsymbol{e}_i(q) := \sum_{p \in \mathscr{N}_L} (\widehat{\boldsymbol{T}}^{L}_{L-1})_{pq}\zeta^p_L \boldsymbol{e}_i(p) = \sum_{p \in \mathscr{N}_L} (\boldsymbol{Q}^{\mathsf{T}}\overline{\boldsymbol{T}}^{L}_{L-1})_{pq}\zeta^p_L \boldsymbol{e}_i(p), \qquad \forall q \in \mathscr{N}_{L-1}.$$

This modification of the transfer operator is only required on the finest level, while all other transfer operators on the coarser levels $\{\boldsymbol{T}^{\ell}_{\ell-1}\}_{\ell=0,\dots,L-1}$ remain the same.

The modified PGS method is employed as a smoother in the generalized multigrid method only on the finest level. It minimizes the energy functional in each local iteration in each smoothing step. At the end of the smoothing iterations, we obtain an active set where the constraints are binding. The most crucial feature of this multigrid method is that the coarse level corrections do not violate the fine level constraints. As a consequence, we solve the constrained optimization problem only on the finest level, while on the coarse levels, we solve the unconstrained linear problem. This is also very convenient, as in this algorithm the representation of the contact constraints is only required on the finest level.

To ensure that the coarse level corrections do not violate the constraint on the finest level, we modify the restriction of the residual and the stiffness matrix, and the prolongation of the coarse level correction. Following the discussion in Section 3.1, the nodal basis functions associated with the coarse level FE space are computed as a linear combination of the nodal basis function defined on the FE space on the finest level. If the value of a nodal basis function on the finest level is set to zero, the basis function constructed on the coarse levels is represented by truncated basis functions. For all DoFs that are in the active set, we set the corresponding entries of the residual or the prolongated correction to zero. While for the stiffness matrix, we set the rows and columns associated with the active set to be zero. This is equivalent to removing the nodal basis function associated with all DoFs in the active set.

As we are employing transfer operators constructed by the pseudo-L^2-projection, this multigrid method including the truncation process can be carried out algebraically. In comparison with the standard multigrid method used for solving linear systems, this algorithm is computationally more expensive. This can be attributed to the cost of computing the orthogonal transformation of the matrix $\overline{\boldsymbol{B}}^{\mathsf{T}}$ and then projecting the problem onto a new basis system. Even though the generalized multigrid method is computationally more expensive, it has optimal convergence properties. Additionally, it is significantly cheaper in comparison with the other iterative methods, e.g., interior-point method or semi-smooth Newton method. If we are solving an optimization problem with inequality constraints, the active set changes in a few initial multigrid iterations. However, once the active set of the solution is identified, the algorithm converges linearly.

In Algorithm 3, we can see the detailed generalized multigrid algorithm with the modified PGS method as a smoother. On the coarse levels, we employ the standard MG methods as described in Algorithm 1, with any regular smoothers. Here, we note that Algorithm 3 is given in an abstract setting for inequality constraints with upper bounds and lower bounds, assuming that the active set may change in each multigrid iteration.

4. Numerical results

In this section, we evaluate the performance of the discretization method and the generalized multigrid method for solving Signorini's problem and the two-body contact problem. We utilize Givens rotation to perform the orthogonal transformation of the matrix $\boldsymbol{B}^{\mathsf{T}}$, as this method produces a sparser matrix $\boldsymbol{Q}$ than the other orthogonalization methods. As an exact solution of the considered problem is not available, we choose correction in the energy norm as a termination criterion, given by

$$\|\boldsymbol{x}^{(k+1)} - \boldsymbol{x}^{(k)}\|_A < 10^{-10}. \tag{26}$$

This termination criterion can be interpreted as a preconditioned residual norm and it provides a good estimation for the algebraic error. Such termination criteria have been used as posterior error estimates for the cascadic multigrid methods and the adaptive multigrid methods [60,61]. Also, we define the convergence rate of an iterative solution scheme as

$$\rho^{k+1} := \frac{\|\boldsymbol{x}^{(k+1)} - \boldsymbol{x}^{(k)}\|_A}{\|\boldsymbol{x}^{(k)} - \boldsymbol{x}^{(k-1)}\|_A}.$$

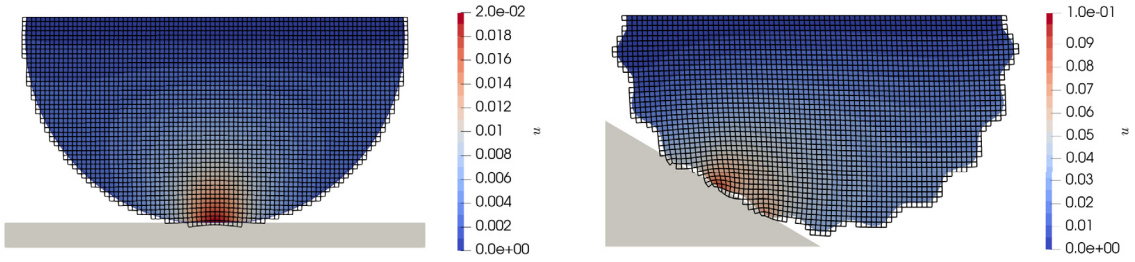
We denote the asymptotic convergence rate as ρ^*, where the iterate $\boldsymbol{x}^{(k+1)}$ satisfies the termination criterion (26). The initial guess $\boldsymbol{x}^{(0)}$ for all solution schemes is chosen as a zero vector.

The implementation and analysis of the discretization schemes and the solution methods are performed using MATLAB R2020a. The experiments have been carried out on a system with an Intel Xeon E5-2650 v3 processor and 32 GB memory.

4.1. Benchmark problems

4.1.1. Signorini's problem

In this section, we describe the problem setup for Signorini's problem for two different types of rigid obstacles. All experiments in this section are carried out on a structured background mesh with the quadrilateral elements. On the coarsest level, the mesh $\widetilde{\mathcal{T}}_0$ is given on a rectangular domain $\widetilde{\Omega} = (-1.09, 1.09) \times (0, 1.09)$, with 100 elements in x-direction and 50 in y-direction, which we denoted as mesh on level $L0$. By uniformly refining the mesh $\widetilde{\mathcal{T}}_0$, we obtain a hierarchy of meshes $\{\widetilde{\mathcal{T}}_\ell\}_{\ell=1,\dots,5}$ associated with levels $L1, L2, \dots, L5$. The Dirichlet



(a) Setup of Signorini's problem for Example 1-SC. (b) Setup of Signorini's problem for Example 2-SC.

Fig. 4. Setup of for experiments: the objects in the gray scale are rigid obstacles. We can see active meshes and the resulting displacement field for Signorini's problem.

boundary condition is defined as $\boldsymbol{u} = (0, 0)$ on $x \in [-1.09, 1.09]$ and $y = 0$. The body force for this example is considered to be zero, i.e., $\boldsymbol{f} = (0, 0)$. In these experiments, the material parameters are chosen as Young's modulus $E = 10\,\text{MPa}$ and Poisson's ratio $\nu = 0.3$. We can compute Lamé parameters λ and μ using the following relation: $\lambda = \frac{E\nu}{(1+\nu)(1-2\nu)}$ and $\mu = \frac{E}{2(1+\nu)}$.

Example 1-SC. We consider a rigid foundation, defined by a line $y = 0.12$. The body Ω is defined in such a way that it violates the non-penetration contact condition in a stress-free configuration. In this experiment, we consider a semicircular domain, where the contact boundary of the domain is defined by the zero-isoline of a level set function $\Lambda_{s_1}(\boldsymbol{X}) := r_{s_1}^2 - \|\boldsymbol{X} - \boldsymbol{c}_1\|_2^2$ with radius $r_{s_1} = 0.9$, and $\boldsymbol{c}_1 = (0, 1)$ denotes the center of the circle. The domain Ω is defined by the region where the value of the level set is positive, $\Lambda_{s_1} > 0$. The setup of this example is depicted in Fig. 4(a), where we see the resulting magnitude of the displacement field due to the contact with a rigid foundation.

Example 2-SC. This example considers a non-symmetric obstacle and possible multiple contact regions. The body Ω is considered as a semicircular domain with wavy boundaries, given as $\Lambda_{s_2}(\boldsymbol{X}) := r^2 - 0.5(\tilde{x}^2 + \tilde{y}^2)(5 + 0.3\sin(\frac{\pi}{36} + 21\arctan(\frac{\tilde{y}}{\tilde{x}})))$ where, $\tilde{x} = \boldsymbol{X}_x - c_y$ and $\tilde{y} = \boldsymbol{X}_y - c_y$. Here, the radius is defined as $r = 1.31111$ and the center is given as $\boldsymbol{c} = (10^{-5}, 1 + 10^{-5})$. The center $\boldsymbol{c}$ is shifted in the x-direction by 10^{-5} in order to avoid the intersection of the zero-isoline of a level set function Λ_{s_2} with the nodes of background meshes. The domain Ω is defined as the region where the level set function Λ_{s_2} has positive values. The rigid foundation is defined as a line which passes through the points $(-1, 0.7)$ and $(0.2, 0)$. In Fig. 4(b), we can observe the setup and magnitude of the displacement field due to contact with the rigid foundation.

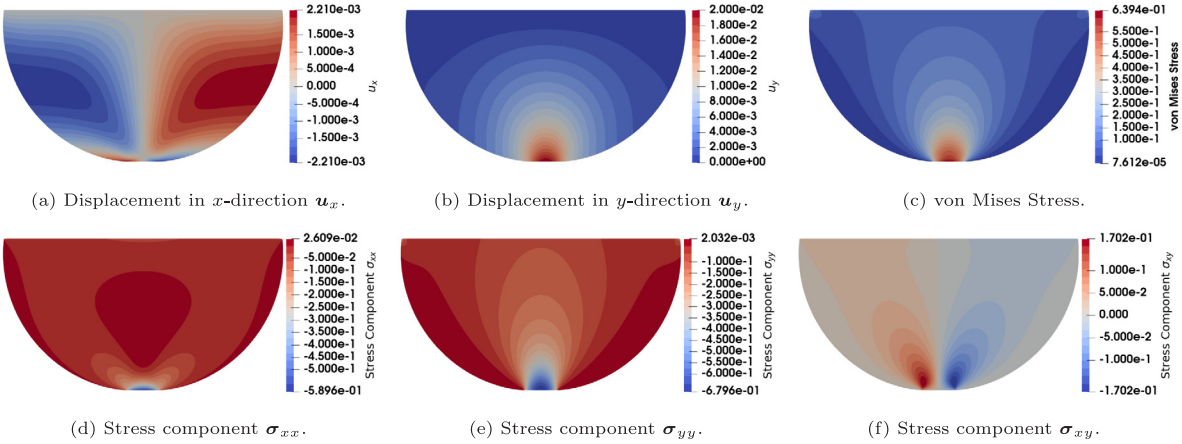
4.1.2. Two-body contact problem

Two bodies are considered to be in contact with each other in the absence of external forces. The mesh $\{\widetilde{\mathcal{T}}_\ell\}_{\ell=0,\dots,5}$ is given in square domain $\widetilde{\Omega} = \Omega = (0, 1)^2$. On the coarsest level, i.e., $L0$, we start with 50 elements in each direction, this mesh is denoted by $\widetilde{\mathcal{T}}_0$. We create a hierarchy of meshes by uniformly refining this mesh until we have 1600 elements in each direction, this mesh is defined as $\widetilde{\mathcal{T}}_5$. The Dirichlet boundary condition is defined as $\boldsymbol{u} = (0, 0)$ on $x \in [0, 1]$ and $y = 0$, while the Neumann boundary condition is defined as $\boldsymbol{\sigma n} = (0, 5)$ on $x \in [0, 1]$ and $y = 1$. The body force for this example is considered to be zero.

Example 1-TC. For this example, we consider a circular contact interface denoted as Γ_c. The circular interface is defined as the zero-isoline of a level set function $\Lambda_c(\boldsymbol{X}) := r_{t_1}^2 - \|\boldsymbol{X} - \boldsymbol{c}_t\|_2^2$, with $r_{t_1}^2 = 3 - 2\cdot 2^{1/2}$, and $\boldsymbol{c}_t$ is the center of the circle, chosen as $(0.5, 0.5)$. The circular interface decomposes the domain Ω into Ω^1, where $\Lambda_c(\boldsymbol{X}) > 0$ and Ω^2 where $\Lambda_c(\boldsymbol{X}) < 0$. For this example, we consider two different sets of material parameters. We choose Young's modulus as $E_1 = 10\,\text{MPa}$ and $E_2 \in \{10\,\text{MPa}, 50\,\text{MPa}\}$ and the Poisson's ratio is chosen as $\nu_1 = \nu_2 = 0.3$.

Example 2-TC. This example considers an elliptical contact interface denoted as Γ_e. The interface is defined as the zero-isoline of level set function

$$\Lambda_e(\boldsymbol{X}) := r_{t_2}^2 - \left|\frac{\boldsymbol{X}_x - c_x}{a}\right|^2 - \left|\frac{\boldsymbol{X}_y - c_y}{b}\right|^2.$$



(a) Displacement in x-direction $\boldsymbol{u}_x$. (b) Displacement in y-direction $\boldsymbol{u}_y$. (c) von Mises Stress.

(d) Stress component $\boldsymbol{\sigma}_{xx}$. (e) Stress component $\boldsymbol{\sigma}_{yy}$. (f) Stress component $\boldsymbol{\sigma}_{xy}$.

Fig. 5. Resulting displacement field and stress field of Example 1-SC.

Here, r_{t_2} denotes the radius of the ellipse, chosen as $r_{t_2}^2 = 2(3-2\cdot 2^{1/2})$. The symbols a and b denote the major and minor axis of the ellipse, chosen as $a = 1, b = 0.8$. Here the center of the circle is chosen as $(0.5, 0.5)$. The elliptical interface decomposes the domain Ω into Ω^1, where $\Lambda_e(\boldsymbol{X}) > 0$ and Ω^2, where $\Lambda_e(\boldsymbol{X}) < 0$. For this example, we consider the same set of material parameters, as used in the previous example. Thus, Young's modulus is chosen as $E_1 = 10\,\text{MPa}$, $E_2 \in \{10\,\text{MPa}, 50\,\text{MPa}\}$ and the Poisson's ratio $\nu_1 = \nu_2 = 0.3$.

4.2. Convergence and stability of discretization method

In this section, we evaluate the performance of the unfitted discretization method introduced in Section 2 against all proposed benchmark contact problems. We use the same mesh hierarchy defined on levels $L0, L1, \ldots, L5$. The solution computed on the mesh on the finest level $L5$ is taken as the reference solution and it is compared against the solutions on different discretization levels from $L0, L1, \ldots, L4$. Also, we are employing the ghost penalty stabilization term in the bilinear form with the parameter $\epsilon_G = 10^{-2}$. The resulting components of the displacement field, Cauchy stresses, and von Mises stress for Example 1-SC are shown in Fig. 5. In Fig. 6, we can observe the resulting displacement field and the stress fields for the two-body problem with circular interface Example 1-TC with $E_1 = E_2 = 10\,\text{MPa}$. Similarly, the result of the two-body contact problem with an elliptic interface Example 2-TC with $E_1 = 10\,\text{MPa}$ and $E_2 = 50\,\text{MPa}$ can be seen in Fig. 7. We can observe that the circle and the ellipse are in contact with the surrounding block on the top and bottom. In Fig. 8, we can observe the resulting normal stresses computed on the whole domain for all examples. From Fig. 8, it is clear that the resultant normal stresses on the embedded interfaces are continuous and they are negative where the two bodies are in contact with each other.

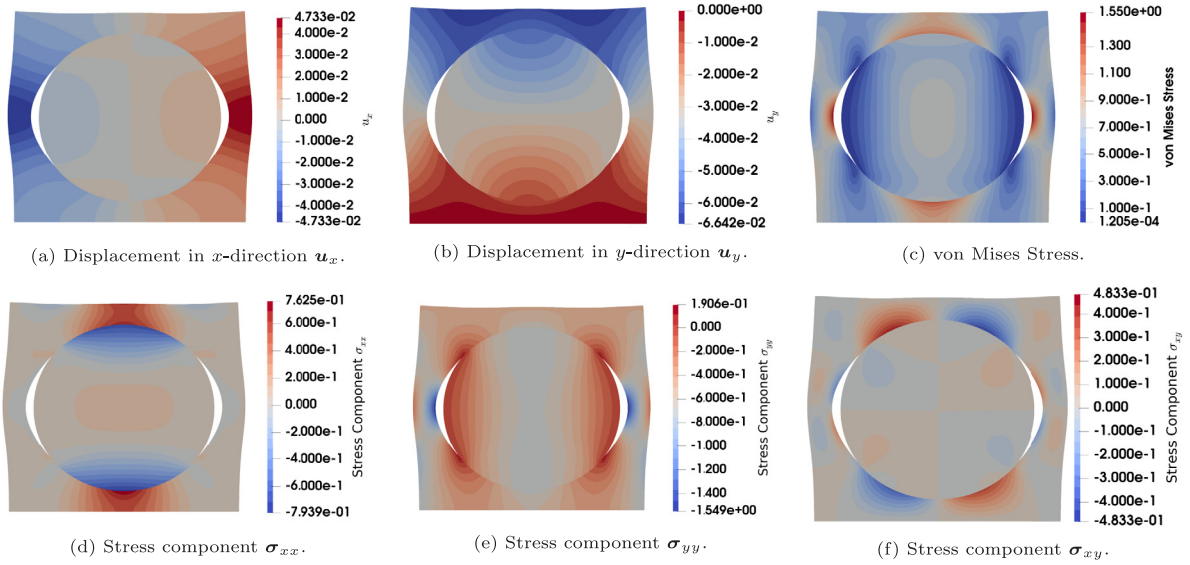
4.2.1. Analysis of discretization error

The discretization error of the displacement field is computed in two different norms, given as the energy norm $\|\cdot\|_{E(\Omega)}$ defined as

$$\|e(u_\ell)\|_{E(\Omega)} := \|u_{\text{ref}} - u_\ell\|_{E(\Omega)} = \left(\sum_{K\in\mathcal{T}_\ell}\int_K \boldsymbol{\sigma}(\boldsymbol{u}_{\text{ref}} - \boldsymbol{u}_\ell) : \boldsymbol{\varepsilon}(\boldsymbol{u}_{\text{ref}} - \boldsymbol{u}_\ell)\, d\Omega\right)^{\frac{1}{2}}$$

and the H_1-norm given as

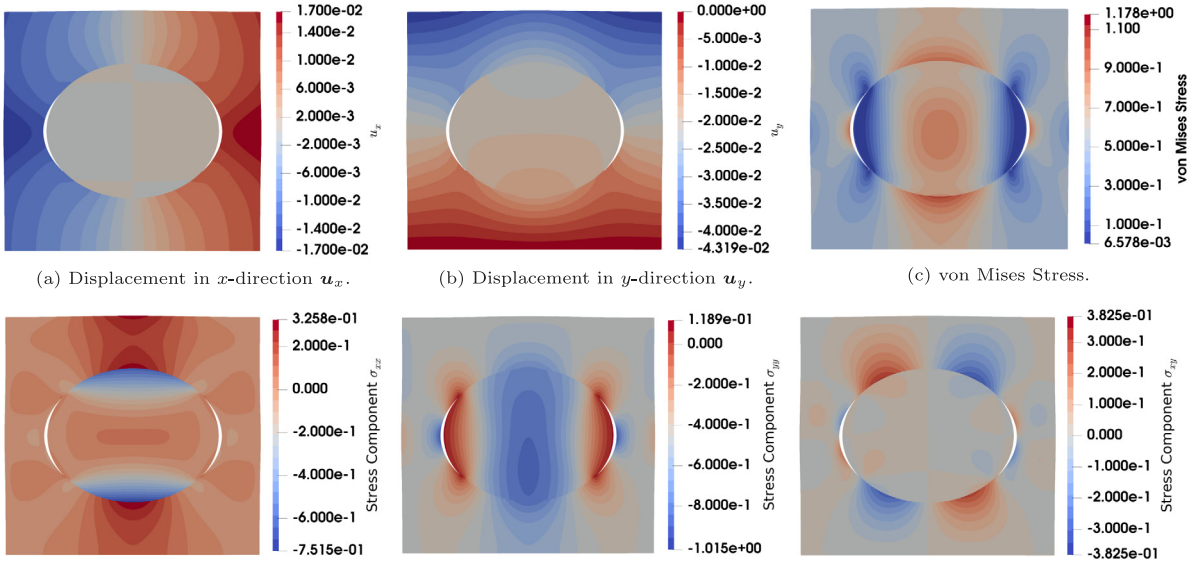
$$\|e(u_\ell)\|_{H^1(\Omega)} := \|u_{\text{ref}} - u_\ell\|_{H^1(\Omega)} = \left(\sum_{K\in\mathcal{T}_\ell}\int_K (\boldsymbol{u}_{\text{ref}} - \boldsymbol{u}_\ell)^2 + (\nabla\boldsymbol{u}_{\text{ref}} - \nabla\boldsymbol{u}_\ell)^2\, d\Omega\right)^{\frac{1}{2}}.$$



(a) Displacement in x-direction $\boldsymbol{u}_x$. (b) Displacement in y-direction $\boldsymbol{u}_y$. (c) von Mises Stress.

(d) Stress component $\boldsymbol{\sigma}_{xx}$. (e) Stress component $\boldsymbol{\sigma}_{yy}$. (f) Stress component $\boldsymbol{\sigma}_{xy}$.

Fig. 6. Resulting displacement field and stress field, as a solution of the two-body contact problem, Example 1-TC, with Young's modulus $E_1 = E_2 = 10\,\text{MPa}$, where the domain Ω^2 is a circle.



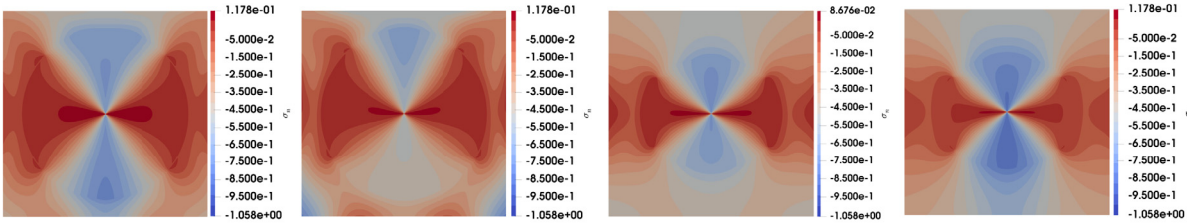
(a) Displacement in x-direction $\boldsymbol{u}_x$. (b) Displacement in y-direction $\boldsymbol{u}_y$. (c) von Mises Stress.

(d) Stress component $\boldsymbol{\sigma}_{xx}$. (e) Stress component $\boldsymbol{\sigma}_{yy}$. (f) Stress component $\boldsymbol{\sigma}_{xy}$.

Fig. 7. Resulting displacement field and stress field, as a solution of the two-body contact problem, Example 2-TC, with Young's modulus $E_1 = 10\,\text{MPa}$ and $E_2 = 50\,\text{MPa}$, where the domain Ω^2 is an ellipse.

Additionally, the discretization error in the normal stresses on the contact boundary is given in a mesh dependent norm on the interface as

$$\|e((\sigma_n)_\ell)\|_{H^{-\frac{1}{2}}(\Gamma_c),h} := \|(\sigma_n)_{\text{ref}} - (\sigma_n)_\ell\|_{H^{-\frac{1}{2}}(\Gamma_c),h} = \left(\sum_{K \in \mathcal{T}_{\ell,\Gamma_c}} \int_{\Gamma_K} h_K \left((\sigma_n)_{\text{ref}} - (\sigma_n)_\ell\right)^2 \, d\Gamma \right)^{\frac{1}{2}}.$$



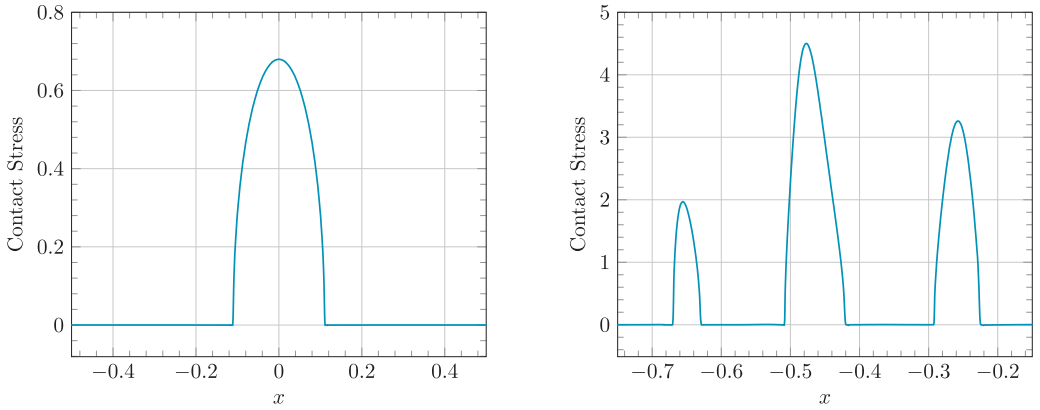
(a) Example 1-TC, $E_1 = 10\,\text{MPa}$, $E_2 = 10\,\text{MPa}$.
(b) Example 1-TC, $E_1 = 10\,\text{MPa}$, $E_2 = 50\,\text{MPa}$.
(c) Example 2-TC, $E_1 = 10\,\text{MPa}$, $E_2 = 10\,\text{MPa}$.
(d) Example 2-TC, $E_1 = 10\,\text{MPa}$, $E_2 = 50\,\text{MPa}$.

Fig. 8. Resulting nodal normal stresses (σ_n) for two-body contact problems in reference configuration.

Table 1
Discretization error in the displacement field at different levels in H^1-norm, energy norm and the normal stresses in mesh dependent norm on contact interface.

h_ℓ	$\|e(u_\ell)\|_{H^1(\Omega)}$	EOC_ℓ	$\|e(u_\ell)\|_{E(\Omega)}$	EOC_ℓ	$\|e((\sigma_n)_\ell)\|_{H^{-\frac{1}{2}}(\Gamma_c),h}$	EOC_ℓ
$2.18 \cdot 10^{-2}$	$1.28662 \cdot 10^{-3}$	–	$6.38102 \cdot 10^{-3}$	–	$3.38268 \cdot 10^{-3}$	–
$1.09 \cdot 10^{-2}$	$6.51964 \cdot 10^{-4}$	0.981	$3.24611 \cdot 10^{-3}$	0.975	$1.20316 \cdot 10^{-3}$	1.491
$5.45 \cdot 10^{-3}$	$3.29916 \cdot 10^{-4}$	0.983	$1.64475 \cdot 10^{-3}$	0.981	$4.98003 \cdot 10^{-4}$	1.273
$2.72 \cdot 10^{-3}$	$1.64492 \cdot 10^{-4}$	1.004	$8.20526 \cdot 10^{-4}$	1.003	$2.06753 \cdot 10^{-4}$	1.268
$1.36 \cdot 10^{-3}$	$7.37150 \cdot 10^{-5}$	1.158	$3.68012 \cdot 10^{-4}$	1.157	$7.20550 \cdot 10^{-5}$	1.521
(a) Example 1-SC.						
h_ℓ	$\|e(u_\ell)\|_{H^1(\Omega)}$	EOC_ℓ	$\|e(u_\ell)\|_{E(\Omega)}$	EOC_ℓ	$\|e((\sigma_n)_\ell)\|_{H^{-\frac{1}{2}}(\Gamma_c),h}$	EOC_ℓ
$2.18 \cdot 10^{-2}$	$7.94344 \cdot 10^{-3}$	–	$4.11494 \cdot 10^{-2}$	–	$3.85899 \cdot 10^{-2}$	–
$1.09 \cdot 10^{-2}$	$4.12698 \cdot 10^{-3}$	0.945	$2.14838 \cdot 10^{-2}$	0.938	$1.34525 \cdot 10^{-2}$	1.520
$5.45 \cdot 10^{-3}$	$2.08019 \cdot 10^{-3}$	0.988	$1.09024 \cdot 10^{-2}$	0.979	$4.34291 \cdot 10^{-3}$	1.631
$2.72 \cdot 10^{-3}$	$1.02215 \cdot 10^{-3}$	1.025	$5.36743 \cdot 10^{-3}$	1.022	$1.46408 \cdot 10^{-3}$	1.569
$1.36 \cdot 10^{-3}$	$4.57430 \cdot 10^{-4}$	1.160	$2.40582 \cdot 10^{-3}$	1.158	$3.86884 \cdot 10^{-4}$	1.920
(b) Example 2-SC.						
h_ℓ	$\|e(u_\ell)\|_{H^1(\Omega)}$	EOC_ℓ	$\|e(u_\ell)\|_{E(\Omega)}$	EOC_ℓ	$\|e((\sigma_n)_\ell)\|_{H^{-\frac{1}{2}}(\Gamma_c),h}$	EOC_ℓ
$2.18 \cdot 10^{-2}$	$7.72191 \cdot 10^{-3}$	–	$3.83188 \cdot 10^{-2}$	–	$5.28718 \cdot 10^{-3}$	–
$1.09 \cdot 10^{-2}$	$3.38210 \cdot 10^{-3}$	1.191	$1.74086 \cdot 10^{-2}$	1.138	$1.89146 \cdot 10^{-3}$	1.483
$5.45 \cdot 10^{-3}$	$1.50492 \cdot 10^{-3}$	1.168	$7.68281 \cdot 10^{-3}$	1.180	$7.30774 \cdot 10^{-4}$	1.372
$2.72 \cdot 10^{-3}$	$7.37569 \cdot 10^{-4}$	1.029	$3.72548 \cdot 10^{-3}$	1.044	$2.69526 \cdot 10^{-4}$	1.439
(c) Example 1-TC, $E_1 = E_2 = 10\,\text{MPa}$.						
h_ℓ	$\|e(u_\ell)\|_{H^1(\Omega)}$	EOC_ℓ	$\|e(u_\ell)\|_{E(\Omega)}$	EOC_ℓ	$\|e((\sigma_n)_\ell)\|_{H^{-\frac{1}{2}}(\Gamma_c),h}$	EOC_ℓ
$2.18 \cdot 10^{-2}$	$1.01254 \cdot 10^{-3}$	–	$5.28718 \cdot 10^{-3}$	–	$5.58060 \cdot 10^{-2}$	–
$1.09 \cdot 10^{-2}$	$4.65540 \cdot 10^{-4}$	1.121	$2.49406 \cdot 10^{-3}$	1.084	$2.24610 \cdot 10^{-2}$	1.313
$5.45 \cdot 10^{-3}$	$2.29725 \cdot 10^{-4}$	1.019	$1.23670 \cdot 10^{-3}$	1.012	$8.09678 \cdot 10^{-3}$	1.472
$2.72 \cdot 10^{-3}$	$1.15982 \cdot 10^{-4}$	0.986	$6.27417 \cdot 10^{-4}$	0.979	$3.43033 \cdot 10^{-3}$	1.239
(d) Example 2-TC, $E_1 = 10\,\text{MPa}$, $E_2 = 50\,\text{MPa}$.						

Signorini's problem. In Tables 1a and 1b, we show the discretization error for Example 1-SC and Example 2-SC in the three different norms and also compute the experimental order of convergence (EOC), $\text{EOC}_\ell = \log\big(\frac{\|e(u_{\ell-1})\|}{\|e(u_\ell)\|}\big)/\log\big(\frac{h_{\ell-1}}{h_\ell}\big)$. From Table 1, we can see that the EOC in the energy-norm and the H^1-norm have optimal rates, as the error reduces by order $\mathcal{O}(h)$. While for the normal stresses on the contact interfaces, we expect the convergence rate in the discretization error to be of order $\mathcal{O}(h^{\frac{3}{2}})$. We can see from Table 1, Example 1-SC demonstrates the optimal EOC for the normal stresses. For Example 2-SC we can see a better convergence rate than the optimal, this behavior can be attributed to the complex geometry, as the geometry is captured more accurately as the mesh is refined and due to this reason the rate of convergence of the normal stresses accelerates.



(a) Example 1-SC. (b) Example 2-SC.

Fig. 9. The induced normal contact stress on the contact boundary.

In Fig. 9, we can see the resultant induced normal stresses on the contact boundary for Example 1-SC and Example 2-SC. The resultant contact stresses also agree with the Hertzian contact theory. As for Example 1-SC, the contact stress exists only in the contact boundary and it is zero everywhere else, and the contact stress also has a parabolic shape. For Example 2-SC, we can also observe three distinct zones where the contact stresses can be seen. Additionally, as the rigid foundation is non-symmetric, the resultant normal stresses are also non-symmetric.

Two body contact problem. The discretization error for Example 1-TC with Young's modulus $E_1 = 10\,\text{MPa}$ $E_2 = 10\,\text{MPa}$ and Example 2-TC with Young's modulus $E_1 = 10\,\text{MPa}$ and $E_2 = 50\,\text{MPa}$ are shown in Tables 1c and 1d. We can observe from Table 1 that the error in H^1-norm and energy-norm reduces with the optimal rates, as the EOC for both norms is close to order $\mathcal{O}(h)$. As mentioned earlier, the discretization error of normal contact stresses on the interface should reduce with order $\mathcal{O}(h^{\frac{3}{2}})$. Here, we can observe that for the normal stresses on the contact the EOC remains close to the optimal rates.

4.2.2. Analysis of inf–sup condition

In this section, we verify if the inf–sup condition (11) is satisfied for our discretization method. We utilize the numerical inf–sup test proposed in [62]. The inf–sup condition (11) in an equivalent algebraic form can be given as

$$\inf_{\boldsymbol{\mu}\in\mathbb{R}^m} \sup_{\boldsymbol{v}\in\mathbb{R}^{nd}} \frac{(\boldsymbol{\mu}^\top \boldsymbol{B}\boldsymbol{v})^2}{h\boldsymbol{\mu}^\top \boldsymbol{M}\boldsymbol{\mu}\boldsymbol{v}^\top \boldsymbol{A}\boldsymbol{v}} \geqslant \beta^2 > 0. \tag{27}$$

The mesh dependent interface norm $\|\mu_h\|^2_{H^{-\frac{1}{2}}(\Gamma_h)}$ is represented by $h\boldsymbol{\mu}^\top \boldsymbol{M}\boldsymbol{\mu}$, where matrix $\boldsymbol{M}$ denotes the mass matrix on the multiplier space $\mathcal{M}_h$. The matrix $\boldsymbol{A}$ corresponds to the stiffness matrix which also consists the ghost penalty stabilization and $\boldsymbol{v}^\top \boldsymbol{A}\boldsymbol{v}$ is equivalent to the energy-norm $|||\boldsymbol{v}_h|||_a^2$.

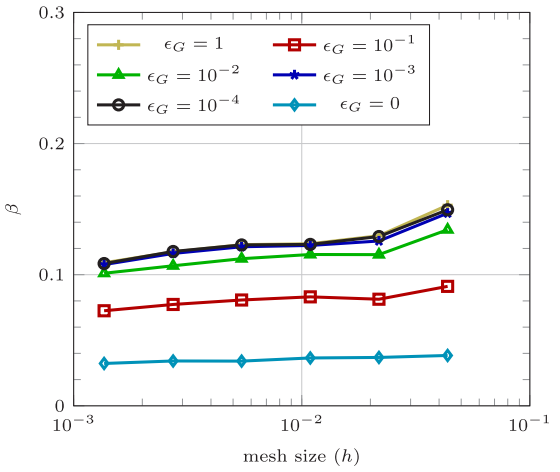
The inf–sup test is motivated from the algebraic inf–sup condition (27). By reformulating the algebraic inf–sup condition, we can write two variants of generalized eigenvalue problems. We utilize following variant of the generalized eigenvalue problem

$$\frac{1}{h}(\boldsymbol{B}^\top \boldsymbol{A}^{-1}\boldsymbol{B})\boldsymbol{\mu} = \lambda \boldsymbol{M}\boldsymbol{\mu}.$$

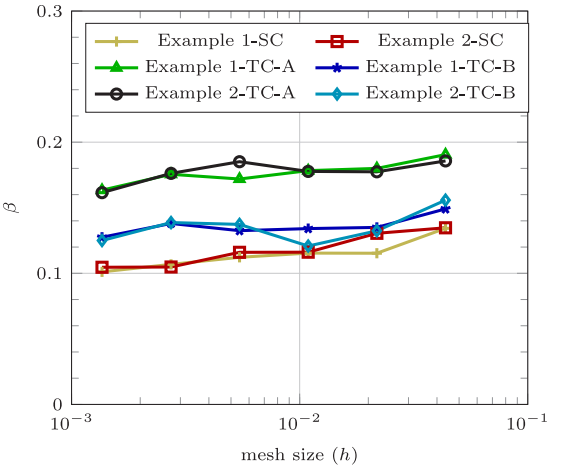
The first non-zero eigenvalue of the generalized eigenvalue problem corresponds to the constant in the inf–sup condition, $\beta = \sqrt{\lambda_{\min}}$.

Fig. 10 shows the numerically computed value of the inf–sup constant for reducing mesh size and for different values of the ghost penalty parameter.

The first set of numerical tests are carried out to evaluate the dependence of the ghost penalty term on the constant β. To this end, we consider Example 1-SC and solve the generalized eigenvalue problem with decreasing



(a) Comparison with respect to various values of ghost penalty parameter for Example 1-SC.



(b) Comparison with different examples with fixed value of the ghost penalty parameter $\epsilon_G = 10^{-2}$.

Fig. 10. Value of inf–sup constant computed using the generalized eigenvalue problem. Here, the symbol A denotes the Young's modulus $E_1 = E_2 = 10\,\text{MPa}$ and the symbol B denotes $E_1 = 10\,\text{MPa}$, $E_2 = 50\,\text{MPa}$.

mesh size. As we can see in Fig. 10(a), the value of the parameter ϵ_G in the ghost penalty term does not affect the inf–sup constant significantly. We can see that for $\epsilon_G = 0$, i.e., without ghost penalty stabilization, the value of β varies the least with the mesh size and this value is the smallest compared to all other considered values of ϵ_G. As the value of ϵ_G is changed, we observe that the value of β increases for fixed mesh size. But the effect of the ghost penalty parameter on the value of β is minimal with respect to decreasing mesh size. For the second set of experiments, we consider the proposed contact problems for a fixed value of the ghost penalty parameter $\epsilon_G = 10^{-2}$ and decreasing mesh size for all benchmark problems. In Fig. 10(b), we can observe that the value of β varies mildly with decreasing mesh size, but we do not observe any drastic changes in the value of the constant. Hence, we can claim that the discretization scheme is stable, as the value of constant β does not degenerate with decreasing mesh size.

4.3. Effect of the ghost penalty term

In Section 2.3.1, we discussed the ghost penalty term which is used to overcome the issue of ill-conditioning. In this section, we evaluate the robustness of this term by comparing the effect of different values of the ghost penalty parameter (ϵ_G). Additionally, we also evaluate the performance of the proposed multigrid method with respect to various values of the parameter ϵ_G.

In this experiment, we consider Example 1-SC on the predefined domain with the discussed boundary conditions. While the domain is kept fixed, we move the background mesh $\widetilde{\mathcal{T}}_h$ in the x-direction. We generate a set of background meshes $\{\widetilde{\mathcal{T}}_h^k\}_{k=0,\dots,10}$. Here, we consider the problem defined on level $L2$, with 400 and 200 elements in x-direction and y-direction, respectively. We recall that the $\widetilde{\mathcal{T}}_h$ is defined on a rectangle of dimension $[-1.09, 1.09]\times[0, 1.09]$. The meshes $\widetilde{\mathcal{T}}_h^k$ are given as $\widetilde{\mathcal{T}}_h^k = \widetilde{\mathcal{T}}_h + \delta_k(h_\ell/2, 0)$, where $\delta_k = 0.1k$ and $h_\ell = 5.45 \cdot 10^{-3}$. A sketch of translated mesh configuration can be seen in Fig. 11. For this experiment, the multigrid method utilizes $V(5,5)$-cycle with the modified PGS method as a smoother on the finest level, while the symmetric Gauss–Seidel method is used as a smoother on coarser levels. On the coarsest level, we use a direct solver. The multigrid hierarchy is equipped with 3 levels, where $L0$ denotes the coarsest level and $L2$ denotes the finest level.

First, we compare the condition number of the stiffness matrix $\kappa(\boldsymbol{A})$ with respect to the different values of the ghost penalty parameter. We can see in Fig. 12 that the condition number of the system matrix is highly unstable if the ghost penalty term is not employed i.e., $\epsilon_G = 0$. In the unfitted FE framework, the elements of the background mesh intersect the interface arbitrarily and on the elements with the small cuts, the gradients of the function are not bounded. If we employ the stabilization term, the condition number of the system matrix becomes stable, and

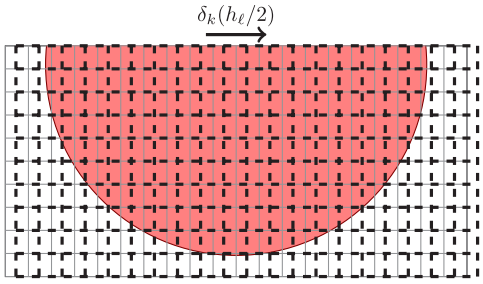


Fig. 11. Moving the background mesh, while keeping the domain Ω fixed. The solid mesh is background mesh configuration for Example 1-SC, while the mesh denoted with dashed line is the mesh moved in x-direction with $\delta_k(h_\ell/2)$ distance.

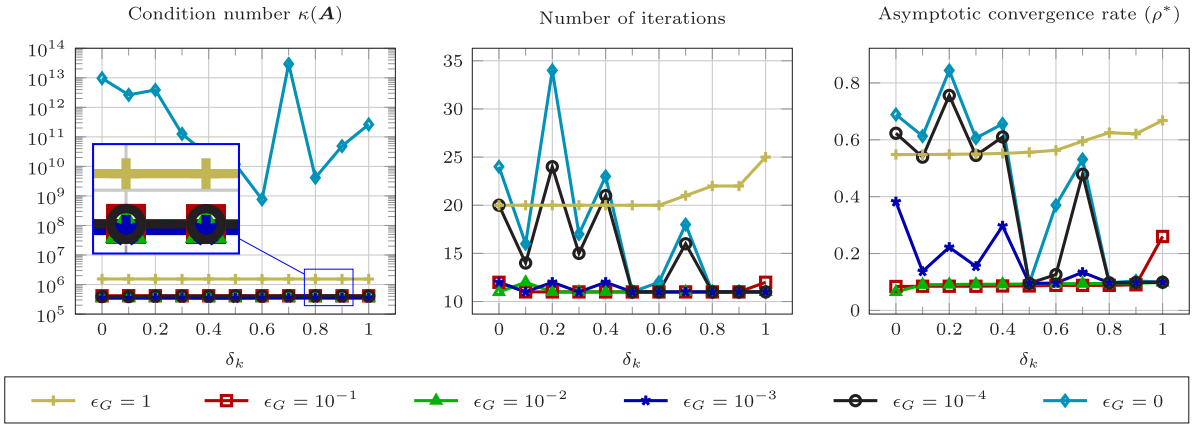


Fig. 12. Comparing the condition number of the system matrix, number of total MG iterations and the asymptotic convergence rate of the MG method for different values of ghost penalty parameter, for Example 1-SC.

also it does not vary with respect to various cut configurations while translating the mesh. We can observe that if smaller values of ϵ_G is chosen, e.g., $\epsilon_G \in \{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}\}$, the condition number of the system matrix becomes stable. In the next part, we compare the number of iterations of the MG method to reach the predefined tolerance criterion and compare also the asymptotic convergence rate. We can see that, for $\epsilon_G \in \{0, 10^{-4}\}$, the number iteration is not stable and varies with respect to different values of δ_k. But still, the number of iterations for $\epsilon_G = 10^{-4}$ is smaller than the case without the ghost penalty term. Similar behavior can also be witnessed in terms of the asymptotic convergence rate, as the asymptotic convergence rate ρ^* is smaller for $\epsilon_G = 10^{-4}$ than for $\epsilon_G = 0$. For $\epsilon_G = 10^{-3}$, the number of iterations is quite stable but the asymptotic convergence rate still oscillates with respect to moving mesh. Interestingly, the number of iterations and the asymptotic convergence rate (ρ^*) is stable for $\epsilon_G = 1$, but these values are still considerably higher than those for the smaller values of the ghost penalty parameter. The number of iterations and the asymptotic convergence rate are most stable for $\epsilon_G \in \{10^{-2}, 10^{-1}\}$. But with close observation, we can claim that for $\epsilon_G = 10^{-2}$, the condition number of the system matrix is the smallest and this also reflects in the performance of the multigrid method. Due to this reason, the default value of the ghost penalty parameter in the previous and the next experiments is chosen as $\epsilon_G = 10^{-2}$.

4.4. Performance of the multigrid method

In this section, the performance of the multigrid method is evaluated for increasing problem size and increasing the number of levels in the multigrid hierarchy. Here, all the experiments are carried out on successively finer refinement levels $L1, L2, \ldots, L5$. We employ $V(5,5)$-cycle and $W(5,5)$-cycle in the multigrid method, with the modified PGS method on the finest level and symmetric Gauss–Seidel method on the coarser levels as smoothers. We increase the number of levels in the multigrid hierarchy with the refinement levels, i.e., MG employs 2 levels for discretization level $L1$ while 6 levels are employed for discretization level $L5$.

Table 2
The number of iterations of the generalized multigrid method to reach a predefined tolerance for solving Signorini's problem.

	# levels	Example 1-SC				Example 2-SC			
		$V(5,5)$		$W(5,5)$		$V(5,5)$		$W(5,5)$	
		# iter	(ρ^*)	# iter	(ρ^*)	# iter	(ρ^*)	# iter	(ρ^*)
$L1$	2	10	(0.072)	10	(0.072)	11	(0.092)	11	(0.092)
$L2$	3	11	(0.061)	10	(0.049)	12	(0.083)	11	(0.062)
$L3$	4	11	(0.069)	10	(0.054)	14	(0.138)	12	(0.097)
$L4$	5	12	(0.086)	11	(0.046)	15	(0.132)	13	(0.086)
$L5$	6	15	(0.136)	13	(0.095)	17	(0.182)	13	(0.094)

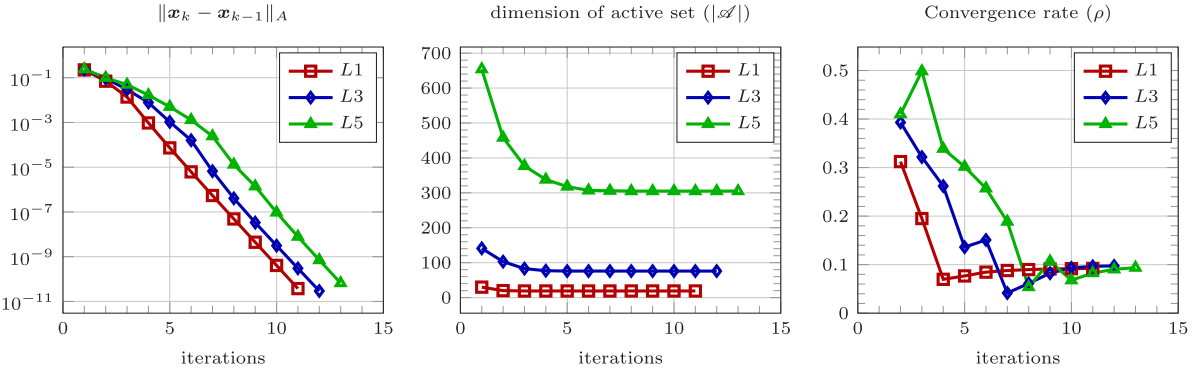


Fig. 13. The history of the correction in the energy norm, dimension of the active set, the convergence rate (ρ) at each iteration of the MG method with $W(5,5)$-cycle for solving Example 2-SC.

Signorini's problem. Table 2 illustrates the number of iterations of the generalized multigrid method to reach the termination criterion (26) and the asymptotic convergence rate for Signorini's problem. It is evident from Table 2, that the number of iterations does not vary significantly with the increasing problem size and an increasing number of levels in the hierarchy. As expected, the MG method with the W-cycle outperforms the V-cycle. While using the V-cycle the asymptotic convergence rate increases slightly with increasing problem size, but if we employ the W-cycle the asymptotic convergence rate becomes stable ($\rho^* < 0.1$). In Fig. 13, we compare the convergence history of the correction, the size of the active set, and the convergence rate at each iteration for $L1$, $L3$, $L5$, while using the MG method with W-cycle. From Fig. 13, we can see that a few initial iterations are spent for identifying the active set. Due to this reason, we can see the convergence rate is quite high and correction in the energy norm is also reducing slowly for a few initial iterations. But once the active set is identified, the norm of correction reduces at the same rate and the convergence rate also becomes stable. As we increase the problem size, the size of the active set also increases, and a few more iterations are required to identify the exact active set. Due to this reason, we observe a small increase in the number of iterations with increasing problem size.

Two-body contact problem. We employ the multigrid method with $W(5,5)$-cycle with the modified PGS method on the finest level and symmetric Gauss–Seidel method on coarse levels as smoothers. Table 3 shows the number of iterations of the generalized multigrid method to reach the termination criterion (26). We can conclude from the table, that the number of iterations does not change significantly with an increasing number of levels in the multigrid hierarchy. Also, in Table 2, we can observe the asymptotic convergence rate of the multigrid method. We can see the difference in the asymptotic convergence rates, despite the number of iterations required to reach the termination criterion is almost the same. For the case with a homogeneous value of Young's modulus, the asymptotic convergence rate is quite low ($\rho^* < 0.1$). While, for the case with different values of Young's modulus, the asymptotic convergence rate is much larger ($\rho^* < 0.55$) for the circular interface, whereas this value, is much smaller for the elliptical interface ($\rho^* < 0.15$). Even though the asymptotic convergence rate is higher for the circular interface with different Young's modulus, the number of iterations and the asymptotic convergence rate do

Table 3
The number of iterations of the generalized multigrid method ($W(5,5)$-cycle) to reach a predefined tolerance for solving two-body contact problems.

	# levels	Example 1-TC				Example 2-TC			
		$E_1 = 10, E_2 = 10$		$E_1 = 10, E_2 = 50$		$E_1 = 10, E_2 = 10$		$E_1 = 10, E_2 = 50$	
		# iter	(ρ^*)	# iter	(ρ^*)	# iter	(ρ^*)	# iter	(ρ^*)
$L1$	2	11	(0.060)	16	(0.526)	10	(0.078)	13	(0.142)
$L2$	3	11	(0.055)	19	(0.488)	11	(0.075)	14	(0.134)
$L3$	4	12	(0.064)	13	(0.165)	11	(0.065)	14	(0.120)
$L4$	5	13	(0.057)	18	(0.533)	13	(0.059)	13	(0.087)
$L5$	6	14	(0.073)	16	(0.533)	13	(0.109)	13	(0.088)

Table 4
The number of iterations of the generalized multigrid (GMG) method, semi-smooth Newton (SSN) method and interior-point (IP) method to reach predefined tolerance.

	Example 1-SC			Example 2-SC			Example 1-TC			Example 2-TC		
	$E = 10$			$E = 10$			$E_1 = 10, E_2 = 50$			$E_1 = 10, E_2 = 50$		
	GMG	SSN	IP	GMG	SSN	IP	GMG	SSN	IP	GMG	SSN	IP
$L1$	10	9	17	11	8	17	16	8	11	13	8	11
$L2$	10	9	17	11	9	18	19	8	11	14	7	12
$L3$	10	11	18	12	11	18	13	11	11	14	9	11
$L4$	11	13	18	13	16	19	18	11	11	13	10	11
$L5$	13	14	18	13	15	19	16	12	10	13	12	11

not increase with increasing problem size. Thus, we can conclude that the proposed generalized multigrid method is robust with respect to the number of levels, the material parameters, the type of obstacle, or the shape of the interface.

4.5. Comparison with other solution methods

In this section, we compare the performance of the generalized MG method with the other solution strategies such as the semi-smooth Newton (SSN) method [63] and interior-point (IP) method [64]. For this comparison, we use the MG method with $W(5,5)$-cycle and increasing number of levels in the multigrid hierarchy as discussed earlier. For the IP method and the SSN method, we employ a direct solver on each iteration. We use a predictor–corrector variant of the IP method [65], for each iteration the system matrix is factorized only once in the predictor step and the same factorization is reused for the corrector step. The IP method is used in its reduced form, hence at each iteration, a linear system with nd unknowns has to be solved. The SSN method cannot be formulated in a reduced form for the linear inequality constraints. Hence, we have to solve an enhanced KKT-system in each SSN iteration, where the linear system is formed with a non-symmetric matrix with $nd + m$ unknowns.

From Table 4, we can see that the multigrid method requires between 10 and 20 iterations to reach the termination criterion. For the SSN method, the number of iterations required to reach the termination criterion is smaller than for the IP method. It is not trivial to make a direct comparison between the IP method and the SSN method, as arising linear system of equations has a different structure. But we can safely claim that the multigrid method is computationally cheaper than the IP method and the SSN method. This is due to the reason that, the IP method and the SSN method require a solution of a linear system at each iteration, whereas the multigrid method needs only a few smoothing steps and a factorization of a linear system on the coarsest level. The computational complexity for Cholesky factorization of the symmetric positive definite FE matrices in 2D have been claim to be $\mathcal{O}(n^{\frac{3}{2}})$ and the number of non-zeros in the factorization is $\mathcal{O}(n \log n)$) [66], while the computational complexity of the multigrid method is $\mathcal{O}(n)$. Thus, the computational cost per iteration for the multigrid method compared to the other methods is significantly low.

In order to do a quantitative study of these three methods, we compare the execution time and the memory requirements of the solution process for all methods for Example 1-SC. In Fig. 14, we can observe the execution

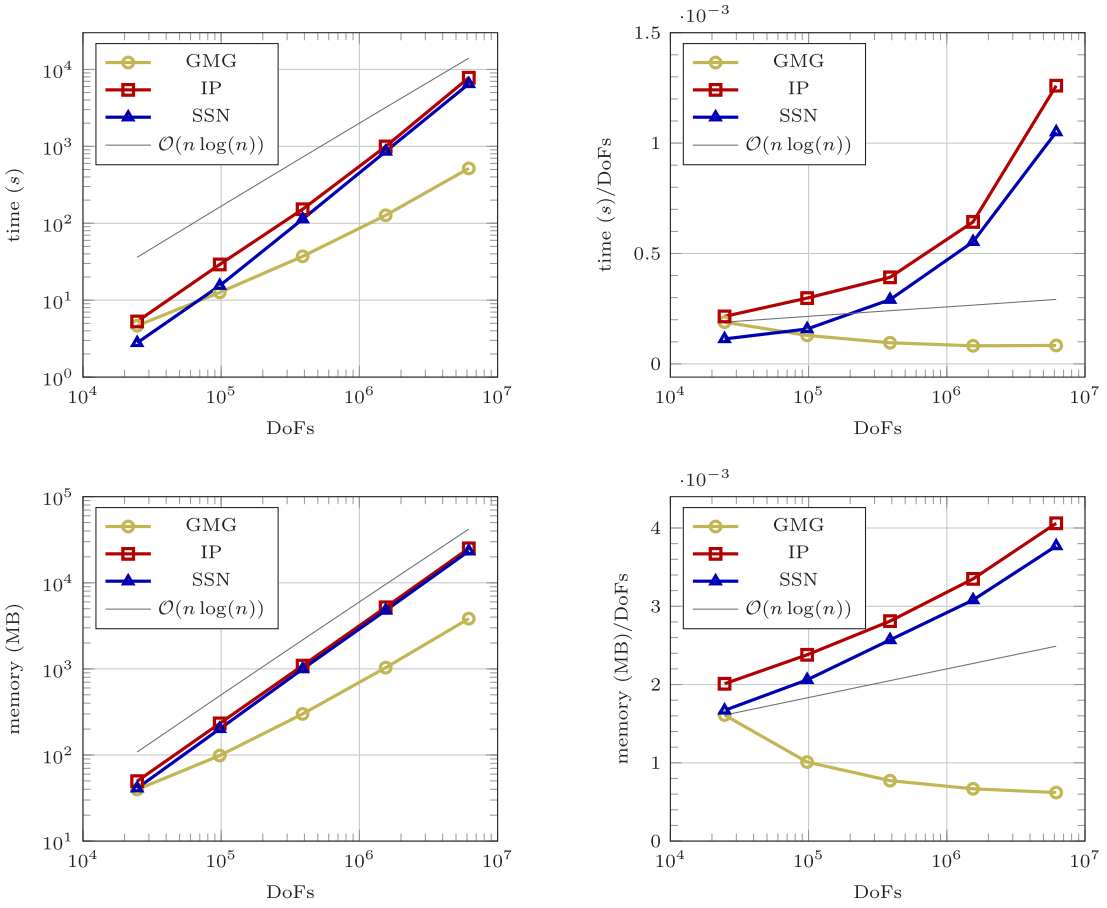


Fig. 14. Comparing the execution time and memory for difference solution schemes.

time for each of these methods in seconds and the memory required during the solution process in MB in the left column. In the right column of Fig. 14, we can observe the execution time and the memory usage per DoFs. It is clear from Fig. 14 that, the execution time and the memory requirements for all methods are comparable for smaller problem sizes. But as the problem size increases, the execution time and the memory requirements for the SSN method and the IP methods grow at a significantly higher rate than for the multigrid method. For the problem on the finest level $L5$, the memory requirement for the SSN method, the IP method, and the multigrid method is around 23.8 GB, 24.5 GB and 3.7 GB, respectively, and the execution time is around 108 min, 130 min, and 8.5 min. We can observe from Fig. 14, the time complexity of the IP method and the SSN method is lower than $\mathcal{O}(n^{\frac{3}{2}})$ but higher than $\mathcal{O}(n \log n)$. In contrast, the computational complexity of the multigrid method appears to be lower than theoretical complexity $\mathcal{O}(n)$. This can be attributed to the many implementation-related optimizations in MATLAB.

Thus, we can claim that the multigrid method is computationally cheaper in comparison with the SSN method and the IP method.

5. Conclusion

In this paper, we introduced an unfitted FE discretization for Signorini's and two-body contact problem. We utilized the vital vertex algorithm to create a stable Lagrange multiplier space which we used for discretizing the non-penetration condition. In the numerical section, we evaluated the convergence of discretization error on two different examples of Signorini's problem, which demonstrated optimal convergence properties of the unfitted FE discretization method. Later, we introduced a generalized multigrid method as an extension of the monotone

multigrid method, which can handle linear inequality constraints. We demonstrated the robustness and the efficiency of the multigrid method for solving Signorini's problem and two-body contact problems.

The generalized multigrid method introduced in this work can be used to solve quadratic constraint minimization problems where the number of constraints is significantly smaller than the number of unknowns. We aim to extend this multigrid method to solve the contact problem with higher-order discretization schemes in fitted and/or unfitted FE frameworks. Additionally, the extension of this method for the hyperelastic material models would also be a quite interesting pursuit.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

The authors would like to thank the Swiss National Science Foundation and the Deutsche Forschungsgemeinschaft (DFG), Germany for their support through the project SPP 1962 " Stress-Based Methods for Variational Inequalities in Solid Mechanics: Finite Element Discretization and Solution by Hierarchical Optimization [186407]". Additionally, we would also like to gratefully acknowledge the support of Platform for Advanced Scientific Computing (PASC) through projects FraNetG: Fracture Network Growth and FASTER: Forecasting and Assessing Seismicity and Thermal Evolution in geothermal Reservoirs.

References

[1] N. Sukumar, N. Moës, B. Moran, T. Belytschko, Extended finite element method for three-dimensional crack modelling, Int. J. Numer. Methods Eng. 48 (11) (2000) 1549–1570, http://dx.doi.org/10.1002/1097-0207(20000820)48:11<1549::AID-NME955>3.0.CO;2-A.

[2] N. Moës, J. Dolbow, T. Belytschko, A finite element method for crack growth without remeshing, Int. J. Numer. Methods Eng. 46 (1) (1999) 131–150, http://dx.doi.org/10.1002/(SICI)1097-0207(19990910)46:1<131::AID-NME726>3.0.CO;2-J.

[3] A. Hansbo, P. Hansbo, An unfitted finite element method, based on Nitsche's method, for elliptic interface problems, Comput. Methods Appl. Mech. Engrg. 191 (47) (2002) 5537–5552, http://dx.doi.org/10.1016/S0045-7825(02)00524-8.

[4] A. Hansbo, P. Hansbo, A finite element method for the simulation of strong and weak discontinuities in solid mechanics, Comput. Methods Appl. Mech. Engrg. 193 (33) (2004) 3523–3540, http://dx.doi.org/10.1016/j.cma.2003.12.041.

[5] E. Burman, S. Claus, P. Hansbo, M.G. Larson, A. Massing, CutFEM: Discretizing geometry and partial differential equations, Int. J. Numer. Methods Eng. 104 (7) (2015) 472–501, http://dx.doi.org/10.1002/nme.4823.

[6] E. Burman, Ghost penalty, C. R. Math. 348 (21–22) (2010) 1217–1220, http://dx.doi.org/10.1016/j.crma.2010.10.006.

[7] P. Wriggers, Computational Contact Mechanics, Springer, 2006.

[8] B. Wohlmuth, Variationally consistent discretization schemes and numerical algorithms for contact problems, Acta Numer. 20 (2011) 569–734, http://dx.doi.org/10.1017/S0962492911000079.

[9] T.A. Laursen, Computational Contact And Impact Mechanics: Fundamentals of Modeling Interfacial Phenomena in Nonlinear Finite Element Analysis, Springer Science & Business Media, 2013.

[10] J. Dolbow, N. Moës, T. Belytschko, An extended finite element method for modeling crack growth with frictional contact, Comput. Methods Appl. Mech. Engrg. 190 (51) (2001) 6825–6846, http://dx.doi.org/10.1016/S0045-7825(01)00260-2.

[11] P. Ladevèze, Nonlinear Computational Structural Mechanics: New Approaches and Non-Incremental Methods of Calculation, Springer Science & Business Media, 2012.

[12] A.R. Khoei, M. Nikbakht, Contact friction modeling with the extended finite element method (X-FEM), J. Mater. Process. Tech. 177 (1) (2006) 58–62, http://dx.doi.org/10.1016/j.jmatprotec.2006.03.185.

[13] F. Liu, R.I. Borja, A contact algorithm for frictional crack propagation with the extended finite element method, Int. J. Numer. Methods Eng. 76 (10) (2008) 1489–1512, http://dx.doi.org/10.1002/nme.2376.

[14] D.S. Mueller-Hoeppe, P. Wriggers, S. Loehnert, Crack face contact for a hexahedral-based XFEM formulation, Comput. Mech. 49 (6) (2012) 725–734, http://dx.doi.org/10.1007/s00466-012-0701-2.

[15] F. Chouly, P. Hild, A nitsche-based method for unilateral contact problems: numerical analysis, SIAM J. Numer. Anal. 51 (2) (2013) 1295–1307, http://dx.doi.org/10.1137/12088344X.

[16] F. Chouly, P. Hild, Y. Renard, Symmetric and non-symmetric variants of Nitsche's method for contact problems in elasticity: theory and numerical experiments, Math. Comput. 84 (293) (2015) 1089–1112, http://dx.doi.org/10.1090/S0025-5718-2014-02913-X.

[17] T. Gustafsson, R. Stenberg, J.H. Videman, Nitsche's method for unilateral contact problems, Port. Math. 75 (3) (2019) 189–204, http://dx.doi.org/10.4171/pm/2016.

[18] T. Gustafsson, R. Stenberg, J. Videman, On Nitsche's method for elastic contact problems, SIAM J. Sci. Comput. 42 (2) (2020) B425–B446, http://dx.doi.org/10.1137/19M1246869.

[19] M. Fabre, J. Pousin, Y. Renard, A fictitious domain method for frictionless contact problems in elasticity using Nitsche's method, SMAI J. Comput. Math. 2 (2016) 19–50, http://dx.doi.org/10.5802/smai-jcm.8.

[20] M. Poluektov, Ł. Figiel, A cut finite-element method for fracture and contact problems in large-deformation solid mechanics, Comput. Methods Appl. Mech. Engrg. 388 (2022) 114234, http://dx.doi.org/10.1016/j.cma.2021.114234.
[21] Y. Renard, Generalized Newton's methods for the approximation and resolution of frictional contact problems in elasticity, Comput. Methods Appl. Mech. Engrg. 256 (2013) 38–55, http://dx.doi.org/10.1016/j.cma.2012.12.008.
[22] S. Claus, P. Kerfriden, A stable and optimally convergent LaTIn-CutFEM algorithm for multiple unilateral contact problems, Int. J. Numer. Methods Eng. 113 (6) (2018) 938–966, http://dx.doi.org/10.1002/nme.5694.
[23] S. Claus, P. Kerfriden, F. Moshfeghifar, S. Darkner, K. Erleben, C. Wong, Contact modeling from images using cut finite element solvers, Adv. Model. Simult. Eng. Sci. 8 (1) (2021) 13, http://dx.doi.org/10.1186/s40323-021-00197-2.
[24] E. Béchet, N. Moës, B. Wohlmuth, A stable Lagrange multiplier space for stiff interface conditions within the extended finite element method, Int. J. Numer. Methods Eng. 78 (8) (2009) 931–954, http://dx.doi.org/10.1002/nme.2515.
[25] B.R. Akula, J. Vignollet, V.A. Yastrebov, MorteX Method for contact along real and embedded surfaces: coupling X-FEM with the Mortar method, 2019, ArXiv:1902.04000.
[26] Z. Csati, N. Moës, T.J. Massart, A stable extended/generalized finite element method with Lagrange multipliers and explicit damage update for distributed cracking in cohesive materials, Comput. Methods Appl. Mech. Engrg. 369 (2020) 113173, http://dx.doi.org/10.1016/j.cma.2020.113173.
[27] T. Belytschko, C. Parimi, N. Moës, N. Sukumar, S. Usui, Structured extended finite element methods for solids defined by implicit surfaces, Int. J. Numer. Methods Eng. 56 (4) (2003) 609–635, http://dx.doi.org/10.1002/nme.686.
[28] H. Ji, J.E. Dolbow, On strategies for enforcing interfacial constraints and evaluating jump conditions with the extended finite element method, Int. J. Numer. Methods Eng. 61 (14) (2004) 2508–2535, http://dx.doi.org/10.1002/nme.1167.
[29] H.J.C. Barbosa, T.J.R. Hughes, Circumventing the Babuška-Brezzi condition in mixed finite element approximations of elliptic variational inequalities, Comput. Methods Appl. Mech. Engrg. 97 (2) (1992) 193–210, http://dx.doi.org/10.1016/0045-7825(92)90163-E.
[30] J. Haslinger, Y. Renard, A new fictitious domain approach inspired by the extended finite element method, SIAM J. Numer. Anal. 47 (2009) http://dx.doi.org/10.1137/070704435.
[31] E. Burman, P. Hansbo, Fictitious domain finite element methods using cut elements: I. A] stabilized Lagrange multiplier method, Comput. Methods Appl. Mech. Engrg. 199 (41–44) (2010) 2680–2686, http://dx.doi.org/10.1016/j.cma.2010.05.011.
[32] E. Burman, P. Hansbo, Deriving robust unfitted finite element methods from augmented Lagrangian formulations, in: S.P.A. Bordas, E. Burman, M.G. Larson, M.A. Olshanskii (Eds.), Geometrically Unfitted Finite Element Methods And Applications, in: Lecture Notes in Computational Science and Engineering, Springer International Publishing, 2017, pp. 1–24.
[33] H.M. Mourad, J. Dolbow, I. Harari, A bubble-stabilized finite element method for Dirichlet constraints on embedded interfaces, Int. J. Numer. Methods Eng. 69 (4) (2007) 772–793, http://dx.doi.org/10.1002/nme.1788.
[34] J.E. Dolbow, L.P. Franca, Residual-free bubbles for embedded Dirichlet problems, Comput. Methods Appl. Mech. Engrg. 197 (45–48) (2008) 3751–3759, http://dx.doi.org/10.1016/j.cma.2008.02.033.
[35] M. Hautefeuille, C. Annavarapu, J.E. Dolbow, Robust imposition of Dirichlet boundary conditions on embedded surfaces, Int. J. Numer. Methods Eng. 90 (1) (2012) 40–64, http://dx.doi.org/10.1002/nme.3306.
[36] F. de Prenter, C.V. Verhoosel, E.H. van Brummelen, J.A. Evans, C. Messe, J. Benzaken, K. Maute, Multigrid solvers for immersed finite element methods and immersed isogeometric analysis, Comput. Mech. (2019) http://dx.doi.org/10.1007/s00466-019-01796-y.
[37] T. Ludescher, S. Gross, A. Reusken, A multigrid method for unfitted finite element discretizations of elliptic interface problems, SIAM J. Sci. Comput. 42 (1) (2020) A318–A342, http://dx.doi.org/10.1137/18M1203353.
[38] S. Saberi, A. Vogel, G. Meschke, Parallel finite cell method with adaptive geometric multigrid, in: M. Malawski, K. Rzadca (Eds.), Euro-Par 2020: Parallel Processing, vol. 12247, Springer International Publishing, Cham, 2020, pp. 578–593, http://dx.doi.org/10.1007/978-3-030-57675-2_36.
[39] H. Kothari, R. Krause, A multigrid method for a Nitsche-based extended finite element method, Int. J. Comput. Vis. Sci. Eng. (2021) http://dx.doi.org/10.51375/IJCVSE.2021.1.8.
[40] R. Kornhuber, R. Krause, Adaptive multigrid methods for Signorini's problem in linear elasticity, Comput. Vis. Sci. 4 (1) (2001) 9–20, http://dx.doi.org/10.1007/s007910100052.
[41] B.I. Wohlmuth, R.H. Krause, Monotone multigrid methods on nonmatching grids for nonlinear multibody contact problems, SIAM J. Sci. Comput. 25 (1) (2003) 324–347, http://dx.doi.org/10.1137/S1064827502405318.
[42] R. Krause, A nonsmooth multiscale method for solving frictional two-body contact problems in 2D and 3D with multigrid efficiency, SIAM J. Sci. Comput. 31 (2) (2009) 1399–1423, http://dx.doi.org/10.1137/070682514.
[43] T.A. Wiesner, A. Popp, M.W. Gee, W.A. Wall, Algebraic multigrid methods for dual mortar finite element formulations in contact mechanics, Int. J. Numer. Methods Eng. 114 (4) (2018) 399–430, http://dx.doi.org/10.1002/nme.5748.
[44] T.A. Wiesner, M. Mayr, A. Popp, M.W. Gee, W.A. Wall, Algebraic multigrid methods for saddle point systems arising from mortar contact formulations, Int. J. Numer. Methods Eng. (2021) http://dx.doi.org/10.1002/nme.6680.
[45] H. Kothari, R. Krause, Multigrid and saddle-point preconditioners for unfitted finite element modelling of inclusions, in: 14th WCCM-ECCOMAS Congress 2020, 2021, http://dx.doi.org/10.23967/wccm-eccomas.2020.211.
[46] R. Kornhuber, Monotone multigrid methods for elliptic variational inequalities I, Numer. Math. 69 (2) (1994) 167–184, http://dx.doi.org/10.1007/BF03325426.
[47] F. Chouly, M. Fabre, P. Hild, R. Mlika, J. Pousin, Y. Renard, An overview of recent results on Nitsche's method for contact problems, in: B. S, B. E, L. M, O. M (Eds.), UCL Workshop 2016, in: Geometrically Unfitted Finite Element Methods and Applications, Vol. 121, Springer, London, United Kingdom, 2016, pp. 93–141, http://dx.doi.org/10.1007/978-3-319-71431-8_4.
[48] S. Sticko, G. Ludvigsson, G. Kreiss, High-order cut finite elements for the elastic wave equation, Adv. Computat. Math. 46 (3) (2020) 45, http://dx.doi.org/10.1007/s10444-020-09785-z.

[49] T.Y. Kim, J. Dolbow, T. Laursen, A mortared finite element method for frictional contact on arbitrary interfaces, Comput. Mech. 39 (3) (2007) 223–235, http://dx.doi.org/10.1007/s00466-005-0019-4.
[50] R. Krause, Monotone multigrid methods for Signorini's problem with friction, (Ph.D. thesis), Freie Universität Berlin, Universität, 2001.
[51] T. Dickopf, R. Krause, Efficient simulation of multi-body contact problems on complex geometries: A flexible decomposition approach using constrained minimization, Int. J. Numer. Methods Eng. 77 (13) (2009) 1834–1862, http://dx.doi.org/10.1002/nme.2481.
[52] R. Krause, C. Mohr, Level set based multi-scale methods for large deformation contact problems, Appl. Numer. Math. 61 (4) (2011) 428–442, http://dx.doi.org/10.1016/j.apnum.2010.11.007.
[53] R. Kornhuber, Monotone multigrid methods for elliptic variational inequalities II, Numer. Math. 72 (4) (1996) 481–499, http://dx.doi.org/10.1007/s002110050178.
[54] W. Hackbusch, Multi-Grid Methods And Applications, Springer-Verlag, 1986.
[55] H. Kothari, Multilevel solution strategies for unfitted finite element methods, (Ph.D. thesis), Università della Svizzera Italiana, Lugano, 2020.
[56] T. Dickopf, R. Krause, Evaluating local approximations of the L^2-orthogonal projection between non-nested finite element spaces, Numer. Math. Theory Methods Appl. 7 (3) (2014) 288–316, http://dx.doi.org/10.1017/S100489790000012X.
[57] B.I. Wohlmuth, A mortar finite element method using dual spaces for the Lagrange multiplier, SIAM J. Numer. Anal. 38 (3) (2000) 989–1012, http://dx.doi.org/10.1137/S0036142999350929.
[58] T. Dickopf, R. Krause, A study of prolongation operators between non-nested meshes, in: Y. Huang, R. Kornhuber, O. Widlund, J. Xu (Eds.), Domain Decomposition Methods In Science And Engineering XIX, vol. 78, Springer Berlin Heidelberg, Berlin, Heidelberg, 2011, pp. 343–350, http://dx.doi.org/10.1007/978-3-642-11304-8_39.
[59] R. Glowinski, Numerical Methods For Nonlinear Variational Problems, Springer-Verlag, 1984.
[60] F. Bornemann, B. Erdmann, R. Kornhuber, Adaptive multivlevel methods in three space dimensions, Int. J. Numer. Methods Eng. 36 (18) (1993) 3187–3203, http://dx.doi.org/10.1002/nme.1620361808.
[61] R. Becker, C. Johnson, R. Rannacher, Adaptive error control for multigrid finite element, Comput. 55 (4) (1995) 271–288, http://dx.doi.org/10.1007/BF02238483.
[62] D. Chapelle, K.J. Bathe, The inf-sup test, Comput. Struct. 47 (4) (1993) 537–545, http://dx.doi.org/10.1016/0045-7949(93)90340-J.
[63] M. Ulbrich, Semismooth Newton Methods For Variational Inequalities And Constrained Optimization Problems In Function Spaces (MPS-SIAM Series On Optimization), Society for Industrial & Applied Mathematics, 2011.
[64] J. Nocedal, S. Wright, Numerical Optimization, Springer, 2000.
[65] S. Mehrotra, On the implementation of a primal-dual interior point method, SIAM J. Optim. 2 (4) (1992) 575–601, http://dx.doi.org/10.1137/0802028.
[66] A. George, E. Ng, On the complexity of sparse QR and LU factorization of finite-element matrices, SIAM J. Sci. Stat. Comput. 9 (5) (1988) 849–861, http://dx.doi.org/10.1137/0909057.