Extended mortar method for contact and mesh-tying applications

Basava Raju Akula

▶ **To cite this version:**

Basava Raju Akula. Extended mortar method for contact and mesh-tying applications. Mechanics of materials [physics.class-ph]. Université Paris sciences et lettres, 2019. English. NNT : 2019PSLEM003 . tel-02200516

HAL Id: tel-02200516

https://pastel.hal.science/tel-02200516v1

Submitted on 31 Jul 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

PSL UNIVERSITÉ PARIS

THÈSE DE DOCTORAT

DE L'UNIVERSITÉ PSL

Préparée à MINES ParisTech

Une extension de la méthode mortar pour application aux contacts et au couplage de maillages

Extended mortar method for contact and mesh-tying applications

Soutenue par

Basava Raju AKULA

Le 04 02 2019

École doctorale n°621

Ingénierie des Systèmes, Matériaux, Mécanique, Energétique

Spécialité

Mécanique

Composition du jury :

Marie-Christine Baietto Directeur de recherche, INSA Lyon	*Président*
Frédéric Lebon Professeur, Aix-Marseille Université	*Rapporteur*
Alexander Popp Professeur, Universität der Bundeswehr München	*Rapporteur*
Nicolas Moës Professeur, École centrale de Nantes	*Examinateur*
Vincent Chiaruttini Ingénieur de recherche, Onera	*Examinateur*
Julien Vignollet Ingénieur de recherche, Safran Tech	*Maître de thèse*
Vladislav Yastrebov Chargé de recherche, MINES ParisTech	*Maître de thèse*
Georges Cailletaud Professeur, MINES ParisTech	*Directeur de thèse*

MINES ParisTech | PSL

To my sister

Résumé

Cette thèse a pour but de développer un ensemble de méthodes permettant de gérer les problèmes de contact et de couplage de maillages dans le cadre de la méthode des éléments finis classiques et étendus. Ces problèmes d'interfaces sont traités le long de surfaces réelles et virtuelles, dites "surfaces immergées". Le premier objectif est d'élaborer une formulation de Mortar tridimensionnelle, efficace et parfaitement cohérente en utilisant la méthode du Lagrangien augmenté monolithique (ALM) pour traiter les problèmes de contact et de frottement. Cet objectif est réalisé dans le cadre de la méthode des éléments finis classique. Divers aspects du traitement numérique du contact sont discutés : la détection, la discrétisation, l'évaluation précise des intégrales de Mortar (projections, découpage, triangulation), la parallélisation du traitement sur des architectures parallèles à mémoire distribuée et l'optimisation de la convergence pour les problèmes impliquant à la fois le contact/frottement et les non-linéarités de comportement des matériaux. Grâce aux formulations de Mortar tirées des méthodes de décomposition de domaines, les problèmes de couplage de maillage pour la classe des interfaces non-compatibles sont également présentés.

En outre, une nouvelle méthode numérique a été élaborée en 2D : nous la dénommons "MorteX", car elle rassemble à la fois des fonctionnalités de la méthode Mortar et de la méthode X-FEM (méthode des éléments finis étendus). Dans ce cas, le couplage des maillages entre des domaines qui se chevauchent ainsi que le contact frottant entre des surfaces réelles d'un solide et certaines surfaces immergées au sein du maillage d'un autre corps peuvent être traités efficacement. Cependant, la gestion du couplage/contact entre des géométries non conformes à l'aide de surfaces immergées pose des problèmes de stabilité numérique. Nous avons donc proposé une technique de stabilisation qui consiste à introduire une interpolation des multiplicateurs de Lagrange à grains grossiers. Cette technique a été testée avec succès sur des "patch-tests" classiques et elle s'est également avérée utile pour les méthodes Mortar classiques, ce qui est illustré par plusieurs exemples pratiques.

La méthode MorteX est aussi utilisée pour traiter des problèmes d'usure en fretting. Dans ce cas, l'évolution des surfaces de contact qui résulte de l'enlèvement de matière dû à l'usure est modélisée comme une évolution de surface virtuelle qui se propage au sein du maillage existant. L'utilisation de la méthode MorteX élimine donc le besoin de recourir aux techniques complexes de remaillage. Les méthodes proposées sont développées et implémentées dans le logiciel éléments finis Z-set. De nombreux exemples numériques ont été considérés pour valider la mise en œuvre et démontrer la robustesse, la performance et la précision des méthodes Mortar et MorteX.

Abstract

In this work we develop a set of methods to handle tying and contact problems along real and virtual (embedded) surfaces in the framework of the finite element method. The first objective is to elaborate an efficient and fully consistent three-dimensional mortar formulation using the monolithic augmented Lagrangian method (ALM) to treat frictional contact problems. Various aspects of the numerical treatment of contact are discussed: detection, discretization, accurate evaluation of mortar integrals (projections, clipping, triangulation), the parallelization on distributed memory architectures and optimization of convergence for problems involving both contact and material non-linearities. With mortar methods being drawn from the domain decomposition methods, the mesh tying problems for the class on non-matching interfaces is also presented.

A new two-dimensional MorteX framework, which combines features of the extended finite element method (X-FEM) and the classical mortar methods is elaborated. Within this framework, mesh tying between overlapping domains and contact between embedded (virtual) boundaries can be treated. However, in this setting, severe manifestation of mesh locking phenomenon can take place under specific problem settings both for tying and contact. Stabilization techniques such as automatic triangulation of blending elements and coarse-grained Lagrange multiplier spaces are proposed to overcome these adverse effects. In addition, the coarse graining of Lagrange multipliers was proven to be useful for classical mortar methods, which is illustrated with relevant numerical examples.

The MorteX framework is used to treat frictional wear problems. Within this framework the contact surface evolution as a result of material removal due to wear is modeled as an evolving virtual surface. Use of MorteX method circumvents the need for complex remeshing techniques to account for contact surface evolution. The proposed methods are developed and implemented in the in-house finite element suite Z-set. Numerous numerical examples are considered to validate the implementation and demonstrate the robustness, performance and accuracy of the proposed methods.

Acknowledgements

My three year period of PhD thesis was a great learning experience. It gave me an opportunity to work with passionate researchers like Vladislav Yastrebov and Julien Vignollet (my thesis advisors). They supported and guided me through this enduring journey. I thank the Centre des matériaux, for providing us an ideal environment to work and interact with fellow researchers and Safran for the financial support. I would like thank my family who have always been a great support.

Contents

Chapter 1

Introduction

Résumé: Dans ce chapitre, les éléments qui ont motivé ce travail sont exposés. Deux types de problèmes d'interface sont présentés: le raccord de maillages non-conformes et le contact entre des domaines qui se chevauchent. Les méthodes numériques permettant de traiter les problèmes d'interface ainsi que les difficultés qui se posent sont présentées. Quelques applications sont également présentées.

1.1 Motivation and objectives

Among a wide spectrum of engineering applications, the class of problems dealing with interface mechanisms (e.g. composites, fracture, contact) is complex both with regard to their mathematical description and numerical treatment. Interface phenomena spanning wide spectrum of spatial scales (macro to nano) can manifest differently at each of these scales. Phenomena such as micro-structure modeling involving material interface, delamination along interfaces in composite structures and multi-body contact with friction demand accurate resolution of interface mechanics to understand the physical behavior/response of the system. Capturing the realistic physical behavior at the interfaces through accurate modeling of these phenomena would contribute significantly to the progress across all fields of mechanical engineering and accuracy of their analysis. Conventional numerical methods like the finite element methods (FEM) are constrained in handling many problems involving interfaces. These limitations stem from its inability to independently represent the interface and geometry discretizations.

In this thesis the focus is laid on developing a new computational framework to address both general and specific issues concerning interfaces. This framework is intended for the two classes of interface problems, namely: mesh tying and contact problems. Firstly, a three-dimensional mortar discretization framework is developed. Its ingredients include algorithms of projection, clipping and triangulation needed for accurate evaluation of interface integrals. The mesh tying problem is resolved using the standard Lagrange multiplier method. For the non-linear contact problem, a fully consistent linearization scheme is implemented within a monolithic augmented Lagrangian (ALM) [Pietrzak and Curnier, 1999] resolution scheme. The resulting framework enables to treat contact problems in an accurate and robust way. In particular, the monolithic ALM scheme, renders the contact problem fully unconstrained and does not require additional active set strategy inherent to the standard Lagrange multiplier method. The requirement of en-

suring conformal interfaces in case of node based discretization methods such as node-to-surface , are removed within the mortar method, in which the constraints are imposed in a weak sense [Bernardi et al., 2005].

The implemented mortar method is then extended to handle a particular case of non-conformal geometries. Here, we propose a two dimensional unified framework, which will be referred as MorteX, to mitigate the stringent requirements of conformity both at the level of geometric representation and interface discretizations. This framework combines the features of the Generalized/Extended Finite element Methods (GFEM/X-FEM) [Dolbow, 1999, Belytschko et al., 2009] and the mortar method [Bernardi, 1994, Belgacem, 1999]. Both the classical mortar and the MorteX methods and all related algorithms are implemented in an in-house finite element suite Z-set [Besson and Foerch, 1997]. In the following sections of this chapter, emphasis is laid on introducing the two interface problems of interest (tying and contact) and discussing relevant applications. The involved computational challenges and strategies to tackle them are elaborated.

1.1.1 Mesh tying

The finite element method is used to solve a wide range of physical and engineering problems. Based on a variational formulation and a discretized representation of the geometry, this method is extremely flexible in handling complex geometries, non-linear and heterogeneous constitutive equations and multi-physical/multi-field problems. A classification of finite element models can be proposed based on the strategy to represent the boundary of the computational mesh. Classical FE meshes fall into the category of "boundary fitted" (BF) methods, where the boundaries of the physical and computational domains coincide [Fig. 1.1(a)]. Alternatively, for "embedded/immersed boundary" (EB) methods, the computational domain is a mesh or a Cartesian grid hosting another physical domain [Fig. 1.1(b)]. Here, the portion of the physical domain (grid mesh) encompassed by the embedded boundary represents the computational domain. Note that material properties or even the governing equations of the host medium and the embedded one can be different. Within the EB method, the geometry contour can be embedded either fully or partially. The BF methods [see Fig. 1.1(a)] can be used to solve boundary value problems where the boundary conditions are prescribed on surfaces, which are explicitly represented by a mesh. The EB methods [see Fig. 1.1(b)] can handle a broader class of problems. First, it can be used to solve the same boundary value problems as BF methods but with the boundary represented by a rather general level-set function independent of the discretization of the physical domain. Second, the embedded boundary can serve as a material interface to model features such as inclusions, voids or even cracks. All these particular features can be achieved within the Generalized/Extended Finite Element framework. In addition, within the so-called CutFEM method [Burman et al., 2015, Claus and Kerfriden, 2018], even more complex problems can be handled, including for example, modeling of woven composites [Claus and Kerfriden, 2018] with interface debonding and contact. In summary, this class of methods dealing with embedded surfaces/interfaces can be easily used to create complex geometries [Belytschko et al., 2003], however prescription of boundary conditions on the embedded surfaces is not straightforward [Duboeuf and Béchet, 2017a, Duboeuf

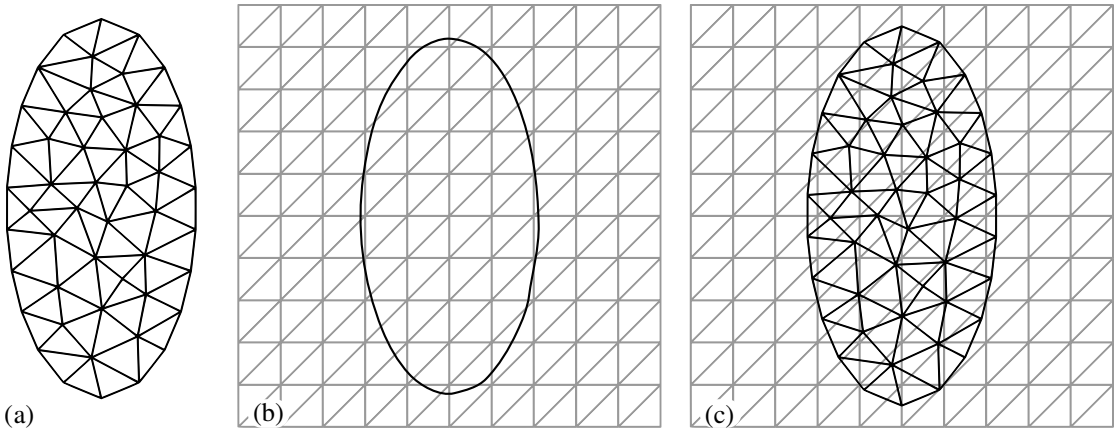
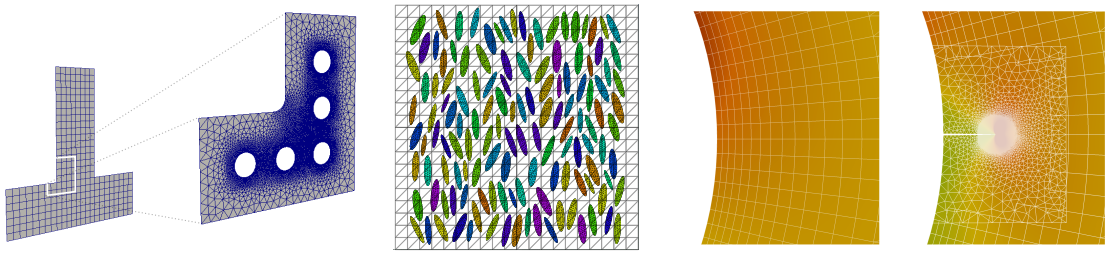


Figure 1.1: Illustration of meshes with (a) fitted boundary; (b) embedded boundary (for example, level set); (c) embedded mesh.

and Béchet, 2017b].

Here, in the MorteX framework we consider a particular combination of BF and EB methods shown in Fig. 1.1(c) which deals simultaneously with two or several superposed meshes, which can represent different physics or physical properties. This is achieved by using the features of mortar methods in the context of mesh tying, and of X-FEM in the context of void/inclusion modeling. A striking example of similar framework is the fictitious domain method, which is based on the idea of overlapping meshes [Voitsekhovskii, 1992, Glowinski et al., 1994]. This framework is created for applications involving fluid-structure interactions (FSI) [Baaijens, 2001, Fournié et al., 2014], where the background mesh represents the fluid and the embedded mesh represents the solid. Here, however we limit ourselves to the context of solid mechanics. With the emphasis laid on the interface discretizations handled by the combination of the X-FEM and the mortar methods, many applications could be cited for mesh tying applications: sub-structuring, inclusion of arbitrary geometrical features into the existing mesh, meshing complex microstructures, localized mesh refinement near crack tips, general static and dynamic mesh refinements [see Fig. 1.2] . Another class of applications deals with contact problems. The MorteX framework creates a comprehensive framework which enables handling complex contact problems involving complex surface geometries, for example, rough topographies [Fig. 1.3(a)] which can be incorporated implicitly in an existing mesh. In addition, the MorteX contact framework enables to treat computationally challenging wear problems without recourse to remeshing techniques.



(a) (b) (c)

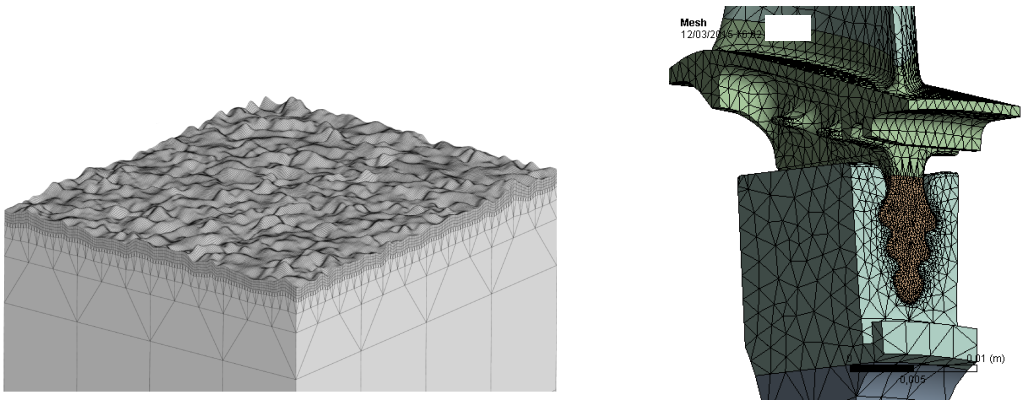
Figure 1.2: Few applications: (a) sub structuring and inclusion of arbitrary geometrical features; (b) micro structure modeling; (c) localized mesh refinement, for example, around an inserted crack.

Within the MorteX framework, the primary issue addressed is the continuity and smoothness of displacement fields across the embedded boundaries. The standard methods to impose stiff continuity or Dirichlet boundary conditions along embedded surfaces include the penalty method, the method of Lagrange multipliers, Nitsche methods and their variants [Moës et al., 2006, Sanders et al., 2009, Haslinger and Renard, 2009, Ramos et al., 2015]. Here, we impose the continuity with the Lagrange multipliers. The challenges associated with this choice are briefly presented in Section 1.3.

Few contributions harnessing the advantages of the mortar method and the X-FEM (but in a different way from what is presented here) are listed below. In [Chahine, 2008, Chahine et al., 2011] the authors used the mortar methods to ensure weak continuity conditions across the interface between a coarser mesh domain and non-intersecting finer mesh surrounding the crack, which in turn is represented by the X-FEM formulation. The tying in this case is limited to the interface with matching geometries but non-conformal discretizations, which is a classical application of the mortar method. A dual mortar contact formulation integrated into X-FEM fluid-structure interaction approach is introduced in [Mayer et al., 2010]. There, the combined X-FEM fluid-structure-contact interaction method (FSCI) allows to compute contact of arbitrarily moving and deforming structures embedded in a fluid.

The proposed method of coupling mortar and X-FEM competes with the volumetric coupling via the Arlequin method [Dhia and Rateau, 2005] and the Polytope FEM for embedded interfaces [Zamani and Eslami, 2011]. The Arlequin method involves superpositioning of mechanical states in transition zone, and involving energy redistribution between these states using weight functions. The Polytope FEM involves the decomposition of elements cut by the embedded interface into new polytope elements. It involves creation of new degrees of freedom (DoFs) along the interface lines. In a more general and flexible framework [Sanders et al., 2012], the authors used Nitsche method for imposing tying constraints for the overlapping domains circumventing mesh-locking; this method, however, requires appropriate and material dependent stabilization.

1.1.2 Contact



(a) (b)

Figure 1.3: (a) Example of a rough surface finite element mesh used to solve contact problems at microscale similar to [Yastrebov et al., 2011b]; (b) example of a finite element mesh of turbine blade-disk assembly used for fretting analysis [Source: Safran].

The transfer of mechanical forces, thermal and electrical conduction, tectonic plate motion, touching and walking are merely a few examples of the ubiquity of contact in nature, engineering and everyday life. The vast majority of phenomena occurring at the contact interface, such as wear, adhesion, damage, lubrication and fretting, as well as mass and energy transfer [Rabinowicz and Tanner, 1966, Raous et al., 1999, Sacco and Lebon, 2012], determine to the greater extent the service life of engineering components. This lays a strong emphasis on the importance to gain a fine understanding of these mechanisms for the accurate analysis and timely prediction of failure. The first work in contact mechanics dates back to 1882 with the publication of the paper by Hertz [Hertz, 1882]. Hertz in his attempt to understand the interference fringes formed between stacked lenses, solved the contact problem of two elastic bodies with curved surfaces. The further progress is associated with the initial contributions from Signorini, who formulated the general frictionless contact with rigid foundation [Signorini, 1933]. Subsequent works in the field have broadened the scope of the contact formulations beyond its original limitations such as pure elasticity, absence of friction and small deformations. One of the first numerical treatment of contact problems within the finite element framework appeared in 70s [Francavilla and Zienkiewicz, 1975]. It was a node based approach which requires conformal meshes at the contact interface. These methods were restricted to small deformations. Interesting to note, that the first engineering problem solved by the code developed by Hibbit and Karlsson, which became later Abaqus, also included frictional contact between rods and also between rods and restraint ring for nuclear power plants. The node based methods were followed by the node-to-surface techniques, which could handle large deformations and sliding [Bathe and Chaudhary, 1985, Wriggers et al., 1990, Laursen, 1992, Laursen and Simo, 1993].

In the last decade, a class of surface-to-surface contact discretizations [Simo et al., 1985, Papadopoulos and Taylor, 1992, Zavarise and Wriggers, 1998], such as mortar method, coupled with appropriate treatment of inequality contact con-

straints has been well established and proved its ability to efficiently treat contact problems [Puso and Laursen, 2004, Fischer and Wriggers, 2006, Popp, 2012]. Nevertheless, the numerical treatment of the contact and related phenomena remains particularly challenging when it involves complex surfaces, especially if they evolve in time [Fig. 1.3]. Within the standard finite element framework, these complexities require to use advanced remeshing algorithms and field transfer procedures, as for instance those encountered in the context of wear. In addition, construction of adequate finite element meshes near contact interfaces and stability of the contact formulations are necessary ingredients to ensure the overall efficiency, accuracy and robustness of the numerical procedures.

In analogy to mortar methods, which were extended to contact problems, the MorteX method, developed first for mesh tying of overlapping domains [Akula et al., 2019b], is extended here to solve contact problems between a virtual surface (represented by the X-FEM) and an explicitly represent surface of the homologue solid [Akula et al., 2019a].

1.2 Methodology

1.2.1 Mortar domain decomposition methods

The notion of domain decomposition (DD) changes with the context of the field of application. In general, it can refer to the parallel and scalable methods for the solution of initial value problems (IVP) and boundary value problems (BVP). It is a "divide and conquer" method for concurrent solving of discretized sub-problems. In this regard, the applications of DDM have a long history in computational field. DDM in a form of sub-structuring [Przemieniecki, 1963] was a technique to address the small memory of computers in the initial days. With emergence of high computing power, the parallel computing has become a dominant paradigm in numerical engineering. Synchronized with these architectures, DDM leads to radical time reductions with multiprocessor computations. The DDMs are broadly classified into overlapping and non-overlapping classes, see e.g. [Toselli and Widlund, 2006].

In DDM the analysis is done by decomposing the global domain into several local subdomains, which are coupled ensuring weak/strong continuity of the solution across the interface. In the standard DDM, discretizations of jointed surfaces are obliged to match and conform at the interface between subdomains [see Fig. 1.4]. This restriction is often infeasible or inconvenient to coordinate the decomposition and reassembly processes. The use of methods enabling coupling/tying of non-conforming surfaces circumvents this difficulty. The class of non-conforming methods includes the mortar method [McGee and Seshaiyer, 2005], which belongs to non-overlapping DDMs and forms the central part of this thesis.

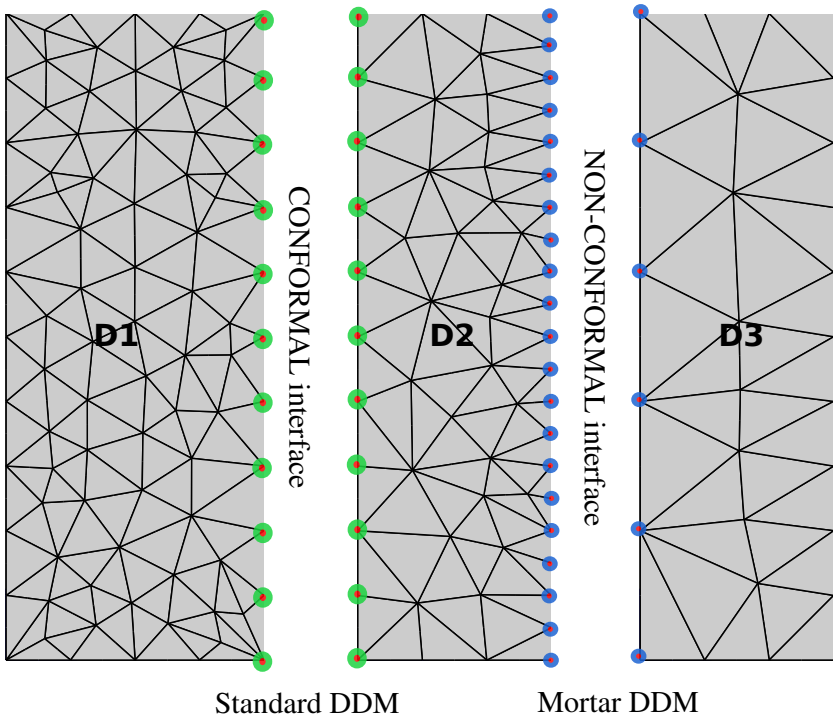


Figure 1.4: A square domain is split into subdomains with conformal and non-conformal interface discretizations.

The word "mortar" is the translation of the original French word "joint" into English. The expression illustrates the agglutinative character of this method irrespective of the meshes (the bricks of the wall) being conformal/non-conformal.

The mortar method provides us with a comprehensive framework to address the limitations of incompatible spatial interface discretizations [Bernardi, 1994]. It was originally introduced for spectral elements [Belgacem and Maday, 1994, Bernardi et al., 1990]. The coupling and tying of different physical models, discretization schemes, and/or non-matching discretizations along interfaces between domains can be ensured by this method. The mathematical optimality and applicability of the mortar method in spectral and finite element frameworks were studied extensively for elliptic problems in [Bernardi et al., 1990, Belgacem and Maday, 1994, Wohlmuth, 2001].
The added advantages of the mortar method over standard DDMs are the following:

- it possesses lesser restrictions on mesh generation by circumventing the requirement of conformity at the interface. The interfaces of subdomains having different element types like tetrahedral and hexahedral and also different interpolation orders (linear/quadratic) can be handled;
- it permits flexible mesh refinements in zones of interest or independent mesh refinement in subdomains;
- it enables tying of independently analyzed structures, for example the assembly of the fuselage and wing structure, avoiding the need for complex and costly transition mesh regions.

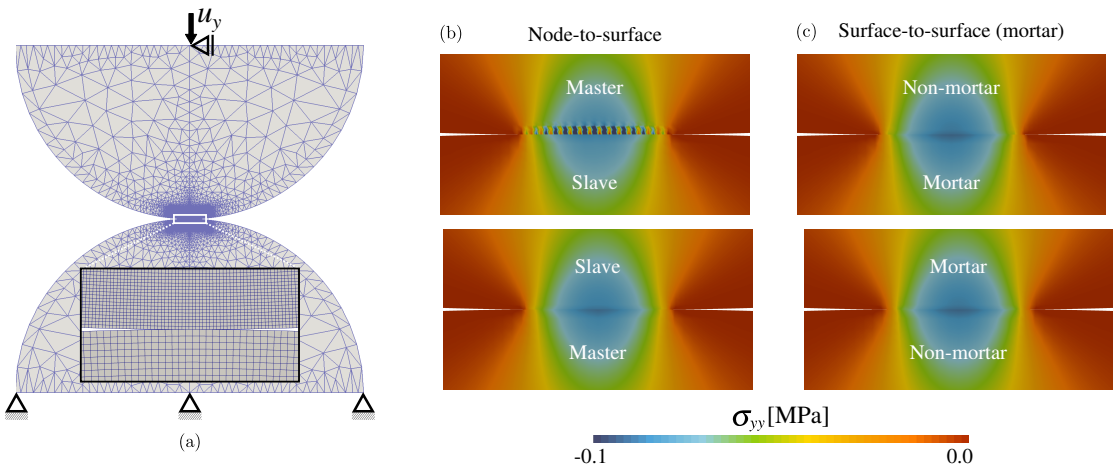


Figure 1.5: (a) Hertz contact set-up with non-matching interface discretization; (b) the stress distribution at the interface which is sensitive to the choice of master slave in case of node-to-surface; (c) smooth interfacial stress distribution irrespective of the choice of mortar and non-mortar sides in case of surface-to-surface.

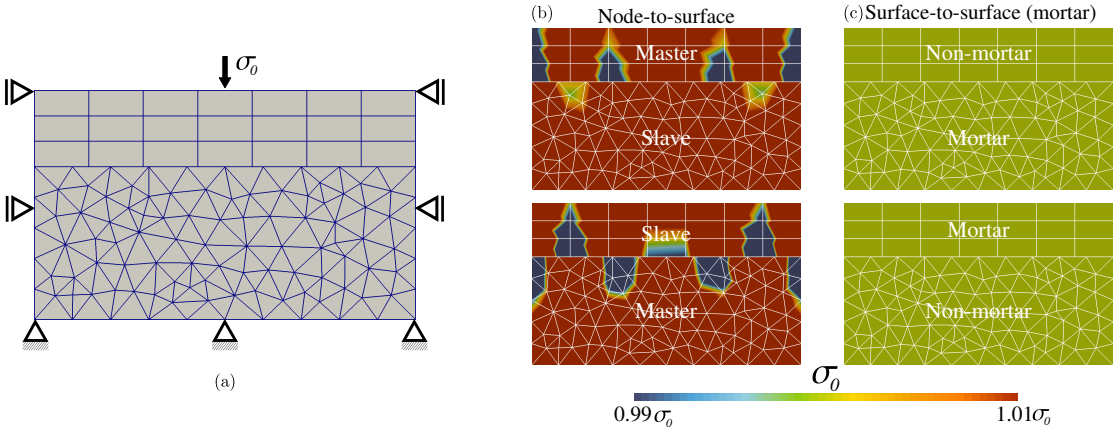


Figure 1.6: (a) Compression patch test set-up with non-matching interface discretization; (b) the node-to-surface contact discretization does not pass the patch test, i.e. non-uniform stress distribution; (c) the surface-to-surface mortar discretization passes the patch test, i.e. a uniform stress distribution.

The mortar methods were brought into the contact realm with the initial contributions from [Belgacem et al., 1998, Hild, 2000, McDevitt and Laursen, 2000]. Subsequent works [Fischer and Wriggers, 2006, Puso and Laursen, 2004, Doca et al., 2014] have extended the application of mortar methods considering higher interpolation order elements along with geometric and material non-linearities. The work of [Popp, 2012] dealt with the class of dual Lagrangian resolution schemes [Wohlmuth, 2000]. As opposed to the node-to-node discretization which is limited to small sliding and the node-to-surface which is sensitive to the choice of master and slave [see Fig. 1.5] and also does not pass patch test [see Fig. 1.6], the surface-to-surface based mortar methods are stable and very accurate in treating contact problems even under finite sliding conditions [El-Abbasi and Bathe, 2001].

1.2.2 The extended finite element method

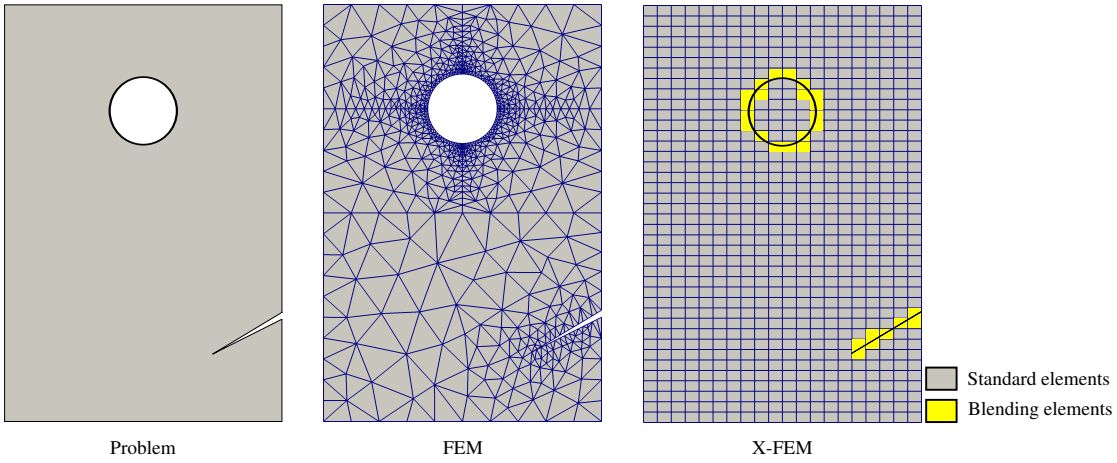


Figure 1.7: Representation of geometric features like voids and notches within FEM and X-FEM.

The FEM, despite its applicability to a broad spectrum of engineering problems, has few drawbacks. The optimal convergence of FEM, is subject to the smoothness of the problem solution represented by the polynomial approximation functions employed. Solutions containing non-smooth behavior like strong discontinuities in the displacement field as in the case of cracks, pose a computational challenge to attain optimal convergence or simply cannot be modeled. Moreover, material interfaces in the classical FEM can be located only at interfaces between elements, i.e. all the interfaces should be geometrically conformal (at least approximately) to the mesh. The extended finite element method (X-FEM) presents an efficient alternative to handle such discontinuities. The X-FEM is an enrichment method based on the partition of unity (PUM) for discontinuous fields [Melenk and Babuška, 1996, Babuška and Melenk, 1997]. In this framework, material interfaces or other discontinuities should not be conformal to the mesh but can be implicitly included in the given discretization [see Fig. 1.7]. Moreover, specific displacement/stress fields even including discontinuities of displacements like in case of cracks can be captured by enrichment functions. In X-FEM, enrichment functions are added to the finite element interpolation using the framework of PUM, to account for non-smooth behavior without compromising on the optimal convergence [Ferté et al., 2014]. The X-FEM methods are extensively used in applications such as fracture mechanics, shock wave front and oxidation front propagation, and other applications involving discontinuities both strong and weak [Daux et al., 2000, Sukumar et al., 2001, Diez et al., 2013, Gross and Reusken, 2007, Ji et al., 2002], and even frictional contact and crack modeling [Dolbow et al., 2001, Ribeaucourt et al., 2007, Khoei and Nikbakht, 2006, Liu and Borja, 2008, Gravouil et al., 2011, Mueller-Hoeppe et al., 2012]. The ability of the X-FEM to model geometric features (e.g. inclusions and voids) independently of the underlying mesh will be exploited in the proposed unified framework to describe complex and evolving surfaces.

1.3 Challenges

The above two classes of interface problems: mesh tying and contact are unified by common computational challenges. For the mesh tying problem, we use the standard Lagrange multiplier methods to impose equality constraints, whereas for contact problems involving inequality constraints the augmented Lagrangian method is used. These Lagrange multiplier based methods result in the so-called mixed finite element formulation [Brezzi and Fortin, 2012]. The choice of Lagrange multipliers functional space strongly affects the convergence rate and can lead to loss of accuracy in the interfacial tractions. These difficulties arise from a locking type phenomena reported for mixed variational formulations as a result of non-satisfaction of Ladyzhenskaya-Babuška-Brezzi (LBB) [Babuška, 1973, Brezzi and Fortin, 2012]. In particular, the issues resulting from the imposing of Dirichlet boundary conditions using Lagrange multipliers methods has been a topic of interest in various domains, such as the classical FEM [Barbosa and Hughes, 1991], Interface-enriched Generalized Finite Element Method (IGFEM) [Ramos et al., 2015], the fictitious domain methods [Burman and Hansbo, 2010], the mesh free methods [Fernández-Méndez and Huerta, 2004], etc. More importantly, this problem has been dealt extensively within the context of the X-FEM. In [Moës et al., 2006, Béchet et al., 2009, Hautefeuille et al., 2012], the authors propose a strategy to construct an optimal Lagrange-multiplier space for the embedded interfaces which permits to apply Dirichlet boundary conditions. As opposed to the strategy of modifying the Lagrange multiplier spaces, the authors in [Sanders et al., 2009] propose a stabilization method to mitigate the oscillatory behaviour of the standard spaces.

In this work, we propose a coarse-grained interpolation for the Lagrange multipliers and triangulation of blending elements as stabilization strategies to mitigate the adverse effects of mesh-locking within the MorteX framework for mesh tying problems. The coarse grained interpolations is an extension of the strategy of modifying Lagrange multiplier spaces [Moës et al., 2006, Béchet et al., 2009, Hautefeuille et al., 2012]. This strategy allows us to address specific problems of mesh-locking, which are inherent to mortar methods for overlapping domains, particularly in presence of a strong contrast of material properties and mesh densities in the vicinity of the interface. In addition, the manifestation of mesh locking effects when imposing contact constraints using Lagrange multipliers within a classical mortar method will be demonstrated for the first time to the best of our knowledge. The proposed coarse graining techniques is shown to be helpful in this context as well.

1.4 Thesis outline

The thesis is organized as follows. In Chapter 2, a brief overview of the required notions from the continuum solid mechanics is given. The majority of the chapter focuses on introducing boundary value problems with tying and contact constraints, that are solved within the thesis. The classical governing equations, the strong and weak forms of the problem settings are recalled here and they are slightly adjusted to handle overlapping domains. The continuous weak forms serve as a starting point, for the numerical framework developed in subsequent chapters.

In Chapter 3, the three dimensional classical mortar discretization methods are presented. The discrete forms of the virtual work for the tying of non-overlapping domains and contact problems are derived here. The former one using the method of Lagrange multipliers, and the latter one using the monolithic augmented Lagrangian method. Relevant aspects of the numerical procedures such as detection, geometrical operations involved in evaluation of mortar integrals, and resolution are discussed. The accuracy and robustness of the implementation are demonstrated using various numerical examples.

In Chapter 4, the two dimensional MorteX framework is presented for mesh tying between overlapping domains. The selective integration scheme of the X-FEM and enhancements to the classical mortar methods to handle constraints along a real and an embedded interface pair are elaborated. The stabilization techniques to mitigate the adverse effects of mesh-locking are introduced here.Patch tests and numerous examples are considered to validate the MorteX framework, to demonstrate its performance, and to prove the favorable effect of coarse-graining of Lagrange multipliers.

In Chapter 5, the MorteX framework is extended to solve contact problems between a real (boundary fitted) and virtual (embedded) surfaces. Discrete kinematic quantities related to the contact problem are redefined to account for the embedded interface. Few numerical examples are presented to demonstrate the performance of the framework. Results are compared with those obtained using the classical mortar method for analogous problems. The applicability of the stabilization techniques to avoid mesh-locking for both the MorteX and mortar frameworks is demonstrated. Wear problem, the key application of this framework, is also presented here: a frictional reciprocal contact is simulated between surfaces, which change as a result of material removal process due to wear.

In Chapter 6, the contributions of the thesis are summarized. Prospective work and further extensions of the elaborated methods are discussed.

Chapter 2

Continuum framework

Résumé: Dans ce chapitre, un abrégé de la mécanique non-linéaire des milieux continus est présenté. Quatre problèmes de valeurs aux limites contraints sont considérés dans la thèse: (1) le couplage de domaines non-chevauchants, (2) le contact entre des solides le long de leurs limites extérieures, (3) le couplage de domaines qui se chevauchent et (4) le contact entre des solides le long de paires surface externe / surface interne. Les quantités cinématiques et les contraintes sont exposées dans leur forme forte. Les méthodes permettant la prise en compte des contrainte, par exemple la méthode de pénalisation et les méthodes basées sur les multiplicateurs de Lagrange, sont discutées. Les formulations faibles des problèmes introduits sont dérivées en utilisant le principe du travail virtuel et des fonctionnelles non-contraintes obtenues en appliquant les techniques d'optimisation.

In this chapter, a brief overview of the non-linear continuum solid mechanics is given. A significant part of this chapter is focused on introducing the continuum description of the four constrained boundary value problem settings considered in the thesis: (1) tying of non-overlapping solids, (2) classical contact between solids along their outer boundaries, (3) tying problems of overlapping solids, and (4) contact between solids along outer and inner surface pair. This chapter also includes a description of necessary kinematic quantities, followed by the constraint definitions formulated within a strong form. The constrained optimization techniques: penalty method, Lagrange multiplier method and augmented Lagrangian method are discussed. The weak formulations of the introduced problems is derived using the principle of virtual work and unconstrained functionals obtained by applying the optimization techniques.

2.1 Basics of solid mechanics

This section introduces the basic governing equations for non-linear solid mechanics. Description is confined to necessary details to elaborate on the following boundary value problems with constraints. A comprehensive introduction and an overview of these basic concepts can be found in, e.g. [Truesdell and Noll, 2004, Bonet and Wood, 1997, Holzapfel, 2002]. The mathematical foundation of the elasticity theory can be found, for example, in [Marsden and Hughes, 1994].

2.1.1 Kinematics

Kinematics is a study of motion, regardless of what is causing it. This section is concerned with the mathematical description of this motion. Each of relevant quantities, for example velocity, deformation gradient or rate of deformation tensor, equips us to describe the motion of solids in general and their deformation in particular and enables formulating constitutive behaviors.

Motion and deformation

In the continuum framework a body is described by a set of continuously distributed points, which are referred to as material points. The location of the body in space can be described by a mapping called configuration. An Euclidean two- or three-dimensional space $\mathbb{R}^{2,3}$ equipped with a Cartesian coordinate system is considered. With the knowledge of location of a body in space, the motion can be described as a continuous set of configurations through which a body can pass in a given time interval. Here, the two configurations, namely the reference configuration Ω representing the domain occupied by the material points $\underline{\boldsymbol{X}}$ at $t = 0$ and the current configuration ω, which can be considered as a bijective mapping of the reference configuration points to the changed position $\underline{\boldsymbol{x}}$ for $t \in \mathbb{R}^+$ [see Fig. 2.1]. For a solid, the undeformed configuration is a natural choice for the reference configuration. The motion and deformation can be obtained using the deformation map φ_t:

$$\underline{\boldsymbol{x}} = \varphi_t(\underline{\boldsymbol{X}}, t). \tag{2.1}$$

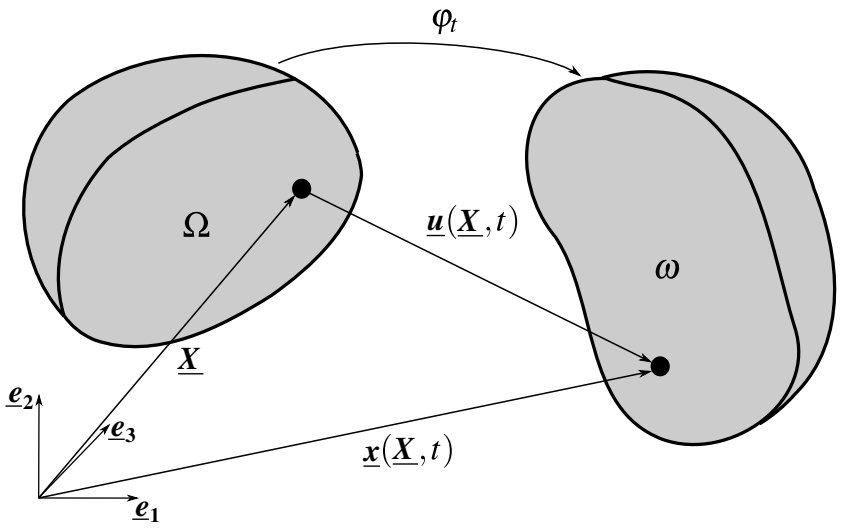


Figure 2.1: Reference (Ω) and current (ω) configurations.

It is possible to formulate the equations of continuum mechanics using either material coordinates (Lagrangian, which fit better to solid mechanics) or the spatial coordinates (Eulerian, which fits better to fluid mechanics). Here, the Lagrangian frame is used to describe various kinematic relations. The displacement $\underline{\boldsymbol{u}}$ is defined as the difference in position vectors of the material point in current and reference configurations.

$$\underline{\boldsymbol{u}}(\underline{\boldsymbol{X}}, t) = \underline{\boldsymbol{x}}(\underline{\boldsymbol{X}}, t) - \underline{\boldsymbol{X}}. \tag{2.2}$$

Deformation gradient

The mapping between the line elements of initial and current configurations called the deformation gradient $\underline{\underline{\boldsymbol{F}}}$ is used to quantify deformation in continuum mechanics. This mapping is one-to-one (i.e. $\underline{\underline{\boldsymbol{F}}}$ cannot be singular, $J = \det \underline{\underline{\boldsymbol{F}}} > 0$)

$$d\underline{\boldsymbol{x}} = \underline{\underline{\boldsymbol{F}}} \cdot d\underline{\boldsymbol{X}}. \tag{2.3}$$

The deformation gradient tensor $\underline{\underline{\boldsymbol{F}}}$, whose components are finite, characterizes the deformation in the neighborhood of a point $\underline{\boldsymbol{X}}$, mapping infinitesimal line elements $d\underline{\boldsymbol{X}}$ emanating from $\underline{\boldsymbol{X}}$ in the reference configuration to the infinitesimal line elements $d\underline{\boldsymbol{x}}$ emanating from $\underline{\boldsymbol{x}}$ in the current configuration. The deformation gradient $\underline{\underline{\boldsymbol{F}}}$ is expressed as the partial derivative of the current configuration with respect to the reference configuration:

$$\underline{\underline{\boldsymbol{F}}} = \frac{\partial \underline{\boldsymbol{x}}}{\partial \underline{\boldsymbol{X}}} = \frac{\partial(\underline{\boldsymbol{u}} + \underline{\boldsymbol{X}})}{\partial \underline{\boldsymbol{X}}} = \underline{\underline{\boldsymbol{I}}} + \frac{\partial \underline{\boldsymbol{u}}}{\partial \underline{\boldsymbol{X}}} = \underline{\underline{\boldsymbol{I}}} + \underline{\underline{\boldsymbol{H}}}, \tag{2.4}$$

where $\underline{\underline{\boldsymbol{H}}}$ is the displacement gradient. Similarly, a mapping for the area elements between the configurations is given by the Nansons formula [Ogden, 1997]

$$d\underline{\boldsymbol{a}} = \underline{\boldsymbol{n}} da = J \underline{\underline{\boldsymbol{F}}}^{-\intercal} \cdot \underline{\boldsymbol{N}} dA = J \, \underline{\underline{\boldsymbol{F}}}^{-\intercal} \cdot d\underline{\boldsymbol{A}}, \tag{2.5}$$

where $\underline{\boldsymbol{N}}$ and $\underline{\boldsymbol{n}}$ are unit normals to the area elements dA and da in reference and current configurations, respectively. The transformation between volume elements of initial and current configuration is provided by the relation

$$dv = J dV. \tag{2.6}$$

2.1.2 Strain measures

The strain measure for the reference configuration is the Green-Lagrange strain $\underline{\underline{\boldsymbol{E}}}$, defined as:

$$\underline{\underline{\boldsymbol{E}}} = \frac{1}{2}(\underline{\underline{\boldsymbol{F}}}^{\intercal} \cdot \underline{\underline{\boldsymbol{F}}} - \underline{\underline{\boldsymbol{I}}}). \tag{2.7}$$

Similarly, the Euler-Almansi strain tensor determines the strain in the current configuration and is defined as:

$$\underline{\underline{\boldsymbol{e}}} = \frac{1}{2}(\underline{\underline{\boldsymbol{I}}} - \underline{\underline{\boldsymbol{F}}}^{-\intercal} \cdot \underline{\underline{\boldsymbol{F}}}^{-1}). \tag{2.8}$$

Both the strain tensors $\underline{\underline{\boldsymbol{E}}}$ and $\underline{\underline{\boldsymbol{e}}}$ are symmetric and give information about the change in the squared length of line elements (along with the included angle between them) between the reference and current configurations.

2.1.3 Stress measures

Cauchy stress tensor

First, the traction vector on an imaginary cut through a point is defined here as a limit value of the ratio of force vector over area; for force $\Delta \underline{\boldsymbol{F}}$ acting on a surface element of area ΔA, it is given as:

$$\underline{\boldsymbol{t}}(\underline{\boldsymbol{n}}) = \lim_{\Delta A \to 0} \frac{\Delta \underline{\boldsymbol{F}}}{\Delta A}, \tag{2.9}$$

where $\underline{\boldsymbol{n}}$ denotes the normal to the imaginary cut. An infinite number of traction vectors act at a point, each acting on different cuts through the point associated with different normals. The traction vector defined on a surface element with area da in the current configuration is called the Cauchy traction vector. The Cauchy's theorem states that there exists a Cauchy stress tensor $\underline{\underline{\boldsymbol{\sigma}}}$ which maps the normal of the cut surface to the traction vector acting on that surface, according to:

$$\underline{\boldsymbol{t}} = \underline{\underline{\boldsymbol{\sigma}}} \cdot \underline{\boldsymbol{n}}. \tag{2.10}$$

The Cauchy stress $\underline{\underline{\boldsymbol{\sigma}}}$ is also called the true stress. Among many available material stress tensors the First and Second Piola-Kirchhoff tensors are presented here.

The first Piola-Kirchhoff stress tensor

Consider a vector surface element $\underline{\boldsymbol{N}}dA$ in the reference configuration. As a result of deformation, it changes into $\underline{\boldsymbol{n}}da$. Assuming that an infinitesimal force $d\underline{\boldsymbol{f}}$ is acting on the surface element in the current configuration. Then according to Cauchy's theorem:

$$d\underline{\boldsymbol{f}} = \underline{\underline{\boldsymbol{\sigma}}} \cdot \underline{\boldsymbol{n}}\, da = \underline{\underline{\boldsymbol{P}}} \cdot \underline{\boldsymbol{N}}\, dA, \tag{2.11}$$

where $\underline{\underline{\boldsymbol{P}}}$ is the first Piola-Kirchhoff stress tensor. It is a two field tensor relating the force acting in the current configuration to the surface element in the reference configuration. Similar to the Cauchy stress vector $\underline{\boldsymbol{t}}$, the first Piola-Kirchhoff traction vector $\underline{\boldsymbol{T}}$ is given as:

$$\underline{\boldsymbol{T}} = \frac{d\underline{\boldsymbol{f}}}{dA} = \underline{\underline{\boldsymbol{P}}} \cdot \underline{\boldsymbol{N}}. \tag{2.12}$$

Whereas the Cauchy traction $\underline{\boldsymbol{t}}$ carries a physical significance, representing the actual force per area on the element in the current configuration, the traction $\underline{\boldsymbol{T}}$ is a fictitious quantity with no physical significance.

The second Piola-Kirchhoff stress tensor

Due to the unsymmetrical nature of $\underline{\underline{\boldsymbol{P}}}$, a symmetrical second Piola-Kirchhoff stress tensor $\underline{\underline{\boldsymbol{S}}}$ is generally used. Like the Cauchy stress tensor quantifies the stress in the current configuration, the second Piola-Kirchhoff stress tensor $\underline{\underline{\boldsymbol{S}}}$ is symmetric and is quantifies the stress in the reference configuration. They are related through the following expression:

$$\underline{\underline{\boldsymbol{S}}} = J\underline{\underline{\boldsymbol{F}}}^{-1} \cdot \underline{\underline{\boldsymbol{\sigma}}} \cdot \underline{\underline{\boldsymbol{F}}}^{-\intercal}. \tag{2.13}$$

The first and second Piola-Kirchhoff stress tensors are related as:

$$\underline{\underline{\boldsymbol{P}}} = \underline{\underline{\boldsymbol{F}}} \cdot \underline{\underline{\boldsymbol{S}}}, \quad \underline{\underline{\boldsymbol{S}}} = \underline{\underline{\boldsymbol{F}}}^{-1} \cdot \underline{\underline{\boldsymbol{P}}}. \tag{2.14}$$

2.1.4 Balance laws

Here the fundamental balance laws of linear momentum, and angular momentum are briefly reviewed. A comprehensive overview of these laws can be found in, e.g. [Holzapfel, 2002, Belytschko et al., 2013].

Balance of of linear momentum

The balance of linear momentum states that the rate of change of momentum is equal to the forces acting on the solid. If there are no forces applied to a system, the total momentum of the system remains constant; the law in this case is known as the law of conservation of (linear) momentum. For an arbitrary control volume ω, with spatial density $\rho(\underline{\boldsymbol{x}}, t)$ and spatial velocity field $\underline{\dot{\boldsymbol{x}}}(\underline{\boldsymbol{x}}, t)$ the principle of linear momentum states:

$$\frac{d}{dt}\int_{\omega} \rho(\underline{\boldsymbol{x}}, t)\underline{\dot{\boldsymbol{x}}}(\underline{\boldsymbol{x}}, t)\, d\omega = \int_{\omega} \underline{\boldsymbol{b}}\, d\omega + \int_{\partial\omega} \underline{\boldsymbol{t}}\, da, \tag{2.15}$$

where $\underline{\boldsymbol{t}}$ represents the surface tractions acting on the boundary and $\underline{\boldsymbol{b}}$ represents the density of body force acting on the control volume. Using the definition of Cauchy traction vector $\underline{\boldsymbol{t}} = \underline{\underline{\boldsymbol{\sigma}}} \cdot \underline{\boldsymbol{n}}$ and applying the Gauss divergence theorem, we get:

$$\int_{\omega} \left(\nabla \cdot \underline{\underline{\boldsymbol{\sigma}}} + \underline{\boldsymbol{b}}\right) = \int_{\omega} \rho \underline{\ddot{\boldsymbol{x}}}. \tag{2.16}$$

Since every part of the solid should be in equilibrium, the above equation (2.16) should hold for any ω in the solid, therefore for every point of the solid the following equation should hold:

$$\nabla \cdot \underline{\underline{\boldsymbol{\sigma}}} + \underline{\boldsymbol{b}} = \rho \underline{\ddot{\boldsymbol{x}}}. \tag{2.17}$$

Similarly, for the reference configuration

$$\nabla \cdot \underline{\underline{\boldsymbol{P}}} + \underline{\boldsymbol{B}} = \rho_0 \underline{\ddot{\boldsymbol{u}}}, \tag{2.18}$$

where $\underline{\boldsymbol{B}}$ is the corresponding body force acting per unit volume in the reference configuration.

Balance of angular momentum

The resultant moment acting on a solid or any of its parts equals the rate of change of the total angular momentum. This is obtained by taking a cross product of each term in (2.15) with a position vector $\underline{\boldsymbol{r}} \in \omega$:

$$\frac{d}{dt}\int_{\omega} \underline{\boldsymbol{r}} \times \rho(\underline{\boldsymbol{x}}, t)\underline{\dot{\boldsymbol{x}}}(\underline{\boldsymbol{x}}, t)\, d\omega = \int_{\omega} \underline{\boldsymbol{r}} \times \underline{\boldsymbol{b}}\, d\omega + \int_{\partial\omega} \underline{\boldsymbol{r}} \times \underline{\boldsymbol{t}}\, da. \tag{2.19}$$

The balance of angular momentum is automatically verified, in absence of volumetric moments [Forest and Amestoy, 2018], as the stress tensor is symmetric according to the Cauchy's second law.

2.1.5 Constitutive laws

The equations introduced so far are essential to characterize kinematics, stresses and balance principles. However, they lack a physical significance i.e. they do not distinguish one material from another. Hence, they must be equipped with additional equations, called constitutive relations or equations of state, which depend on the material that the body is made of. A constitutive relation approximates

the observed physical behaviour of a material under specific conditions of interest. They are incorporated in the stress-strain relation and complete the required set of equations to formulate the boundary value problem for a single body.

In this work we consider isotropic continua, under the infinitesimal and finite deformation frameworks. The Saint-Venant-Kirchhoff constitutive law in this regard can be considered as a basic constitutive equation that employs the same stress-strain relationship as the Hook's law but not only for small but also for large strain regimes. Its general form includes the second Piola-Kirchhoff stress tensor $\underline{\underline{\boldsymbol{S}}}$ (2.13) and the Green-Lagrange strain tensor $\underline{\underline{\boldsymbol{E}}}$ (2.7) for finite deformations:

$$\underline{\underline{\boldsymbol{S}}} = \lambda \mathrm{tr}(\underline{\underline{\boldsymbol{E}}})\underline{\underline{\boldsymbol{I}}} + 2\mu\underline{\underline{\boldsymbol{E}}}. \tag{2.20}$$

where $\underline{\underline{\boldsymbol{I}}}$ is the identity tensor, and λ and μ are the Lamé constants, which can be expressed through Young's modulus E and Poisson's ratio ν as:

$$\lambda = \frac{\nu E}{(1+\nu)(1-2\nu)}, \quad \mu = \frac{E}{2(1+\nu)}.$$

2.1.6 Boundary value problem

In addition to the equations governing the kinematics, the equilibrium and the constitutive behavior, the boundary conditions are required for the boundary value problem (BVP) set-up. For this purpose we consider a domain Ω in the reference configuration. Its closure $\partial\Omega$ is constituted of Dirichlet boundary Γ_u and Neumann boundary Γ_t. The displacements $\hat{\underline{\boldsymbol{u}}}$ and tractions $\hat{\underline{\boldsymbol{T}}}$ are prescribed on them, respectively. The boundaries Γ_u and Γ_t are defined such that:

$$\Gamma_u \cup \Gamma_t = \partial\Omega, \quad \Gamma_u \cap \Gamma_t = \varnothing. \tag{2.21}$$

Note that the BVP can be analogously stated in the current configuration using the counterparts of ω for the domain and γ_u and γ_t for the Dirichlet and Neumann boundaries, respectively. Neglecting the inertia terms, the boundary value problem in the reference configuration reads as follows:

$$\nabla \cdot \underline{\underline{\boldsymbol{P}}} + \underline{\boldsymbol{B}} = 0 \qquad \text{in } \Omega, \tag{2.22}$$

$$\underline{\underline{\boldsymbol{P}}} \cdot \underline{\boldsymbol{N}} = \hat{\underline{\boldsymbol{T}}} \qquad \text{on } \Gamma_t, \tag{2.23}$$

$$\underline{\boldsymbol{u}} = \hat{\underline{\boldsymbol{u}}} \qquad \text{on } \Gamma_u. \tag{2.24}$$

Equations (2.67)-(2.69), are referred to as a strong form. A closed form solution to these equations can only be found for simple problem settings and in certain special cases [Muskhelishvili,]. Numerical solution to these partial differential equations (PDE) typically involves discretization techniques. The finite difference method, is one such technique where the discrete linear/non-linear algebraic equations from the PDE's are derived. In this work we use the Finite Element Method (FEM) as the numerical framework which relies on an integral representation of these equations called the weak form. This framework requires interpolation functions that are combined with the weak form to obtain the discrete finite element equations [Holzapfel, 2002, Reddy, 2014].

2.2 Boundary value problems with constraints

In this section a boundary value problems with constraints[1] are formulated: namely, a tying problem and a contact problem. Both these problems involve additional constraints being imposed on the solution. Equality constraints are imposed at the interface in mesh tying problems, while inequality constraints are needed in contact problems. Constraint enforcement strategies similar to those of the optimization theory can be directly used for the tying and contact problems which are typically solved as constrained minimization problems. Formulating suitable minimization functionals in these cases using the principle of virtual work renders the problem unconstrained or semi-unconstrained.

The penalty method is a widely used technique to handle constraints in the optimization theory. Owing to a formulation solely based on the primal displacement variables, the penalty methods are amongst the easiest techniques to implement. The constraint violations are penalized by a fast increase of the objective function determined by penalty parameters. The main drawback of the method, is that it does not ensure exact fulfillment of constraints. For contact problems in particular, this translates into unphysical penetrations. This can be minimized by using higher values of penalty parameters, which could however lead to ill conditioned system of equations. Its application to mesh tying problems can be found e.g. in [Pantano and Averill, 2007, Prokopyshyn, 2016], and for contact problems e.g. in [Chabrand et al., 1998, Yang et al., 2005, Fischer and Wriggers, 2006, Kuss and Lebon, 2009].

The Lagrange multiplier method fulfills the contact constraints exactly by introducing additional variables, called Lagrange multipliers. This leads to a mixed variational formulation and a saddle point type system of equations. These methods are very popular for imposing equality constraints like in the mesh tying problems [Puso, 2004, Parks et al., 2007, Boer et al., 2007] . The Lagrange multiplier method was applied to frictionless problems e.g. in [Fischer and Wriggers, 2005] and for frictional case e.g. in [Tur et al., 2009]. For the contact problem with inequality constraints, this method does not render the problem fully unconstrained, since the Lagrange multiplier field should still satisfy inequality $\lambda \leq 0$. Hence, in the discretized setting the resolution needs to be complemented with an active set strategy to resolve the active and passive constraints within the convergence loop [Björkman et al., 1995, Dumont, 1995, Wriggers, 2012]. In the thesis, the Lagrange multiplier method is employed solely for the tying problem to enforce equality constraints along the tying interfaces.

A particular combination of the penalty method and the Lagrange multipliers method leads to the so-called augmented Lagrangian method (ALM), which was first proposed in [Hestenes, 1969, Powell, 1969] for equality constraints and later extended to inequality constraints [Rockafellar, 1973]. It converges to the exact solution for a finite value of the penalty parameter (also called augmentation parameter) and renders the problem completely unconstrained and equipped with a smooth functional, which proves to be an advantage from the numerical point of view. With a smooth minimization functional in place, the standard Newton's technique can be applied. However, for the case of contact problems with

[1] To keep it concise, we will also term this problem as a "constrained boundary value problem", however this terminology could be considered non-rigorous.

non-smooth functionals, a generalized Newton scheme was proposed by Alart and Curnier for the ALM methods [Alart and Curnier, 1991, Alart, 1997]. The first applications of the ALM method for frictionless problems can be found in [Glowinski and Le Tallec, 1989], and in the report by Wriggers [Wriggers et al., 1985]. Within the Uzawa algorithm [Wriggers, 2012], which uses nested update of dual degrees of freedom, the ALM was also implemented in [Simo and Laursen, 1992]. The *monolithic* ALM method, which handles simultaneously the primal and dual degrees of freedom, was applied to frictional problems in [Alart, 1988, Alart and Curnier, 1991, Pietrzak and Curnier, 1997] within node-to-segment contact discretization. In this work, a monolithic augmented Lagrangian method is implemented for treating contact problems within the mortar framework.

2.2.1 Tying of non-overlapping domains

Fig. 2.2 shows a three dimensional tying problem set-up between two non-overlapping open domains Ω^1 and Ω^2 ($\Omega^1 \cap \Omega^2 = \varnothing$). Without loss of generality, only the case of a single tied interface is presented, an extension to multiple interfaces is straightforward. Each subdomain Ω^i is constituted of closed boundaries, the Dirichlet boundary (Γ^i_u), Neumann boundary (Γ^i_t) and the interface Γ^i_g (the subscript "g" refers to the "gluing"), i.e. $\partial\Omega^i = \Gamma^i_u \cup \Gamma^i_t \cup \Gamma^i_g$. In the continuum setting there exist no gaps between the two subdomain, i.e. $\Gamma^1_g \equiv \Gamma^2_g$. The tying problem is concerned with the enforcement of displacement continuity along the interface made of boundaries Γ^i_g.

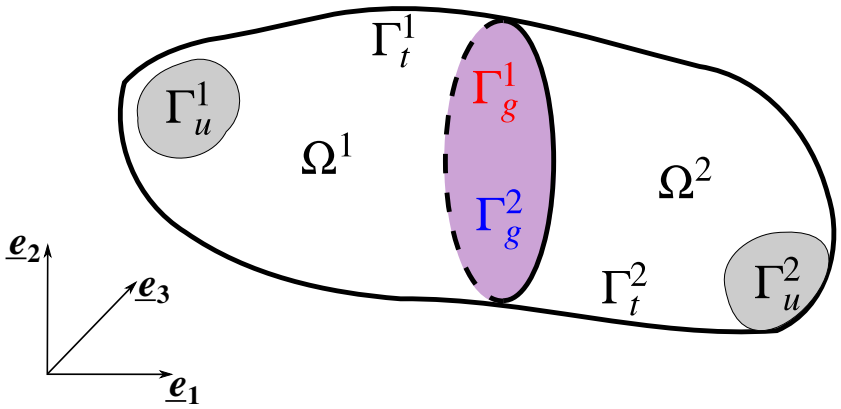


Figure 2.2: 3D continuum setting for tying of two non-overlapping domains Ω_1 and Ω_2 (reference configuration).

Kinematics

The gap is the fundamental quantity that determines the kinematics along tying interface formed from the surfaces Γ^1_g and Γ^2_g. It is defined as the relative displacement between the two sub domains assuming that in the reference configuration the two surfaces Γ^i_g ($i = 1, 2$) coincide. The gap function is given as:

$$\underline{\boldsymbol{g}}(\underline{\boldsymbol{X}}^1, \underline{\boldsymbol{X}}^2) = \underline{\boldsymbol{u}}^1(\underline{\boldsymbol{X}}^1) - \underline{\boldsymbol{u}}^2(\underline{\boldsymbol{X}}^2), \tag{2.25}$$

where $\underline{\boldsymbol{X}}^1 = \underline{\boldsymbol{X}}^2$ and $\underline{\boldsymbol{X}}^i \in \Gamma^i_g$. The displacement continuity of the tying problem translates into ensuring that the gap between the two bodies is zero at every point

$\underline{\boldsymbol{g}} = \underline{\mathbf{0}}$. However, extending the idea of constraining the displacements we could use the framework to impose periodic boundary conditions [see Section 3.7.3]. In this case the value of the gap function will be equal to a predefined value of the period based on the initial configuration.

Strong form

The constrained BVP for the tying problem involving two bodies ($i = 1, 2$), taking into account the displacement continuity along the interface Γ_g^1 is:

$$\nabla \cdot \underline{\underline{\boldsymbol{P}}}^i + \underline{\boldsymbol{B}}^i = 0 \qquad \text{in } \Omega^i, \tag{2.26}$$

$$\underline{\underline{\boldsymbol{P}}}^i \cdot \underline{\boldsymbol{N}}^i = \underline{\hat{\boldsymbol{T}}}^i \qquad \text{on } \Gamma_t^i, \tag{2.27}$$

where the displacements are selected from

$$\boldsymbol{\mathcal{K}}^i = \{\underline{\boldsymbol{u}}^i \in H^2(\Omega^i) \,|\, \underline{\boldsymbol{u}}^i = \underline{\hat{\boldsymbol{u}}}_0^i \text{ on } \Gamma_u^i \text{ and } \underline{\boldsymbol{u}}^1 = \underline{\boldsymbol{u}}^2 \text{ on } \Gamma_g^{1,2}\}, \tag{2.28}$$

where H^2 is the second order Sobolev space. Alternatively, $\underline{\boldsymbol{u}}^1 = \underline{\boldsymbol{u}}^2$ can be rewritten as $\underline{\boldsymbol{g}}(\underline{\boldsymbol{X}}^1, \underline{\boldsymbol{X}}^2) = 0$.

Weak form

The boundary problem with equality constraints is resolved using the Lagrange multiplier method. The equality constraints are imposed by the Lagrange multipliers representing tractions. It leads to a mixed variational saddle point formulation. The Lagrangian of the problem is given by:

$$\mathcal{L} = W_s + W_g, \tag{2.29}$$

where W_s is the underlying potential energy of the system, and the tying constraint is given by the following integral $W_g = \int\limits_{\Gamma_g^1} \underline{\boldsymbol{\lambda}} \cdot \underline{\boldsymbol{g}} \, d\Gamma = 0$.

The weak form derivation requires the following solution space $\boldsymbol{\mathcal{U}}^i$ and the test function space $\boldsymbol{\mathcal{V}}^i$ definitions:

$$\boldsymbol{\mathcal{U}}^i = \{\underline{\boldsymbol{u}}^i \in H^1(\Omega^i) \,|\, \underline{\boldsymbol{u}}^i = \underline{\hat{\boldsymbol{u}}}_0^i \,\text{on}\, \Gamma_u^i\},, \tag{2.30}$$

$$\boldsymbol{\mathcal{V}}^i = \{\delta\underline{\boldsymbol{u}}^i \in H^1(\Omega^i) \,|\, \delta\underline{\boldsymbol{u}}^i = 0 \,\text{on}\, \Gamma_u^i\} \tag{2.31}$$

where $\delta\underline{\boldsymbol{u}}^i$ represents the virtual displacements and $H^1(\Omega^i)$ denotes the first order Sobolev space. The Lagrange multiplier is chosen from the trace space $\boldsymbol{\mathcal{W}}^1 = H^{1/2}(\Gamma_g^1)$ of $\boldsymbol{\mathcal{U}}^1$.

The weak form of the BVP is accordingly divided into the standard solid mechanics part (δW_s) and the contribution from the tying (δW_g). The virtual work of the system is obtained by variating the Lagrangian:

$$\delta\mathcal{L}(\underline{\boldsymbol{u}}, \underline{\boldsymbol{\lambda}}) = \delta W_s + \delta W_g. \tag{2.32}$$

where

$$\delta W_s = \sum_{i=1}^{2} \Bigg(\underbrace{\int\limits_{\Omega^i} \underline{\underline{\boldsymbol{S}}}^i : \delta\underline{\underline{\boldsymbol{E}}} \, d\Omega}_{\delta W_{\text{int}}} \underbrace{- \int\limits_{\Omega^i} \underline{\boldsymbol{B}}^i \cdot \delta\underline{\boldsymbol{u}}^i \, d\Omega - \int\limits_{\Gamma_t^i} \underline{\hat{\boldsymbol{T}}}^i \cdot \delta\underline{\boldsymbol{u}}^i \, d\Gamma}_{\delta W_{\text{ext}}} \Bigg), \tag{2.33}$$

where $\delta W_{\text{int}}, \delta W_{\text{ext}}$ are the change in internal energy and the virtual work of forces, respectively. The tying virtual work is given as:

$$\delta W_g = \int_{\Gamma_g^1} \underline{\boldsymbol{\lambda}} \cdot \delta \underline{\boldsymbol{g}} \, d\Gamma + \int_{\Gamma_g^1} \delta \underline{\boldsymbol{\lambda}} \cdot \underline{\boldsymbol{g}} \, d\Gamma. \tag{2.34}$$

The first term in Eq. 2.34 represents the virtual work contribution from the tying interface, and the second term represents the weak form of the equality constraints.

2.2.2 Contact along real interfaces

Without loss of generality, we confine the continuum description of the contact problem to contact between two deformable bodies, along a single contact zone. Fig. 2.3 shows the current deformed configuration. The domains ω^i, $i = 1, 2$ have Dirichlet boundary (γ_u^i), Neumann boundary (γ_t^i) and potential contact boundaries (γ_c^i), such that $\partial\omega^i = \gamma_u^i \cup \gamma_t^i \cup \gamma_c^i$.

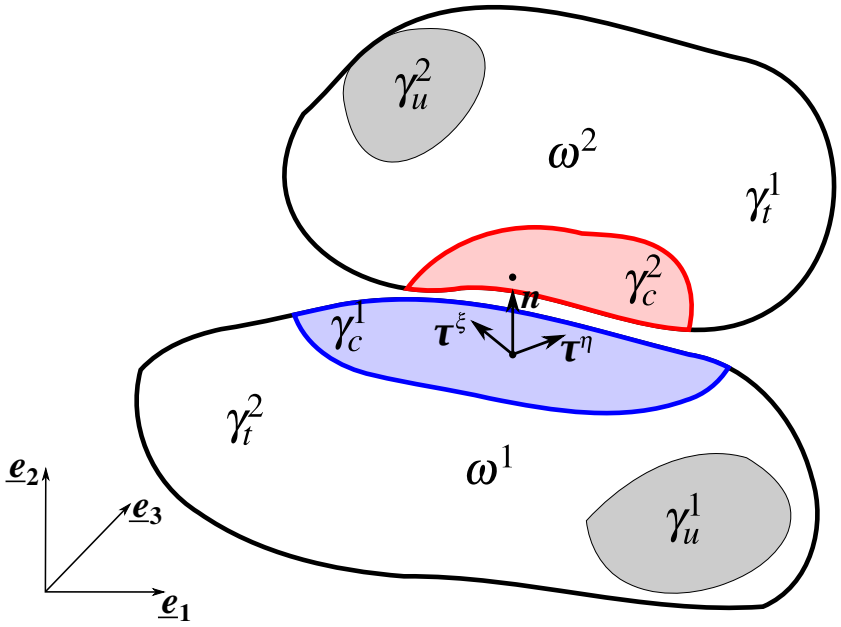


Figure 2.3: 3D continuum setting of two body contact problem (current configuration).

Kinematics

Unlike the tying problem, where the tying boundary is known a priori in the reference configuration, the contact surface γ_c^i for each contacting body is unknown and may continuously change with time, making the contact problem non-linear. The active contact surface is determined as a part of the solution scheme. Note that the inactive part of the potential contact surface γ_c^i will be interpreted as a Neumann boundary. The gap function defines the signed distance between the two surfaces and is the fundamental measure that defines the status of contact. Here, we define the gap vector (2.35) as a vector pointing from a point onto the surface γ_c^1 to its projection along the local normal on the surface γ_c^2 (the choice of

these surfaces is arbitrary). Within a discrete setting, the surfaces γ_c^1 and γ_c^2 will be referred to as mortar and non-mortar surfaces, respectively.

$$\underline{\boldsymbol{g}}(\underline{\boldsymbol{x}}^1, \gamma_c^2) = \underline{\boldsymbol{x}}^1 - \hat{\underline{\boldsymbol{x}}}^2(\underline{\boldsymbol{x}}^1, \gamma_c^2). \tag{2.35}$$

where $\hat{\underline{\boldsymbol{x}}}^2$ is the projection of $\underline{\boldsymbol{x}}^1$ onto the surface γ_c^2. The projection is defined by a mapping $\psi : \gamma_c^1 \to \gamma_c^2$ along the surface normal $\underline{\boldsymbol{n}}$ [Yang et al., 2005]. Other possible projection methods are a essentially similar ray-tracing projection [Poulios and Renard, 2015], the standard closest point projection [Konyukhov and Schweizerhof, 2008], and shadow projection method [Yastrebov, 2013] . The gap function is thus obtained as a dot product of the gap vector $\underline{\boldsymbol{g}}$ and the outward unit normal $\underline{\boldsymbol{n}}$ to the surface γ_c^1. Based on this normal gap function, there exists three possible statuses of contact, namely: no contact ($g_n > 0$), in contact ($g_n = 0$) and penetration ($g_n < 0$) [see Fig. 2.4].

$$g_n(\underline{\boldsymbol{x}}^1, \gamma_c^2) = \underline{\boldsymbol{g}} \cdot \underline{\boldsymbol{n}}. \tag{2.36}$$

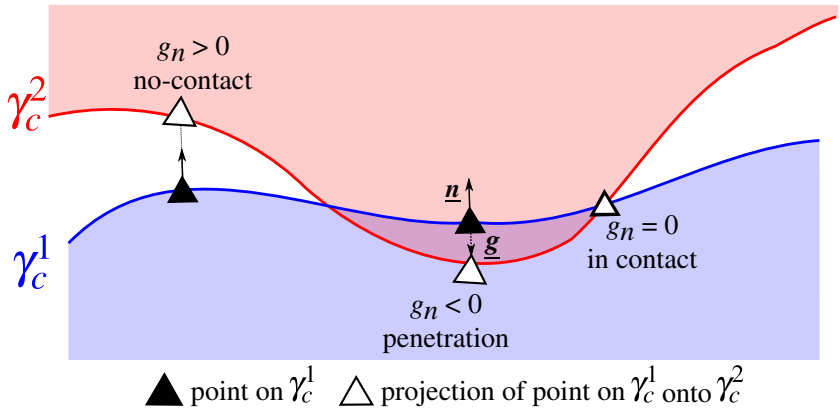


Figure 2.4: Contact status based on g_n.

For the frictional contact, an additional measure called the tangential relative velocity is required. It is given as:

$$\dot{\underline{\boldsymbol{g}}}_\tau(\underline{\boldsymbol{x}}^1) = (\underline{\underline{\boldsymbol{I}}} - \underline{\boldsymbol{n}} \otimes \underline{\boldsymbol{n}}) \cdot [\dot{\underline{\boldsymbol{x}}}^1(\underline{\boldsymbol{x}}^1) - \dot{\hat{\underline{\boldsymbol{x}}}}^2(\underline{\boldsymbol{x}}^1)], \tag{2.37}$$

where $(\underline{\underline{\boldsymbol{I}}} - \underline{\boldsymbol{n}} \otimes \underline{\boldsymbol{n}})$ is the projection tensor and $\underline{\underline{\boldsymbol{I}}}$ the identity tensor [Laursen, 2013]. In order to preserve the consistency of units in the weak form, hereinafter the incremental slip will be used instead of the tangential slip velocity as below:

$$\mathring{\underline{\boldsymbol{g}}}_\tau(\underline{\boldsymbol{x}}^1) = (\underline{\underline{\boldsymbol{I}}} - \underline{\boldsymbol{n}} \otimes \underline{\boldsymbol{n}}) \cdot [\mathring{\underline{\boldsymbol{x}}}^1(\underline{\boldsymbol{x}}^1) - \mathring{\hat{\underline{\boldsymbol{x}}}}^2(\underline{\boldsymbol{x}}^1)], \tag{2.38}$$

where

$$\mathring{(\cdot)} = \frac{d(\cdot)}{dt}\Delta t, \tag{2.39}$$

where $0 < \Delta t < \infty$ is an arbitrary time increment. This replacement does not change the physical sense of all following equations. The normal vector $\underline{\boldsymbol{n}}$, and tangent vectors $\underline{\boldsymbol{\tau}}^\xi = \partial \underline{\boldsymbol{x}}^1 / \partial \xi$ and $\underline{\boldsymbol{\tau}}^\eta = \partial \underline{\boldsymbol{x}}^1 / \partial \eta$ form a local basis at the material point $\underline{\boldsymbol{x}}^1$, and is referred to as the slip advected basis vectors. The plane defined by the vectors $(\underline{\boldsymbol{\tau}}^\xi, \underline{\boldsymbol{\tau}}^\eta)$, is the tangential slip plane [see Fig. 2.3]. Ensuring frame

indifference for measures such as the incremental slip is crucial for accurate frictional contact formulation. The incremental slip $\mathring{\underline{\boldsymbol{g}}}_\tau$ is only frame indifferent under perfect sliding conditions ($g_n = 0$). Appropriate modifications to this definition within a discretized setup will be introduced and discussed in Section 3.3.2.

In addition to the classification of the kinematic quantities into the normal and tangential components, the contact traction vector $\underline{\boldsymbol{t}}_c^1 = \underline{\underline{\boldsymbol{\sigma}}} \cdot \underline{\boldsymbol{n}}^1$ is also decomposed into its respective counterparts:

$$\underline{\boldsymbol{t}}_c^1 = p_n \underline{\boldsymbol{n}} + \underline{\boldsymbol{p}}_\tau; \quad p_n = \underline{\boldsymbol{t}}_c^1 \cdot \underline{\boldsymbol{n}}, \; \underline{\boldsymbol{p}}_\tau = p_\tau^\xi \underline{\boldsymbol{\tau}}^\xi + p_\tau^\eta \underline{\boldsymbol{\tau}}^\eta. \tag{2.40}$$

The balance of the linear momentum on the contact surface implies

$$\underline{\boldsymbol{t}}_c^1 = -\underline{\boldsymbol{t}}_c^2, \tag{2.41}$$

where $\underline{\boldsymbol{t}}_c^2 = \underline{\underline{\boldsymbol{\sigma}}} \cdot \underline{\boldsymbol{x}}^2$ is the contact traction on the surface γ_c^2. It is worth noting that in a 2D setting, the incremental slip $\mathring{g}_\tau$ and the tangential contact traction p_τ are scalar quantities.

Normal contact constraints

Along the contact interface, the classical Hertz-Signorini-Moreau conditions [Kikuchi and Oden, 1988] also known as KKT conditions in the theory of optimization [Hestenes, 1969, Powell, 1978] have to be satisfied. These conditions are formulated using the gap g_n and the contact pressure p_n as [see Fig. 2.5]:

$$g_n \geq 0, \quad p_n \leq 0, \quad p_n g_n = 0. \tag{2.42}$$

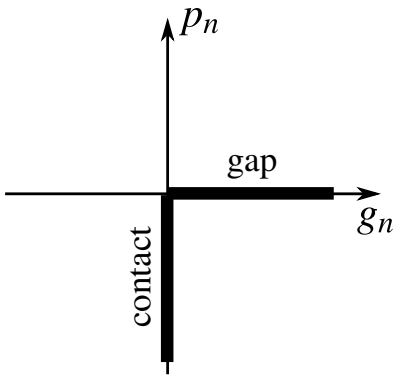


Figure 2.5: Graphical representation of the KKT normal contact conditions.

Tangential contact constraints

The friction is described using the Coulomb's friction law. The frictional constraints are formulated using tangential sliding velocity or equivalently incremental slip $\mathring{\underline{\boldsymbol{g}}}_\tau$, tangential traction $\underline{\boldsymbol{p}}_\tau$ and contact pressure p_n. The incremental slip vanishes $||\mathring{\underline{\boldsymbol{g}}}_\tau|| = 0$ when the tangential traction is below the frictional threshold $||\underline{\boldsymbol{p}}_\tau|| \leq \mu |p_n|$, non-zero incremental slip $||\mathring{\underline{\boldsymbol{g}}}_\tau|| \neq 0$ is possible when $||\underline{\boldsymbol{p}}_\tau|| = \mu |p_n|$. These conditions can also be formulated as KKT conditions [see Fig. 2.6]:

$$||\mathring{\underline{\boldsymbol{g}}}_\tau|| \geq 0, \quad ||\underline{\boldsymbol{p}}_\tau|| - \mu |p_n| \leq 0, \quad \left(\mu |p_n| - ||\underline{\boldsymbol{p}}_\tau||\right) ||\mathring{\underline{\boldsymbol{g}}}_\tau|| = 0, \tag{2.43}$$

where μ is the Coulomb's coefficient of friction.

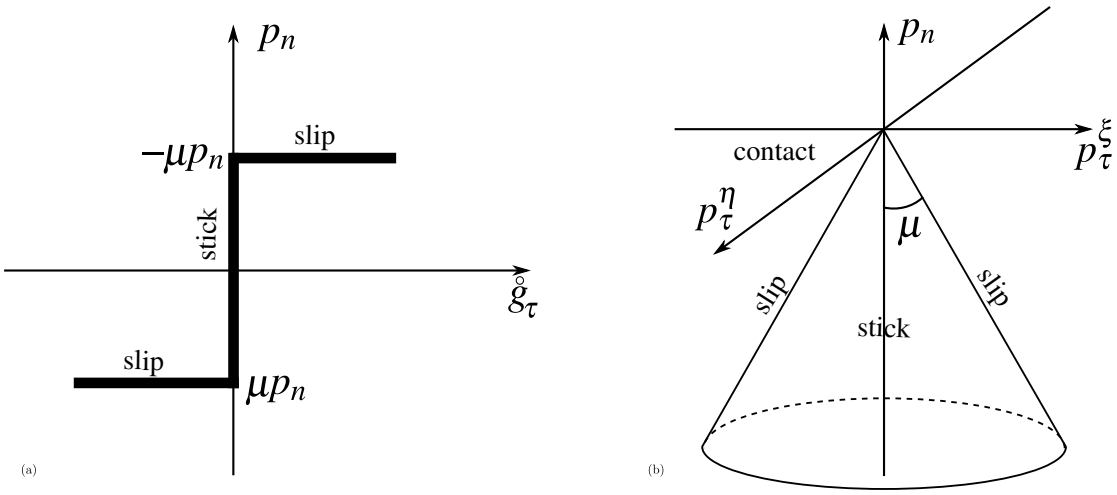


Figure 2.6: (a) Graphical representation of KKT tangential contact conditions; (b) Coulomb friction cone in traction space $\{p_n,\, p_\tau^\xi,\, p_\tau^\eta\}$.

Strong form

The BVP for the two body ($i = 1, 2$) contact problem taking into account the contact constraints (2.42),(2.43) is stated as following:

$$\nabla \cdot \underline{\underline{\boldsymbol{\sigma}}}^i + \underline{\boldsymbol{b}}^i = 0 \qquad \text{in } \omega^i, \tag{2.44}$$

$$\underline{\underline{\boldsymbol{\sigma}}}^i \cdot \underline{\boldsymbol{n}}^i = \hat{\underline{\boldsymbol{t}}}^i \qquad \text{on } \gamma_t^i, \tag{2.45}$$

$$\tag{2.46}$$

where the displacements are selected from

$$\boldsymbol{\mathcal{K}}^i = \{\underline{\boldsymbol{u}}^i \in H^2(\Omega^i) \,|\, \underline{\boldsymbol{u}}^i = \hat{\underline{\boldsymbol{u}}}_0^i \,\text{on}\, \Gamma_u^i \,\text{and}\, g_n \geq 0 \,\text{on}\, \gamma_c^1\} \tag{2.47}$$

The contact between the bodies imposes constraints which can be treated as additional configuration-dependent non-adhesive tractions $\underline{\underline{\boldsymbol{\sigma}}}^i \cdot \underline{\boldsymbol{n}}^i = \underline{\boldsymbol{t}}_c^i$ on γ_c^i, which ensure non-penetration of solids.

Weak form

The boundary value problem with contact inequality constraints is reformulated in a weak form using the monolithic augmented Lagrangian method. The weak form derivation requires the following solution space $\boldsymbol{\mathcal{U}}^i$ and the test function space $\boldsymbol{\mathcal{V}}^i$ definitions:

$$\boldsymbol{\mathcal{U}}^i = \{\underline{\boldsymbol{u}}^i \in H^1(\Omega^i) \,|\, \underline{\boldsymbol{u}}^i = \hat{\underline{\boldsymbol{u}}}_0^i \,\text{on}\, \Gamma_u^i\},, \tag{2.48}$$

$$\boldsymbol{\mathcal{V}}^i = \{\delta\underline{\boldsymbol{u}}^i \in H^1(\Omega^i) \,|\, \delta\underline{\boldsymbol{u}}^i = 0 \,\text{on}\, \Gamma_u^i\} \tag{2.49}$$

where $H^1(\Omega^i)$ denotes the first order Sobolev space. Note that the contact constraints are already removed from these spaces as they will be included in the weak formulation using the ALM. The virtual work contribution from the structural part, which contains only primal variables is given as:

$$\delta W_s = \sum_{i=1}^{2} \Bigg(\underbrace{\int_{\omega^i} \underline{\underline{\boldsymbol{\sigma}}}^i : \delta\underline{\underline{\boldsymbol{e}}}\, d\omega}_{\delta W_{\text{int}}} - \underbrace{\int_{\omega^i} \underline{\boldsymbol{b}}^i \cdot \delta\underline{\boldsymbol{u}}^i\, d\omega - \int_{\gamma_t^i} \hat{\underline{\boldsymbol{t}}}^i \cdot \delta\underline{\boldsymbol{u}}^i\, d\gamma}_{\delta W_{\text{ext}}} \Bigg), \tag{2.50}$$

Unlike the equality constraints in the tying problem, the contact problem leads to a constrained problem with inequality constraints, also referred to as the variational inequalities. This implies restrictions on the associated functional spaces for the displacements, that need to satisfy the non-penetration contact conditions [Fichera, 1973, Kikuchi and Oden, 1988]. However, as already stated, using the augmented Lagrange method as the resolution scheme removes these additional restrictions on the functional spaces for the displacements. The Lagrange multipliers are chosen from the trace space $\boldsymbol{\mathcal{W}}^1 = H^{1/2}(\gamma_c^1)$ of $\boldsymbol{\mathcal{U}}^1$.

For the virtual work contribution from the resulting contact forces we consider the generalized Newton method proposed by Alart and Curnier within the augmented Lagrangian method (ALM) framework [Alart and Curnier, 1991]. The ALM is a mixed dual formulation that introduces Lagrange multiplier vector $\underline{\boldsymbol{\lambda}}$ as dual variables along with the primal displacement variables. Lagrange multiplier vector fields can also be decomposed into normal and tangential components as:

$$\underline{\boldsymbol{\lambda}} = \lambda_n \underline{\boldsymbol{n}} + \underline{\boldsymbol{\lambda}}_\tau. \tag{2.51}$$

$$\lambda_n = \underline{\boldsymbol{\lambda}} \cdot \underline{\boldsymbol{n}}. \tag{2.52}$$

$$\underline{\boldsymbol{\lambda}}_\tau = (\underline{\underline{\boldsymbol{I}}} - \underline{\boldsymbol{n}} \otimes \underline{\boldsymbol{n}}) \cdot \underline{\boldsymbol{\lambda}}. \tag{2.53}$$

The fields of Lagrange multipliers λ_n, $\underline{\boldsymbol{\lambda}}_\tau$ are equivalents of contact pressure p_n and the tangential friction shear $\underline{\boldsymbol{p}}_\tau$ introduced earlier (2.40). The augmented Lagrangian pseudo functionals l_n and l_τ for the normal and frictional contact, respectively, are given by the following expressions [Pietrzak and Curnier, 1997, Pietrzak and Curnier, 1999]:

$$l_n(g_n, \lambda_n) = \begin{cases} \lambda_n g_n + \frac{\varepsilon_n}{2} g_n^2, & \hat{\lambda}_n \leq 0, \\ -\frac{1}{2\varepsilon_n} \lambda_n^2, & \hat{\lambda}_n > 0. \end{cases} \tag{2.54}$$

$$l_\tau(\mathring{\underline{\boldsymbol{g}}}_\tau, \underline{\boldsymbol{\lambda}}_\tau, \hat{p}_n) = \begin{cases} \left\{ \begin{array}{ll} \underline{\boldsymbol{\lambda}}_\tau \cdot \mathring{\underline{\boldsymbol{g}}}_\tau + \frac{\varepsilon_\tau}{2}(\mathring{\underline{\boldsymbol{g}}}_\tau \cdot \mathring{\underline{\boldsymbol{g}}}_\tau), & ||\underline{\boldsymbol{\lambda}}_\tau|| \leq -\mu \hat{p}_n \\ -\frac{1}{2\varepsilon_\tau}(\underline{\boldsymbol{\lambda}}_\tau \cdot \underline{\boldsymbol{\lambda}}_\tau + 2\mu \hat{p}_n ||\underline{\boldsymbol{\lambda}}_\tau|| + \mu^2 \hat{p}_n^2), & ||\underline{\boldsymbol{\lambda}}_\tau|| \geq -\mu \hat{p}_n \end{array} \right\}, & \hat{\lambda}_n \leq 0 \\ -\frac{1}{2\varepsilon_\tau} \underline{\boldsymbol{\lambda}}_\tau \cdot \underline{\boldsymbol{\lambda}}_\tau, & \hat{\lambda}_n > 0, \end{cases} \tag{2.55}$$

where $\hat{\lambda}_n, \hat{\underline{\boldsymbol{\lambda}}}_\tau, \hat{p}_n$ are the augmented normal and tangential Lagrange multipliers, and the augmented pressure, respectively:

$$\hat{\lambda}_n = \lambda_n + \varepsilon_n g_n, \quad \hat{\underline{\boldsymbol{\lambda}}}_\tau = \underline{\boldsymbol{\lambda}}_\tau + \varepsilon_\tau \mathring{\underline{\boldsymbol{g}}}_\tau, \quad \hat{p}_n = p_n + \varepsilon_n g_n, \tag{2.56}$$

where ε_n and ε_τ are the normal and tangential augmentation parameters, respectively. The augmented pressure determining the frictional threshold is replaced by the corresponding augmented Lagrange multiplier $\hat{\lambda}_n$, however $\hat{p}_n$ is not subjected to variation, therefore a different notation is used to highlight this subtle difference [Pietrzak and Curnier, 1997]. Integrating the functionals (2.54) and (2.55) over the contact surface γ_c^1 incorporates the contact constraints in the Lagrangian: Frictionless:

$$\mathcal{L}^{\text{aug}} = W_s + W_c = W_s + \int_{\gamma_c^1} l_n(g_n, \lambda_n). \tag{2.57}$$

Frictional:

$$\mathcal{L}^{\text{aug}} = W_s + W_c = W_s + \int_{\gamma_c^1} l_n(g_n, \lambda_n) + l_\tau(\mathring{\underline{\boldsymbol{g}}}_\tau, \underline{\boldsymbol{\lambda}}_\tau, \hat{p}_n)\, d\gamma. \tag{2.58}$$

The problem of constrained optimization transforms into the min-max or saddle point problem for the Lagrangian, whose solution is equivalent to the solution of the variational inequality optimization problem on a non-convex domain and its boundary [Hestenes, 1969, Powell, 1978, Pietrzak and Curnier, 1997]. The min-max problem is solved by searching the stationary point minimizing the Lagrangian with respect to primal (displacement) field and maximizing it with respect to the dual field (Lagrange multipliers). The contact virtual work for the frictionless case is:

$$\delta W_c = \int_{\gamma_c^1} \delta \left[l_n(g_n, \lambda_n) d\gamma \right] = \tag{2.59}$$

$$= \int_{\gamma_c^1} \lambda_n \delta g_n \, d\gamma + g_n \delta \lambda_n \, d\gamma + \underbrace{g_n \lambda_n}_{0} \delta d\gamma \tag{2.60}$$

Note that the last term in the frictionless weak form represents the complementary KKT condition (2.42) and is always zero in the weak form. This term is dropped for implementation purposes, and thus results in a non symmetric tangent operator for the frictionless case, in the incremental solution procedure. However, as was remarked in [Fischer and Wriggers, 2005], a consistent linearization with this term included, results in a symmetric tangent matrix. The contact virtual work for the frictional case is given by:

$$\delta W_c = \int_{\gamma_c^1} \left[\delta l_n(g_n, \lambda_n) + \delta l_\tau(\mathring{\underline{\boldsymbol{g}}}_\tau, \underline{\boldsymbol{\lambda}}_\tau, \hat{p}_n) \right] d\gamma = \tag{2.61}$$

$$= \int_{\gamma_c^1} \left[\lambda_n \delta g_n + \underline{\boldsymbol{\lambda}}_\tau \delta \mathring{\underline{\boldsymbol{g}}}_\tau \right] d\gamma + \tag{2.62}$$

$$+ \int_{\gamma_c^1} g_n \delta \lambda_n \, d\gamma + \qquad \text{frictional contact virtual work}$$

weak normal contact contribution

$$+ \int_{\gamma_c^1} \mathring{\underline{\boldsymbol{g}}}_\tau \delta \lambda_\tau \, d\gamma \tag{2.63}$$

weak tangential contact contribution.

The contact virtual work (2.61) is expressed as a summation of the contact contribution to the virtual work resulting from the variations of the positions (2.62), the weak contribution of normal contact constraints from the variation of the normal Lagrange multiplier (frictional contact virtual work) and the weak contribution of tangential contact constraints from the variation of the tangential Lagrange multiplier (2.63). Note that the variations and derivatives of piece-wise smooth pseudo potentials l_n and l_τ can be seen as sub-derivatives [Pietrzak and Curnier, 1997]. We recall that the variation of $\hat{p}_n$ is not required. For the frictionless case, the

possible statuses are "in contact" and "no contact" and for the frictional case they are "stick", "slip" and "no contact". For the frictional case the contact surface γ_c^1 is divided into three distinct sub surfaces:

$$\gamma_c^1 = \gamma_{\text{slip}} \cup \gamma_{\text{slip}} \cup \gamma_{\text{nc}}.$$

Based on the contact statuses, the contact contribution to the virtual work can be split as follows for frictionless and frictional cases:

$$\delta W_c = \begin{cases} \int\limits_{\gamma_c^1} \hat{\lambda}_n \delta g_n + g_n \delta \lambda_n \, d\gamma, & \hat{\lambda}_n \leq 0 \text{ (in contact)}, \\ \int\limits_{\gamma_c^1} -\dfrac{1}{\varepsilon_n} \lambda_n \delta \lambda_n \, d\gamma, & \hat{\lambda}_n > 0. \end{cases} \tag{2.64}$$

$$\delta W_c = \begin{cases} \displaystyle\int\limits_{\gamma_{\text{stick}}} \left[\hat{\lambda}_n \delta g_n + g_n \delta \lambda_n + \hat{\underline{\boldsymbol{\lambda}}}_\tau \delta \mathring{\underline{\boldsymbol{g}}}_\tau + \mathring{\underline{\boldsymbol{g}}}_\tau \delta \hat{\underline{\boldsymbol{\lambda}}}_\tau \right] d\gamma, \\ \qquad\qquad \hat{\lambda}_n \leq 0, ||\hat{\underline{\boldsymbol{\lambda}}}_\tau|| < -\mu \hat{\lambda}_n \text{ (stick)}, \\ \displaystyle\int\limits_{\gamma_{\text{slip}}} \left[\hat{\lambda}_n \delta g_n + g_n \delta \lambda_n - \mu \hat{p}_n \frac{\hat{\underline{\boldsymbol{\lambda}}}_\tau}{||\hat{\underline{\boldsymbol{\lambda}}}_\tau||} \delta \mathring{\underline{\boldsymbol{g}}}_\tau - \frac{1}{\varepsilon_\tau} \left(\hat{\underline{\boldsymbol{\lambda}}}_\tau + \mu \hat{p}_n \frac{\hat{\underline{\boldsymbol{\lambda}}}_\tau}{||\hat{\underline{\boldsymbol{\lambda}}}_\tau||} \right) \delta \hat{\underline{\boldsymbol{\lambda}}}_\tau \right] d\gamma, \\ \qquad\qquad \hat{\lambda}_n \leq 0, ||\hat{\underline{\boldsymbol{\lambda}}}_\tau|| \geq -\mu \hat{\lambda}_n \text{ (slip)}, \\ \displaystyle\int\limits_{\gamma_{\text{nc}}} \left[-\frac{1}{\varepsilon_n} \lambda_n \delta \lambda_n - \frac{1}{\varepsilon_\tau} \underline{\boldsymbol{\lambda}}_\tau \delta \underline{\boldsymbol{\lambda}}_\tau \right] d\gamma, \\ \qquad\qquad \hat{\lambda}_n > 0. \end{cases} \tag{2.65}$$

2.2.3 Tying of overlapping domains

We consider two open domains Ω^1 and Ω^2 with an overlap region $\Omega^1 \cap \Omega^2 \neq \varnothing$ [Fig. 2.7(a)]. Solid Ω^1 has only outer surfaces which are split into Dirichlet, Neumann and tying boundaries $\Gamma_u^1, \Gamma_t^1, \Gamma_g^1$, respectively, which are such that $\Gamma_u^1 \cup \Gamma_t^1 \cup \Gamma_g^1 = \partial\Omega^1$. We refer to the domain Ω^2 as the "host" domain as it hosts the partially embedded domain Ω^1. In addition to outer boundaries (the Dirichlet boundary Γ_u^2 and Neumann boundary Γ_t^2), the host domain has the embedded[2] boundary $\tilde{\Gamma}_g^2 = \Gamma_g^1 \cap \Omega^2$. We assume that two solids are glued together along the interface formed by the boundaries $\tilde{\Gamma}_g^2$ and Γ_g^1; the physics in the overlap zone is determined by solid Ω^1. Therefore, in the reference configuration $\tilde{\Gamma}_g^2 = \Gamma_g^1$ and must remain so in any configuration.

[2]We adapt the tilde $\tilde{\bullet}$ notations for representing all the quantities concerning the embedded surfaces within the host domain.

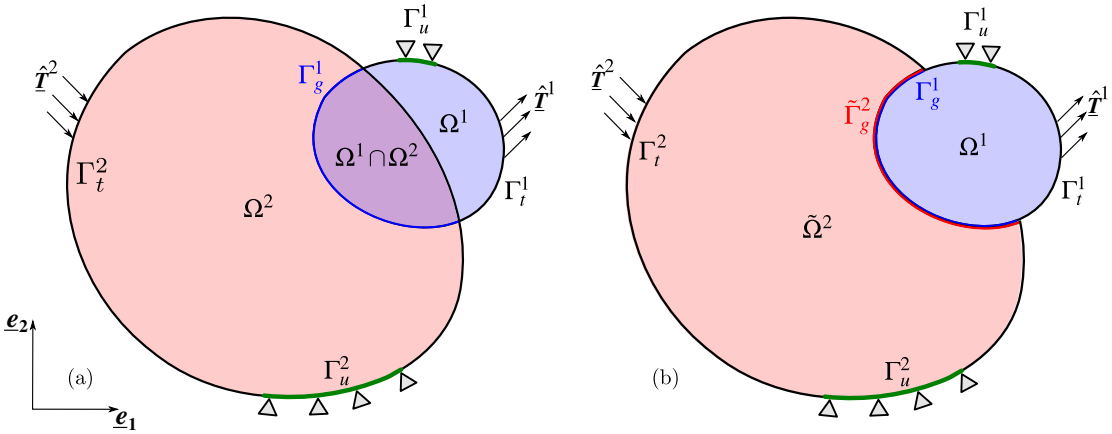


Figure 2.7: (a) Continuum setting of the two overlapping domains with applied boundary conditions; (b) an equivalent continuum problem without overlap.

Strong form

The constrained BVP for the tying problem of overlapping domains, ensuring the displacement continuity along the embedded interface Γ_g^1 is:

$$\nabla \cdot \underline{\underline{\boldsymbol{P}}}^1 + \underline{\boldsymbol{B}}^1 = 0 \qquad \text{in } \Omega^1 \tag{2.66}$$

$$\nabla \cdot \underline{\underline{\boldsymbol{P}}}^2 + \underline{\boldsymbol{B}}^2 = 0 \qquad \text{in } \tilde{\Omega}^2 \tag{2.67}$$

$$\underline{\underline{\boldsymbol{P}}}^i \cdot \underline{\boldsymbol{N}}^i = \hat{\underline{\boldsymbol{T}}}^i \qquad \text{on } \Gamma_t^i, \tag{2.68}$$

$$\underline{\boldsymbol{u}}^i = \hat{\underline{\boldsymbol{u}}}^i \qquad \text{on } \Gamma_u^i, \tag{2.69}$$

$$\underline{\boldsymbol{u}}^1 = \underline{\boldsymbol{u}}^2 \qquad \text{on } \Gamma_g^1. \tag{2.70}$$

where $\tilde{\Omega}^2 = \Omega^2 \setminus \bar{\Omega}^1$ is the effective non-overlapping region of the host domain volume [Fig. 2.7(b)], where the bar-notation denotes the open domain united with its closure, i.e. $\bar{\Omega}^1 = \Omega^1 \cup \partial\Omega^1$. The advantage of such formulation involving tying between overlapping domains lies in the greater flexibility of the associated discretized problem.

Weak form

The virtual work of the structural part given in (2.33) has to be modified to account for the effective volume of the host domain $\tilde{\Omega}^2$ only and the effective continuum domain $(\Omega^1 \cup \tilde{\Omega}^2)$ respectively:

$$\delta \tilde{W}_s = \underbrace{\int_{\Omega^1} \underline{\underline{\boldsymbol{P}}}^1 : \delta \underline{\underline{\boldsymbol{E}}}^1 \, d\Omega + \int_{\tilde{\Omega}^2} \underline{\underline{\boldsymbol{P}}}^2 : \delta \underline{\underline{\boldsymbol{E}}}^2 \, d\Omega}_{\delta W_{\text{int}}} - \underbrace{\int_{\Omega^1 \cup \tilde{\Omega}^2} \underline{\boldsymbol{B}}^i \cdot \delta \underline{\boldsymbol{u}}^i \, d\Omega - \int_{\Gamma_t^i} \hat{\underline{\boldsymbol{T}}}^i \cdot \delta \underline{\boldsymbol{u}}^i \, d\Gamma}_{\delta W_{\text{ext}}} . \tag{2.71}$$

Similar to the non-overlapping tying problem, the Lagrange multiplier method is applied here to enforce the displacement continuity along Γ_g^1.

$$\delta \mathcal{L}(\underline{\boldsymbol{u}}, \underline{\boldsymbol{\lambda}}) = \delta \tilde{W}_s + \delta \tilde{W}_g = \delta \tilde{W}_s + \int_{\Gamma_g^1} \left[\underline{\boldsymbol{\lambda}} \cdot \delta \underline{\tilde{\boldsymbol{g}}} + \delta \underline{\boldsymbol{\lambda}} \cdot \underline{\tilde{\boldsymbol{g}}} \right] d\Gamma = 0. \tag{2.72}$$

2.2.4 Contact between an outer and an internal/embedded surfaces

Fig. 2.8(b), shows the classical continuum contact setting where the solids come in contact along the domain-fitted boundary. Here we propose an equivalent, from continuum point of view, setting where contact boundary γ_c^1 is still an outer boundary of ω^1, while the homologue deformable body is now $\tilde{\omega}^2$ and its contact boundary $\tilde{\gamma}_c^2$ is embedded in the domain ω^2, whose part $\omega^2 \backslash \tilde{\omega}^2$ is discarded from the computation. This setting of contact along embedded boundaries is presented for the two dimensional case only. Its extension to 3D case is straightforward for the continuum formulation, but becomes complicated in a discrete setting. In contrast to the tying of overlapping domains formulated on the reference configuration, the contact is formulated on the current configuration.

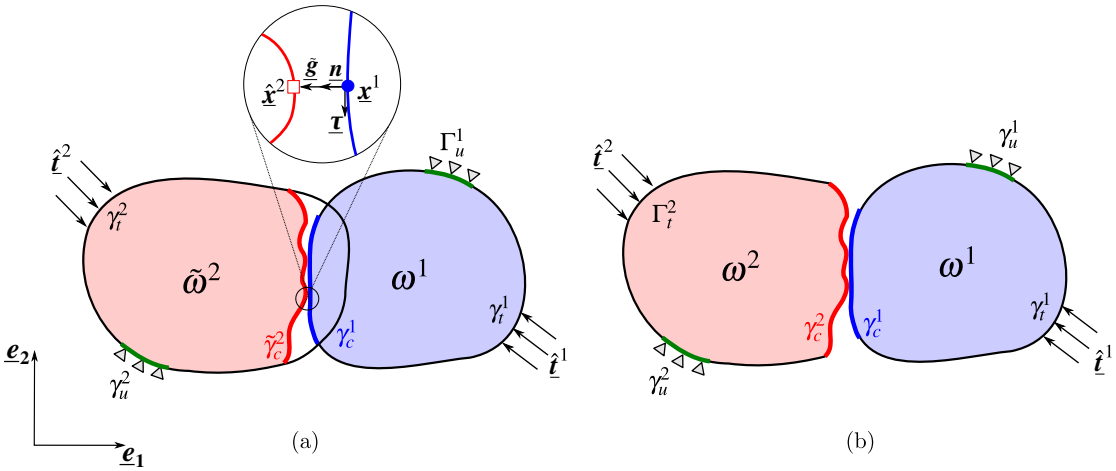


Figure 2.8: (a) Continuum setting of contact between a domain-fitted boundary and an embedded boundary $\tilde{\gamma}_c^2$; (b) an equivalent continuum problem with contact between domain-fitted boundaries.

The contact related kinematicquantities are slightly modified as a result of introducing embedded boundary in the domain. The gap vector is defined as a vector pointing from a point on the domain-fitted boundary γ_c^1 to its projection along the local normal onto the embedded surface $\tilde{\gamma}_c^2$:

$$\tilde{\underline{\boldsymbol{g}}}(\underline{\boldsymbol{x}}^1, \tilde{\gamma}_c^2) = \underline{\boldsymbol{x}}^1 - \hat{\underline{\boldsymbol{x}}}^2(\underline{\boldsymbol{x}}^1, \tilde{\gamma}_c^2), \tag{2.73}$$

where $\hat{\underline{\boldsymbol{x}}}^2$ is the projection of $\underline{\boldsymbol{x}}^1$ onto the embedded surface $\tilde{\gamma}_c^2$. The gap function is thus obtained as a dot product of the gap vector $\tilde{\underline{\boldsymbol{g}}}$ (2.73) and the outward unit normal $\underline{\boldsymbol{n}}$ to the surface γ_c^1.

$$\tilde{g}_n(\underline{\boldsymbol{x}}^1, \tilde{\gamma}_c^2) = \tilde{\underline{\boldsymbol{g}}} \cdot \underline{\boldsymbol{n}}. \tag{2.74}$$

The incremental slip is given as:

$$\overset{\circ}{\tilde{g}}_\tau(\underline{\boldsymbol{x}}^1) = \underline{\boldsymbol{\tau}}(\underline{\boldsymbol{x}}^1) \cdot \left(\overset{\circ}{\underline{\boldsymbol{x}}}^1 - \overset{\circ}{\hat{\underline{\boldsymbol{x}}}}^2(\underline{\boldsymbol{x}}^1) \right), \tag{2.75}$$

where $\underline{\boldsymbol{\tau}}(\underline{\boldsymbol{x}}^1)$ is a unit tangential vector to γ_c^1 at point $\underline{\boldsymbol{x}}^1$. In comparison to the vector valued incremental slip in (2.38) in a three dimensional setting, here

in (2.75) is simply a scalar value. With these notations at hand, the contact traction in 2D can be decomposed into normal and tangential contributions:

$$\underline{\boldsymbol{t}}_c^1 = p_n \underline{\boldsymbol{n}} + p_\tau \underline{\boldsymbol{\tau}}. \tag{2.76}$$

The normal (2.42) and tangential (2.43) contact conditions for the two dimensional embedded case remain the same:

$$\tilde{g}_n \geq 0, \quad p_n \leq 0, \quad p_n \tilde{g}_n = 0. \tag{2.77}$$

$$|\mathring{\tilde{g}}_\tau| \geq 0, \quad |p_\tau| - \mu |p_n| \leq 0, \quad (\mu |p_n| - |p_\tau|)\, |\mathring{\tilde{g}}_\tau| = 0. \tag{2.78}$$

Strong form

The BVP for the two body contact problem along embedded surfaces, taking into account the contact constraints (2.77),(2.78), reads as:

$$\nabla \cdot \underline{\underline{\boldsymbol{\sigma}}}^1 + \underline{\boldsymbol{b}}^1 = 0 \qquad \text{in } \omega^1 \tag{2.79}$$

$$\nabla \cdot \underline{\underline{\boldsymbol{\sigma}}}^2 + \underline{\boldsymbol{b}}^2 = 0 \qquad \text{in } \tilde{\omega}^2 \tag{2.80}$$

$$\underline{\underline{\boldsymbol{\sigma}}}^i \cdot \underline{\boldsymbol{n}}^i = \hat{\underline{\boldsymbol{t}}}^i \qquad \text{on } \gamma_t^i, \tag{2.81}$$

$$\underline{\boldsymbol{u}}^i = \hat{\underline{\boldsymbol{u}}}^i \qquad \text{on } \gamma_u^i, \tag{2.82}$$

$$\underline{\underline{\boldsymbol{\sigma}}}^1 \cdot \underline{\boldsymbol{n}}^1 = \underline{\boldsymbol{t}}_c^1 \qquad \text{on } \gamma_c^1, \tag{2.83}$$

$$\underline{\underline{\boldsymbol{\sigma}}}^2 \cdot \underline{\boldsymbol{n}}^2 = \underline{\boldsymbol{t}}_c^2 \qquad \text{on } \tilde{\gamma}_c^2. \tag{2.84}$$

Weak form

Similar to the domain tying problem, the virtual work of the structural part given in (2.50) has to be modified to account for the effective volume of the host domain $\tilde{\omega}^2$ only:

$$\delta \tilde{W}_s = \underbrace{\int_{\omega^1} \underline{\underline{\boldsymbol{\sigma}}}^1 : \delta \underline{\underline{\boldsymbol{e}}}^1 \, d\omega + \int_{\tilde{\omega}^2} \underline{\underline{\boldsymbol{\sigma}}}^2 : \delta \underline{\underline{\boldsymbol{e}}}^2 \, d\omega}_{\delta W_{\text{int}}} - \underbrace{\int_{\omega^1 \cup \tilde{\omega}^2} \underline{\boldsymbol{b}}^i \cdot \delta \underline{\boldsymbol{u}}^i \, d\omega - \int_{\gamma_t^i} \hat{\underline{\boldsymbol{t}}}^i \cdot \delta \underline{\boldsymbol{u}}^i \, d\gamma}_{\delta W_{\text{ext}}}, \tag{2.85}$$

The contact constraints imposed between the domain-fitted boundary γ_c^1 and the embedded boundary $\tilde{\gamma}_c^2$ are resolved using the augmented Lagrangian scheme. The scalar fields of Lagrange multipliers λ_n, λ_τ are introduced. They are equivalent to the components of the contact traction vector resolved into contact pressure p_n and the tangential friction shear p_τ introduced in (2.76). The associated pseudo potentials in 2D setting are:

$$l_n(\tilde{g}_n, \lambda_n) = \begin{cases} \lambda_n \tilde{g}_n + \frac{\varepsilon_n}{2} \tilde{g}_n^2, & \hat{\lambda}_n \leq 0, \\ -\frac{1}{2\varepsilon_n} \lambda_n^2, & \hat{\lambda}_n > 0. \end{cases} \tag{2.86}$$

$$l_\tau(\mathring{\tilde{g}}_\tau, \lambda_\tau, \hat{p}_n) = \begin{cases} \left\{ \begin{array}{cc} \lambda_\tau \mathring{\tilde{g}}_\tau + \frac{\varepsilon_\tau}{2} \mathring{\tilde{g}}_\tau^2, & |\hat{\lambda}_\tau| \leq -\mu \hat{p}_n \\ -\frac{1}{2\varepsilon_\tau} \left(\lambda_\tau^2 + 2\mu \hat{p}_n |\hat{\lambda}_\tau| + \mu^2 \hat{p}_n^2 \right), & |\hat{\lambda}_\tau| > -\mu \hat{p}_n \end{array} \right\}, & \hat{\lambda}_n \leq 0 \\ -\frac{1}{2\varepsilon_\tau} \lambda_\tau^2, & \hat{\lambda}_n > 0. \end{cases} \tag{2.87}$$

Accordingly, the Lagrangian's (2.57), (2.58), and virtual works (2.64), (2.65) for the frictionless and frictional cases are also adopted for the two dimensional case involving contact along embedded boundaries as follows:
Frictionless:

$$\mathcal{L}^{\text{aug}} = \tilde{W}_s + \tilde{W}_c = \tilde{W}_s + \int_{\gamma_c^1} l_n(\tilde{g}_n, \lambda_n). \tag{2.88}$$

$$\delta\tilde{W}_c = \begin{cases} \displaystyle\int_{\gamma_c^1} \hat{\lambda}_n \delta\tilde{g}_n + \tilde{g}_n \delta\lambda_n \, d\gamma, & \hat{\lambda}_n \leq 0 \text{ (in contact)}, \\ \displaystyle\int_{\gamma_c^1} -\frac{1}{\varepsilon_n} \lambda_n \delta\lambda_n \, d\gamma, & \hat{\lambda}_n > 0. \end{cases} \tag{2.89}$$

Frictional:

$$\mathcal{L}^{\text{aug}} = \tilde{W}_s + \tilde{W}_c = \tilde{W}_s + \int_{\gamma_c^1} l_n(\tilde{g}_n, \lambda_n) + l_\tau(\mathring{\tilde{g}}_\tau, \lambda_\tau, \hat{p}_n) \, d\gamma. \tag{2.90}$$

$$\delta\tilde{W}_c = \begin{cases} \displaystyle\int_{\gamma_{\text{stick}}} \left[\hat{\lambda}_n \delta\tilde{g}_n + \tilde{g}_n \delta\lambda_n + \hat{\lambda}_\tau \delta\mathring{\tilde{g}}_\tau + \mathring{\tilde{g}}_\tau \delta\lambda_\tau \right] d\gamma, & \\ & \hat{\lambda}_n \leq 0, \, |\hat{\lambda}_\tau| \leq -\mu \hat{p}_n \text{ (stick)}, \\ \displaystyle\int_{\gamma_{\text{slip}}} \left[\hat{\lambda}_n \delta\tilde{g}_n + \tilde{g}_n \delta\lambda_n - \mu \hat{p}_n \text{sign}(\hat{\lambda}_\tau) \delta\mathring{\tilde{g}}_\tau - \frac{1}{\varepsilon_\tau} \left(\lambda_\tau - \mu \hat{p}_n \text{sign}(\hat{\lambda}_\tau) \right) \delta\lambda_\tau \right] d\gamma, & \\ & \hat{\lambda}_n \leq 0, \, |\hat{\lambda}_\tau| > -\mu \hat{p}_n \text{ (slip)}, \\ \displaystyle\int_{\gamma_{\text{nc}}} \left[-\frac{1}{\varepsilon_n} \lambda_n \delta\lambda_n - \frac{1}{\varepsilon_\tau} \lambda_\tau \delta\lambda_\tau \right] d\gamma, & \\ & \hat{\lambda}_n > 0. \end{cases} \tag{2.91}$$

2.3 Summary

In this chapter we have formulated the continuous strong and weak forms for the four classes of interface problems. The tying of non-overlapping domains [Section 2.2.1] and contact between outer surfaces of solids [Section 2.2.2] are presented in a general three-dimensional context. Whereas, tying of overlapping domains [Section 2.2.3], contact along embedded surfaces [Section 2.2.4] are presented (and subsequently implemented) only in a two-dimensional context. Note, however, that the difference in the formulation between the three and two dimensions is very subtle and is completely missing for the tying problem. On the contrary, the implementation of the two latter problems presents a really challenging task in three-dimensions and is not addressed in this thesis. The required ingredients for the formulation such as the derivation of kinematics quantities, and the modified Lagrangian functionals accounting for the imposed constraints are presented. This chapter equips us with the continuous set of equations describing the problem statement. In the subsequent chapters, the discretized versions of the introduced problems will be derived, and solved within the FEM framework.

Chapter 3

Classical mortar method

Résumé: Dans ce chapitre, la formulation de la méthode Mortar dans le cadre de la méthode des éléments finis est présentée. Les formes discrètes du travail virtuel pour le couplage des domaines qui ne se chevauchent pas, le contact sans frottement et les problèmes de contact frottant sont aussi dérivées. Les algorithmes numériques nécessaires à l'évaluation des intégrales résultant de ces dérivations seront également présentés. Outre la mise au point de la méthode de Mortar, les aspects de détection et de parallélisation du traitement sur des architectures parallèles à mémoire distribuée seront discutés. Quelques aspects intéressants du traitement de contact seront présentés sous forme d'études de cas simples. La performance des méthodes élaborées sera démontrée sur des cas test numériques.

In this chapter a classical mortar finite element discretization framework will be presented. Discrete forms of virtual work for the tying of non-overlapping domains, the frictionless contact and the frictional contact problems will be derived here. The numerical algorithms required for the evaluation of the resulting integrals will also be presented. Apart from elaborating the mortar discretization framework, the aspects of contact detection and parallel resolution will be discussed. Few interesting aspects of the contact treatment will be presented in the form of simple case studies. The performance of the elaborated methods will be demonstrated on selected numerical test cases.

3.1 Mortar framework

Here emphasis is laid on introducing the spatial discretizations employed within the mortar finite element discretization techniques for the interfaces. For this we use the subspaces $\boldsymbol{\mathcal{U}}^h$ ($\boldsymbol{\mathcal{U}}^h \subset \boldsymbol{\mathcal{U}}$) and $\boldsymbol{\mathcal{V}}^h$ ($\boldsymbol{\mathcal{V}}^h \subset \boldsymbol{\mathcal{V}}$), which are the approximations of the solution and test function spaces introduced in Equations (2.30),(2.31). The details of the underlying standard finite element discretization[1] can be found in the classical textbooks on FEM [Zienkiewicz et al., 1977, Bathe, 1995, Belytschko et al., 2013] and others.Here, we will present the tying and contact interface problems between incompatible meshes/discretizations using the mortar method.

Within the mortar discretization framework the interfaces are typically classified into mortar and non-mortar sides. In this work, the superscript "1" refers to

[1]Note that for brevity and simplicity hereinafter we preserve the same notations for discretized entities as were introduced in the continuous problem statement.

the mortar side of the interface and "2" to the non-mortar side. Without loss of generality, the choice of mortar and non-mortar sides is arbitrary. For a problem assumed in $\mathbb{R}^{dim}$, the resulting interfaces γ_c^i will be $\mathbb{R}^{dim-1}$ dimensional manifolds in $\mathbb{R}^{dim}$. Concerning the two classes of interfaces problems of tying and contact, the kinematic quantities of interest are the displacements and current interfacial nodal coordinates respectively. By virtue of iso-parametric elements employed in this work, the same shape functions are applied for both as:

$$\underline{\boldsymbol{x}}^1(\underline{\boldsymbol{\xi}}^1) = N_m^1(\underline{\boldsymbol{\xi}}^1)\underline{\boldsymbol{x}}_m^1, \quad \underline{\boldsymbol{x}}^2(\underline{\boldsymbol{\xi}}^2) = N_i^2(\underline{\boldsymbol{\xi}}^2)\underline{\boldsymbol{x}}_i^2, \tag{3.1}$$

$$\underline{\boldsymbol{u}}^1(\underline{\boldsymbol{\xi}}^1) = N_m^1(\underline{\boldsymbol{\xi}}^1)\underline{\boldsymbol{u}}_m^1, \quad \underline{\boldsymbol{u}}^2(\underline{\boldsymbol{\xi}}^2) = N_i^2(\underline{\boldsymbol{\xi}}^2)\underline{\boldsymbol{u}}_i^2, \quad m \in [1, \mathrm{M}],\ i \in [1, \mathrm{N}]. \tag{3.2}$$

$N_m^1(\underline{\boldsymbol{\xi}}^1)$ and $N_i^2(\underline{\boldsymbol{\xi}}^2)$ are the interpolation functions of mortar and non-mortar sides, respectively; whereas M and N is the number of nodes per segment of the mortar and non-mortar sides, respectively. Hereinafter, Einstein summation over repetitive index is used. The parameterizations $\underline{\boldsymbol{\xi}}^i$ in case of three dimensions can be equivalently expressed as as vector with components (ξ^i, η^i) and simply as scalar (ξ^i) in case of two dimensional problems. The Lagrange multiplier vector $\underline{\boldsymbol{\lambda}}$ (2.51) is interpolated using functions Φ_l:

$$\underline{\boldsymbol{\lambda}}(\underline{\boldsymbol{\xi}}^1) = \Phi_l(\underline{\boldsymbol{\xi}}^1)\underline{\boldsymbol{\lambda}}^l, \quad l \in [1, \mathrm{L}], \tag{3.3}$$

where L is the number of nodes used for interpolation over every edge, where L can be less than or equal to M. Similar to the primal displacement interpolation functions chosen from a finite dimensional subspace, the Lagrange multiplier interpolation functions are chosen from the subspace $\boldsymbol{\mathcal{W}}^h$ ($\boldsymbol{\mathcal{W}}^h \subset \boldsymbol{\mathcal{W}}$). This particular choice is guided by the requirement of the mixed formulations to fulfill the inf-sup stability conditions [Bathe, 2001]. Note that the mortar side of the interface carries the dual degrees of freedom (DoFs) the Lagrange multipliers, and all the interface related evaluations of virtual work are carried out on the mortar side of the interface. Also though out the thesis, the primal and dual DoFs of the mortar side are accompanied by the superscript m and l, respectively, and for the primal DoFs on the non-mortar side accompanied by superscript i.

3.1.1 Mortar interface element

A mortar interface element is formed between segments (1D line elements in case of two dimension and 2D surface elements in case of three dimension) of the mortar and non-mortar sides [see Fig. 3.1]. Each contact interface element consists of (M + N) nodes, each with "dim" (spatial dimension of the problem) primal DoFs in 2D, and of L dual DoFs associated with Lagrange multipliers. Accounting for the accurate areas of contact between the discretized mortar and non-mortar surfaces, requires a virtual sub-discretization. This is achieved by projections in 2D and by projections and clipping in 3D (see Section 3.4). The sub-discretized mortar side representing the accurate areas of contact will be referred to as mortar domain (S^{el}). Moreover, for the sake of simplicity, we omitted averaging of normals at nodes and their interpolation, which was elaborated in [Yang et al., 2005, Popp, 2012]. The normal averaging has many numerical advantages, but complicates the implementation.

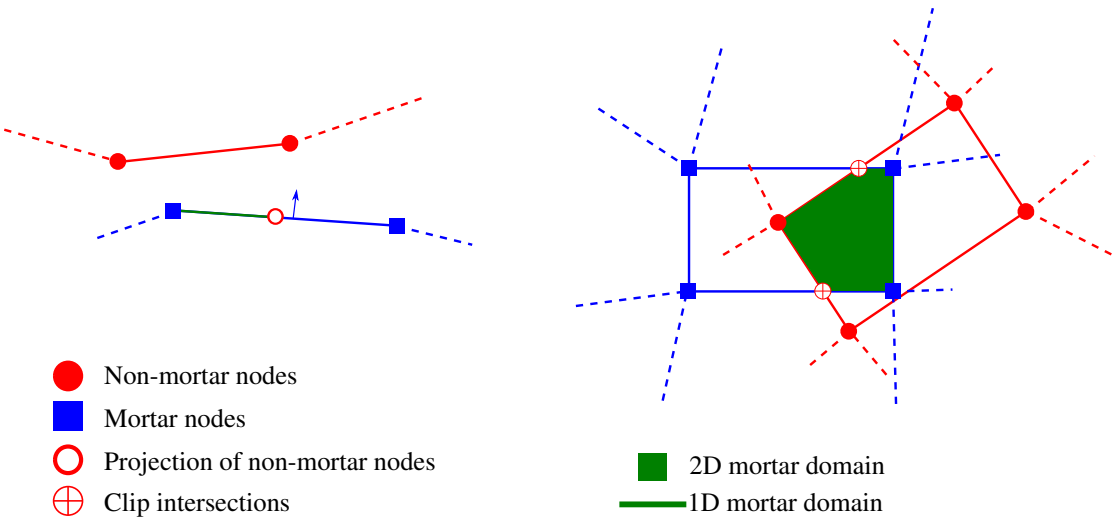


Figure 3.1: Mortar domain (S^{el}) in 2D and 3D.

3.2 Discrete mesh tying problem

3.2.1 Discrete virtual work

Substituting the displacement (3.2) and Lagrange multiplier interpolations (3.3) into the weak form (2.34)

$$\delta W_g^{\text{el}} = \left(\underline{\boldsymbol{\lambda}}^l \cdot \delta\underline{\boldsymbol{u}}^m + \delta\underline{\boldsymbol{\lambda}}^l \cdot \underline{\boldsymbol{u}}^m\right) D_{lm} - \left(\underline{\boldsymbol{\lambda}}^l \cdot \delta\underline{\boldsymbol{u}}^i + \delta\underline{\boldsymbol{\lambda}}^l \cdot \underline{\boldsymbol{u}}^i\right) M_{li}, \tag{3.4}$$

$$l \in [1, \mathrm{L}],\ i \in [1, \mathrm{N}],\ m \in [1, \mathrm{M}]$$

where D_{lm} and M_{li} are the integrals evaluated over the mortar domain S^{el}

$$D_{lm} = \int_{S^{\text{el}}} \Phi_l^1(\underline{\boldsymbol{\xi}}^1) N_m^1(\underline{\boldsymbol{\xi}}^1)\, d\Gamma, \tag{3.5}$$

$$M_{li} = \int_{S^{\text{el}}} \Phi_l^1(\underline{\boldsymbol{\xi}}^1) N_i^2(\underline{\boldsymbol{\xi}}^2)\, d\Gamma. \tag{3.6}$$

The evaluation of the integrals D_{lm} and M_{li} forms the core of the mortar discretization scheme. The used for their evaluations is detailed out in the Section 3.4. The nodal blocks of the mortar matrices denoted as $\mathbf{D}$ ($\mathrm{L} \times \mathrm{M}$) and $\mathbf{M}$ ($\mathrm{L} \times \mathrm{N}$) can be expressed as:

$$\mathbf{D}(l, m) = D_{lm}\underline{\underline{\boldsymbol{I}}}, \tag{3.7}$$

$$\mathbf{M}(l, i) = M_{li}\underline{\underline{\boldsymbol{I}}}, \tag{3.8}$$

where $\underline{\underline{\boldsymbol{I}}}$ is the identity tensor of the spatial dimension of the problem. Using the matrix notations, Eq. (3.4) reads

$$\delta W_g^{\text{el}} = \begin{bmatrix} \mathbf{D}^{\intercal} \cdot \mathbf{L} \\ -\mathbf{M}^{\intercal} \cdot \mathbf{L} \\ \mathbf{D} \cdot \mathbf{U}^1 - \mathbf{M} \cdot \mathbf{U}^2 \end{bmatrix}^{\intercal} \cdot \begin{bmatrix} \delta\mathbf{U}^1 \\ \delta\mathbf{U}^2 \\ \delta\mathbf{L} \end{bmatrix} \tag{3.9}$$

where arrays $\mathbf{U}^1, \mathbf{U}^2, \mathbf{L}$ store current values of associated nodal primal (on mortar and non-mortar sides) and dual (mortar) DoFs:

$$\mathbf{U}^1 = \left[\underline{\boldsymbol{u}}_1^1, \ldots, \underline{\boldsymbol{u}}_M^1\right]^{\intercal}, \quad \mathbf{U}^2 = \left[\underline{\boldsymbol{u}}_1^2, \ldots, \underline{\boldsymbol{u}}_N^2\right]^{\intercal}, \quad \mathbf{L} = \left[\underline{\boldsymbol{\lambda}}^1, \ldots, \underline{\boldsymbol{\lambda}}^L\right]^{\intercal},$$

whereas their variations are denoted $\delta\mathbf{U}^1, \delta\mathbf{U}^2, \delta\mathbf{L}$. The tangent operator for the mortar interface element is obtained by taking the derivatives of the vector from (3.9) with respect to its DoFs:

$$\mathbf{K} = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{D}^\intercal \\ \mathbf{0} & \mathbf{0} & -\mathbf{M}^\intercal \\ \mathbf{D} & -\mathbf{M} & 0 \end{bmatrix}. \tag{3.10}$$

This tangent operator has zero blocks for primal DoFs, non-zero blocks to link primal with dual DoFs, and zero blocks on the main diagonal, which is a typical structure for the saddle-point system.

3.3 Discrete contact problem

3.3.1 Discrete normal contact constraints

As a first step towards the elaboration of the contact part of the virtual work, the interpolations (3.1) and (3.3) are inserted in the continuous weak form representing the normal contact condition (frictional contact virtual work). First, we focus on the derivation of a single term of this integral, which enables us to introduce concise notations and simplifications to be used for the other terms. It is obtained for every active contact element by integrating over the mortar domain S^{el}.

$$\int_{\gamma_c^1} g_n \delta\lambda_n \, d\gamma \approx \int_{S^{\mathrm{el}}} g_n \delta\lambda_n \, d\gamma = \int_{S^{\mathrm{el}}} \left(\Phi_l(\underline{\boldsymbol{\xi}}^1) N_m^1(\underline{\boldsymbol{\xi}}^1) \underline{\boldsymbol{x}}_m^1 - \Phi_l(\underline{\boldsymbol{\xi}}^1) N_i^2(\underline{\boldsymbol{\xi}}^2) \underline{\boldsymbol{x}}_i^2 \right) \cdot \underline{\boldsymbol{n}} \, \delta\lambda_n^l \, d\gamma = \tag{3.11}$$

$$= \underline{\boldsymbol{n}} \cdot \left[\left(\int_{S^{\mathrm{el}}} \Phi_l(\underline{\boldsymbol{\xi}}^1) N_m^1(\underline{\boldsymbol{\xi}}^1) \, d\gamma \right) \underline{\boldsymbol{x}}_m^1 - \left(\int_{S^{\mathrm{el}}} \Phi_l(\underline{\boldsymbol{\xi}}^2) N_i^2(\underline{\boldsymbol{\xi}}^2) \, d\gamma \right) \underline{\boldsymbol{x}}_i^2 \right] \delta\lambda_n^l. \tag{3.12}$$

where $\underline{\boldsymbol{n}}$ is the unit outward normal of the mortar segment. The definition of the integral gap for a mortar node l is then given as:

$$\bar{\underline{\boldsymbol{g}}}^l = \left[D_{lm} \underline{\boldsymbol{x}}_m^1 - M_{li} \underline{\boldsymbol{x}}_i^2 \right], \tag{3.13}$$

where D_{lm} and M_{li} are the mortar integrals given by:

$$D_{lm} = \int_{S^{\mathrm{el}}} \Phi_l^1(\underline{\boldsymbol{\xi}}^1) N_m^1(\underline{\boldsymbol{\xi}}^1) \, d\gamma, \tag{3.14}$$

$$M_{li} = \int_{S^{\mathrm{el}}} \Phi_l^1(\underline{\boldsymbol{\xi}}^1) N_i^2(\underline{\boldsymbol{\xi}}^2) \, d\gamma. \tag{3.15}$$

Using the introduced notations, the continuous weak form representing the normal contact condition (frictional contact virtual work) takes a simple form:

$$\int_{S^{\mathrm{el}}} g_n \delta\lambda_n \, d\gamma = \bar{g}_n^l \delta\lambda_n^l, \tag{3.16}$$

where

$$\bar{g}_n^l = \bar{\underline{\boldsymbol{g}}}^l \cdot \underline{\boldsymbol{n}} \tag{3.17}$$

is the integral normal gap. The variation of the integral normal gap is given by [Puso and Laursen, 2004]:

$$\delta\bar{g}_n^l = \left[D_{lm}\delta\underline{\boldsymbol{x}}_m^1 - M_{li}\delta\underline{\boldsymbol{x}}_i^2\right]\cdot\underline{\boldsymbol{n}} + \underbrace{\left[D_{lm}\underline{\boldsymbol{x}}_m^1 - M_{li}\underline{\boldsymbol{x}}_i^2\right]\cdot\delta\underline{\boldsymbol{n}}}_{0}. \tag{3.18}$$

3.3.2 Discrete tangential contact constraints

In analogy, inserting the interpolations into the weak tangential contact condition (2.63)

$$\int_{\gamma_c^1}\mathring{\underline{\boldsymbol{g}}}_\tau\cdot\delta\underline{\boldsymbol{\lambda}}_\tau\,d\gamma \approx \int_{S^{\text{el}}}\mathring{\underline{\boldsymbol{g}}}_\tau\cdot\delta\underline{\boldsymbol{\lambda}}_\tau\,d\gamma = \int_{S^{\text{el}}}\left(\Phi_l(\underline{\boldsymbol{\xi}}^1)N_m^1(\underline{\boldsymbol{\xi}}^1)\mathring{\underline{\boldsymbol{x}}}_m^1 - \Phi_l(\underline{\boldsymbol{\xi}}^1)N_i^2(\underline{\boldsymbol{\xi}}^2)\mathring{\underline{\boldsymbol{x}}}_i^2\right)\cdot\delta\underline{\boldsymbol{\lambda}}_\tau^l\,d\gamma = \tag{3.19}$$

$$= (\underline{\underline{\boldsymbol{I}}} - \underline{\boldsymbol{n}}\otimes\underline{\boldsymbol{n}})\cdot\left[\left(\int_{S^{\text{el}}}\Phi_l(\underline{\boldsymbol{\xi}}^1)N_m^1(\underline{\boldsymbol{\xi}}^1)\,d\gamma\right)\mathring{\underline{\boldsymbol{x}}}_m^1 - \left(\int_{S^{\text{el}}}\Phi_l(\underline{\boldsymbol{\xi}}^2)N_i^2(\underline{\boldsymbol{\xi}}^1)\,d\gamma\right)\mathring{\underline{\boldsymbol{x}}}_i^2\right]\cdot\delta\underline{\boldsymbol{\lambda}}^l. \tag{3.20}$$

Eq. (3.20) is obtained by substituting (2.53) and grouping terms of (3.19). Using the concise mortar integral notations, the integral incremental slip for a mortar node l is:

$$\mathring{\underline{\boldsymbol{g}}}_\tau^l = (\underline{\underline{\boldsymbol{I}}} - \underline{\boldsymbol{n}}\otimes\underline{\boldsymbol{n}})\cdot\left[D_{lm}\mathring{\underline{\boldsymbol{x}}}_m^1 - M_{li}\mathring{\underline{\boldsymbol{x}}}_i^2\right]. \tag{3.21}$$

We recall that the notation $\mathring{(\cdot)}$ is used for the incremental form given as:

$$\mathring{(\cdot)} = \frac{d(\cdot)}{dt}\Delta t. \tag{3.22}$$

Frame-indifferent measure

The definition in Eq. 3.21 for the incremental slip lacks invariance, i.e. $\mathring{\underline{\boldsymbol{x}}} = 0$ is not ensured for rigid body motions. Therefore an alternative approach similar to the one proposed in [Puso and Laursen, 2004]is used here. An incremental form of the integral gap (3.13) is used for this purpose. In case of flat surfaces that remain in contact $\mathring{\underline{\boldsymbol{g}}}^l = 0$ (for curved surfaces $\mathring{\underline{\boldsymbol{g}}}^l \approx 0$).

$$\mathring{\underline{\boldsymbol{g}}}^l = \left[D_{lm}\mathring{\underline{\boldsymbol{x}}}_m^1 - M_{li}\mathring{\underline{\boldsymbol{x}}}_i^2\right] + \left[\mathring{D}_{lm}\underline{\boldsymbol{x}}_m^1 - \mathring{M}_{li}\underline{\boldsymbol{x}}_i^2\right] = 0. \tag{3.23}$$

$$\Longrightarrow \left[D_{lm}\mathring{\underline{\boldsymbol{x}}}_m^1 - M_{li}\mathring{\underline{\boldsymbol{x}}}_i^2\right] = -\left[\mathring{D}_{lm}\underline{\boldsymbol{x}}_m^1 - \mathring{M}_{li}\underline{\boldsymbol{x}}_i^2\right] \tag{3.24}$$

The left side of (3.24) resembles the non-invariant measure in (3.21). Since integrals D_{lm} and M_{li} are invariant under rigid body motion, their incremental change is zero, i.e. $\mathring{D}_{lm} = 0$ and $\mathring{M}_{li} = 0$ under such motion. Thus the right hand side of (3.24) is taken as the frame invariant measure of the incremental slip. Projecting the right hand side of (3.24) on to the tangential plane, the nodal integral incremental slip can be written as:

$$\mathring{\underline{\boldsymbol{g}}}_\tau^l = -(\underline{\underline{\boldsymbol{I}}} - \underline{\boldsymbol{n}}\otimes\underline{\boldsymbol{n}})\cdot\left[\mathring{D}_{lm}\underline{\boldsymbol{x}}_m^1 - \mathring{M}_{li}\underline{\boldsymbol{x}}_l^2\right]. \tag{3.25}$$

Using the backward Euler time discretization, the incremental form can be equivalently represented as:

$$\overset{\circ}{(\cdot)} = \frac{d(\cdot)}{dt}\Delta t = (\cdot)(t_j) - (\cdot)(t_{j-1}). \tag{3.26}$$

Expanding (3.25) we get:

$$\overset{\circ}{\underline{\boldsymbol{g}}}{}^l_\tau = -(\underline{\underline{\boldsymbol{I}}} - \underline{\boldsymbol{n}} \otimes \underline{\boldsymbol{n}}) \cdot \left[\left(D_{lm}(t_j) - D_{lm}(t_{j-1}) \right) \underline{\boldsymbol{x}}^1_m(t_{j-1}) - \left(M_{li}(t_j) - M_{li}(t_{j-1}) \right) \underline{\boldsymbol{x}}^2_i(t_{j-1}) \right]. \tag{3.27}$$

Eq. 3.27 is used for the incremental slip. Substituting (3.26) in (3.21) we get:

$$\overset{\circ}{\underline{\boldsymbol{g}}}{}^l_\tau = (\underline{\underline{\boldsymbol{I}}} - \underline{\boldsymbol{n}} \otimes \underline{\boldsymbol{n}}) \cdot \left[D_{lm}(\underline{\boldsymbol{x}}^1_m(t_{j-1}) - \underline{\boldsymbol{x}}^1_m(t_j)) - M_{li}(\underline{\boldsymbol{x}}^2_i(t_{j-1}) - \underline{\boldsymbol{x}}^2_i(t_j)) \right]. \tag{3.28}$$

Taking variation of (3.28) and neglecting variations of quantities from previous time increment $(\cdot)(t_{j-1})$, we get [Puso and Laursen, 2004]:

$$\delta\overset{\circ}{\underline{\boldsymbol{g}}}{}^l_\tau = (\underline{\underline{\boldsymbol{I}}} - \underline{\boldsymbol{n}} \otimes \underline{\boldsymbol{n}}) \cdot \left[D_{lm}\delta\underline{\boldsymbol{x}}^1_m - M_{li}\delta\underline{\boldsymbol{x}}^2_i \right]. \tag{3.29}$$

3.3.3 Discrete contact virtual work

Within the ALM formulation all the interface contact elements contribute to the system virtual work, irrespective of the contact status (active or inactive). This feature of the ALM results in a smoother energy potential and continuity of virtual work. The discretized form of the virtual work for every contact elementfor the frictionless case is:

$$\delta W^{\text{el}}_{\text{c}} = \begin{bmatrix} \delta\underline{\boldsymbol{x}}^1_m \\ \delta\underline{\boldsymbol{x}}^2_i \\ \delta\lambda^l_n \end{bmatrix}^{\intercal} \cdot \begin{bmatrix} \left(\hat{\lambda}^l_n \underline{\boldsymbol{n}}\right) D_{lm} \\ -\left(\hat{\lambda}^l_n \underline{\boldsymbol{n}}\right) M_{li} \\ \bar{g}^l_n \end{bmatrix}, \qquad \hat{\lambda}^l_n \leq 0 \,(\text{in contact}), \tag{3.30}$$

$$\delta W^{\text{el}}_{\text{nc}} = \begin{bmatrix} \delta\underline{\boldsymbol{x}}^1_m \\ \delta\underline{\boldsymbol{x}}^2_i \\ \delta\lambda^l_n \end{bmatrix}^{\intercal} \cdot \begin{bmatrix} 0 \\ 0 \\ -\frac{1}{\varepsilon_n}\lambda^l_n \end{bmatrix}, \qquad \hat{\lambda}^l_n > 0 \,(\text{no contact}). \tag{3.31}$$

In the frictional case, the discretized form of the virtual work for every contact elementcan be obtained for three different contact statuses, namely the stick, slip and no contact as follows:

$$\delta W^{\text{el}}_{\text{stick}} = \begin{bmatrix} \delta\underline{\boldsymbol{x}}^1_m \\ \delta\underline{\boldsymbol{x}}^2_i \\ \delta\underline{\boldsymbol{\lambda}}^l \end{bmatrix}^{\intercal} \cdot \begin{bmatrix} \left(\hat{\lambda}^l_n \underline{\boldsymbol{n}} + \hat{\underline{\boldsymbol{\lambda}}}^l_\tau\right) D_{lm} \\ -\left(\hat{\lambda}^l_n \underline{\boldsymbol{n}} + \hat{\underline{\boldsymbol{\lambda}}}^l_\tau\right) M_{li} \\ \bar{g}^l_n \underline{\boldsymbol{n}} + \overset{\circ}{\underline{\boldsymbol{g}}}{}^l_\tau \end{bmatrix}, \qquad \hat{\lambda}^l_n \leq 0\,, ||\hat{\underline{\boldsymbol{\lambda}}}^l_\tau|| \leq -\mu\hat{\lambda}^l_n \,(\text{stick}), \tag{3.32}$$

$$\delta W^{\text{el}}_{\text{slip}} = \begin{bmatrix} \delta\underline{\boldsymbol{x}}^1_m \\ \delta\underline{\boldsymbol{x}}^2_i \\ \delta\underline{\boldsymbol{\lambda}}^l \end{bmatrix}^{\intercal} \cdot \begin{bmatrix} \left(\hat{\lambda}^l_n \underline{\boldsymbol{n}} - \mu\hat{\lambda}^l_n \frac{\hat{\underline{\boldsymbol{\lambda}}}^l_\tau}{||\hat{\underline{\boldsymbol{\lambda}}}^l_\tau||}\right) D_{lm} \\ -\left(\hat{\lambda}^l_n \underline{\boldsymbol{n}} - \mu\hat{\lambda}^l_n \frac{\hat{\underline{\boldsymbol{\lambda}}}^l_\tau}{||\hat{\underline{\boldsymbol{\lambda}}}^l_\tau||}\right) M_{li} \\ \bar{g}^l_n \underline{\boldsymbol{n}} - \frac{1}{\varepsilon_t}\left(\underline{\boldsymbol{\lambda}}^l_\tau + \mu\hat{\lambda}^l_n \frac{\hat{\underline{\boldsymbol{\lambda}}}^l_\tau}{||\hat{\underline{\boldsymbol{\lambda}}}^l_\tau||}\right) \end{bmatrix}, \qquad \hat{\lambda}^l_n \leq 0\,, ||\hat{\underline{\boldsymbol{\lambda}}}^l_\tau|| > -\mu\hat{\lambda}^l_n \,(\text{slip}), \tag{3.33}$$

$$\delta W_{\text{nc}}^{\text{el}} = \begin{bmatrix} \delta \underline{\boldsymbol{x}}_m^1 \\ \delta \underline{\boldsymbol{x}}_i^2 \\ \delta \underline{\boldsymbol{\lambda}}^l \end{bmatrix}^{\mathsf{T}} \cdot \begin{bmatrix} 0 \\ 0 \\ -\frac{1}{\varepsilon_n} \underline{\boldsymbol{\lambda}}^l \end{bmatrix}, \qquad \hat{\lambda}_n^l > 0 \,(\text{no contact}). \tag{3.34}$$

Note, the concise notations of variations of δg_n^l and $\delta \underline{\mathring{\boldsymbol{g}}}_\tau^l$ in Equations (2.64), (2.65) are replaced with the underlying variations of the nodal positions of the mortar and non-mortar segments given in (3.18) and (3.29). Note also that the status detection and the evaluation of the residual vector is based on the nodal integral quantities (3.17),(3.27):

$$\hat{\lambda}_n^l = \lambda_n^l + \varepsilon_n \bar{g}_n^l, \tag{3.35}$$

$$\underline{\hat{\boldsymbol{\lambda}}}_\tau^l = \underline{\boldsymbol{\lambda}}_\tau^l + \varepsilon_n \underline{\mathring{\boldsymbol{g}}}_\tau^l. \tag{3.36}$$

where ε_n, ε_τ are the augmentation parameters of the mortar interface element. In this thesis, we propose to assign values to these parameters based on the stiffness contributions from primal DoFs of mortar interface element. The local augmentation parameters for each mortar interface element is evaluated as:

$$\varepsilon_n = \varepsilon_\tau = \frac{k_{\text{mortar}}^{\text{seg}} k_{\text{non-mortar}}^{\text{seg}}}{k_{\text{mortar}}^{\text{seg}} + k_{\text{non-mortar}}^{\text{seg}}} \tag{3.37}$$

where $k_{\text{mortar}}^{\text{seg}}$ and $k_{\text{non-mortar}}^{\text{seg}}$ represent the stiffness contributions from the mortar and non-mortar segments that constitute the mortar interface element. Eq. 3.37 is problem dependent and can serve the purpose of an initial guess. A manual tuning like scaling for these parameters might be necessary to ensure better convergence. Also it could be beneficial to chose different augmentation parameters for the normal and tangential directions [Wriggers, 2012]. This strategy is applied selectively to some of the numerical examples [see Section 3.7] presented in this thesis.

The status check is done for every element independently. Unlike the linear mesh tying problem, the contact forces are evaluated on the unknown contact surface which constitutes the solution of the problem. This makes the contact problem non-linear in nature. However, the contact between conformal surfaces can be considered linear in absence of friction. Whereas, for friction problems even for conformal surfaces the problem remains non-linear as the stick and slip zones of contact are not known in advance. A generalized Newton-Raphson method is employed for the resulting system of non-linear equations. This procedure involves an adequate linearization needed to replace the actual non-linear problem with a set of linear equations, which can then be solved by the standard linear algebra tools. This approach is often referred to as the linear iteration approach. For the frictionless case, taking the second variation of the contact force vectors (in contact and no contact), we derive the expression for the tangent operator for the mortar contact element:

$$\mathbf{K}_c^{\text{el}} = \begin{bmatrix} \left(\hat{\lambda}_n^l \underline{\boldsymbol{n}}\right) \Delta D_{lm} + \Delta\left(\hat{\lambda}_n^l \underline{\boldsymbol{n}}\right) D_{lm} \\ -\left(\hat{\lambda}_n^l \underline{\boldsymbol{n}}\right) \Delta M_{li} - \Delta\left(\hat{\lambda}_n^l \underline{\boldsymbol{n}}\right) M_{li} \\ \Delta \bar{g}_{nj} \end{bmatrix}, \tag{3.38}$$

$$\mathbf{K}_c^{\mathrm{el}} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ -\frac{1}{\varepsilon_n}\mathbf{I} \end{bmatrix}. \tag{3.39}$$

For the frictional case based on the three possible contact statuses, the corresponding tangent operators are:

$$\mathbf{K}_{\mathrm{stick}}^{\mathrm{el}} = \begin{bmatrix} \left(\hat{\lambda}_n^l \underline{\boldsymbol{n}} + \hat{\underline{\boldsymbol{\lambda}}}_\tau^l\right)\Delta D_{lm} + \Delta\left(\hat{\lambda}_n^l \underline{\boldsymbol{n}} + \hat{\underline{\boldsymbol{\lambda}}}_\tau^l\right) D_{lm} \\ -\left(\hat{\lambda}_n^l \underline{\boldsymbol{n}} + \hat{\underline{\boldsymbol{\lambda}}}_\tau^l\right)\Delta M_{li} - \Delta\left(\hat{\lambda}_n^l \underline{\boldsymbol{n}} + \hat{\underline{\boldsymbol{\lambda}}}_\tau^l\right) M_{li} \\ \Delta(\bar{g}_n^l \underline{\boldsymbol{n}} + \mathring{\underline{\boldsymbol{g}}}_\tau^l) \end{bmatrix}, \tag{3.40}$$

$$\mathbf{K}_{\mathrm{slip}}^{\mathrm{el}} = \begin{bmatrix} \left(\hat{\lambda}_n^l \underline{\boldsymbol{n}} - \mu\hat{\lambda}_n^l \frac{\hat{\underline{\boldsymbol{\lambda}}}_\tau^l}{||\hat{\underline{\boldsymbol{\lambda}}}_\tau^l||}\right)\Delta D_{lm} + \Delta\left(\hat{\lambda}_n^l \underline{\boldsymbol{n}} - \mu\hat{\lambda}_n^l \frac{\hat{\underline{\boldsymbol{\lambda}}}_\tau^l}{||\hat{\underline{\boldsymbol{\lambda}}}_\tau^l||}\right) D_{lm} \\ -\left(\hat{\lambda}_n^l \underline{\boldsymbol{n}} - \mu\hat{\lambda}_n^l \frac{\hat{\underline{\boldsymbol{\lambda}}}_\tau^l}{||\hat{\underline{\boldsymbol{\lambda}}}_\tau^l||}\right)\Delta M_{li} - \Delta\left(\hat{\lambda}_n^l \underline{\boldsymbol{n}} - \mu\hat{\lambda}_n^l \frac{\hat{\underline{\boldsymbol{\lambda}}}_\tau^l}{||\hat{\underline{\boldsymbol{\lambda}}}_\tau^l||}\right) M_{li} \\ \Delta\bar{g}_n^l \underline{\boldsymbol{n}} - \frac{1}{\varepsilon_t}\Delta\left(\underline{\boldsymbol{\lambda}}_\tau^l + \mu\hat{\lambda}_n^l \frac{\hat{\underline{\boldsymbol{\lambda}}}_\tau^l}{||\hat{\underline{\boldsymbol{\lambda}}}_\tau^l||}\right) \end{bmatrix}, \tag{3.41}$$

$$\mathbf{K}_{\mathrm{nc}}^{\mathrm{el}} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ -\frac{1}{\varepsilon_n}\Delta\underline{\boldsymbol{\lambda}}^l \end{bmatrix}. \tag{3.42}$$

3.4 Mortar integrals

The evaluation of mortar integrals [(3.5), (3.6), (3.14), (3.15)] forms the central part of the mortar discretization method. Although the problems of tying and contact differ, the numerical procedures to evaluate the mortar integrals remain practically the same for both. In the following sections, we will recall the procedures introduced and used in [Puso, 2004, Yang et al., 2005, Popp, 2012].

3.4.1 Evaluation of integrals in 2D

The evaluation of the mortar side integrals D_{lm} is straightforward as it involves the product of shape functions from the mortar side only (3.5), (3.14). This is in contrast with the evaluation of the mortar integral M_{li} which combines shape functions from both the mortar and non-mortar sides [(3.6), (3.15)]. Its evaluation requires a mapping between them. For this purpose, the non-mortar nodes $\underline{\boldsymbol{x}}_i^2$ ($i \in [1, N]$) are projected onto the mortar segment along the mortar segment normal $\underline{\boldsymbol{n}}$. [see Fig. 3.2]. The local coordinates ξ_i^1 of the projections are found by solving

$$\left(N_m^1(\xi_i^1)\underline{\boldsymbol{x}}_m^1 - \underline{\boldsymbol{x}}_i^2\right) \times \underline{\boldsymbol{n}} = 0. \tag{3.43}$$

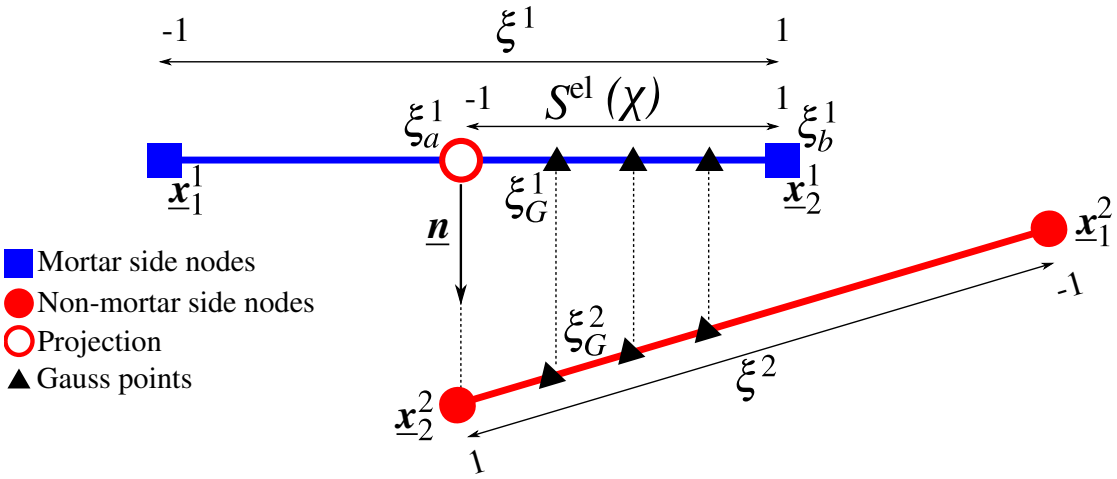


Figure 3.2: The nodes of the non-mortar segment are projected onto mortar segment, establishing a mapping and determining the mortar domain χ.

The limits of the mortar domain S^{el} are defined either by mortar segment nodes or the projection of non-mortar nodes. The mortar integrals are evaluated over the mortar domain S^{el}, and is parametrized by $\chi \in [-1;1]$. The projection coordinates $\xi^1_{a/b}$determine the limits of the mortar domain. The mapping between original parameterization ξ^1 and new parameterization χ is given by:

$$\xi^1(\chi) = \frac{1}{2}(1-\chi)\xi_a^1 + \frac{1}{2}(1+\chi)\xi_b^1. \tag{3.44}$$

To evaluate the integrals using Gauss quadrature, the mortar-side Gauss points ξ_G^1 are projected along mortar segment normal $\underline{\boldsymbol{n}}$ onto the non-mortar side and the corresponding local coordinates ξ_G^2 are determined by:

$$\left[N_i^2(\xi_G^2)\underline{\boldsymbol{x}}_i^2 - N_m^1(\xi_G^1)\underline{\boldsymbol{x}}_m^1\right] \times \underline{\boldsymbol{n}} = 0. \tag{3.45}$$

The mortar integrals, evaluated using the Gauss quadrature rule, take the following form:

$$D_{lm} = \int_{S^{\mathrm{el}}} \Phi_l(\xi^1) N_m^1(\xi^1)\, d\Gamma = \sum_{G=1}^{N_G} w_G \Phi_l(\xi_G^1) N_m^1(\xi_G^1) J_{\mathrm{seg}}(\xi_G^1), \tag{3.46}$$

$$M_{li} = \int_{S^{\mathrm{el}}} \Phi_l(\xi^1) N_i^2(\xi^2)\, d\Gamma = \sum_{G=1}^{N_G} w_G \Phi_l(\xi_G^1) N_i^2(\xi_G^2) J_{\mathrm{seg}}(\xi_G^1), \tag{3.47}$$

where w_G is Gauss integration weight and as previously, $l \in [1, L]$, $m \in [1, M]$, $i \in [1, N]$ and $G \in [1, N_G]$, where N_G is number of Gauss integration points, J_{seg} is the normalized Jacobian

$$J_{\mathrm{seg}}(\xi_G^1) = \left| \frac{\partial N^1{}_i}{\partial \xi^1} \frac{\partial \xi^1}{\partial(\chi)} \underline{\boldsymbol{x}}_i \right|. \tag{3.48}$$

The factor $\partial\xi^1/\partial(\chi)$ reflects the fact that the integral can be evaluated only over a part of the mortar segment, the mortar domain. Note that the integrals (3.46), (3.47) and the Jacobian are evaluated in the reference or in the current configuration depending the problem being solved: tying or contact, respectively.

Linearization

Having derived the expressions for the mortar integrals D_{jk} and M_{jl}, the necessary linearization of the integrals for evaluating the tangent operator for the mortar element is presented here [Yang et al., 2005, Popp, 2012].

$$\begin{aligned}\Delta D_{lm} = & \, w_G \Delta\Phi_l(\xi_G^1)\, N_m^1(\xi_G^1)\, J_{\text{seg}}(\xi_G^1) + \\ & + w_G \Phi_l(\xi_G^1)\, \Delta N_m^1(\xi_G^1)\, J_{\text{seg}}(\xi_G^1) + \\ & + w_G \Phi_l(\xi_G^1)\, N_m^1(\xi_G^1)\, \Delta J_{\text{seg}}(\xi_G^1).\end{aligned} \tag{3.49}$$

$$\begin{aligned}\Delta M_{li} = & \, w_G \Delta\Phi_l(\xi_G^1)\, N_i^2(\xi_G^2)\, J_{\text{seg}}(\xi_G^1) + \\ & + w_G \Phi_l(\xi_G^1)\, \Delta N_i^2(\xi_G^2)\, J_{\text{seg}}(\xi_G^1) + \\ & + w_G \Phi_l(\xi_G^2)\, N_i^2(\xi_G^2)\, \Delta J_{\text{seg}}(\xi_G^1).\end{aligned} \tag{3.50}$$

Evidently for the mesh tying problems the mortar integrals remain unchanged and their variation is zero.

3.4.2 Evaluation of integrals in 3D

Conceptually the 3D implementation is similar to the 2D one, but involves implementation of additional algorithms like clipping and triangulation to determine the mortar domain. The steps involved in numerical mortar integral evaluation for the three-dimensional case are shown in Fig. 3.3 [Puso, 2004, Popp, 2012].

1. An auxiliary plane is constructed passing through the center of the mortar segment $\underline{\boldsymbol{x}}_c$ with mortar segment normal $\underline{\boldsymbol{n}}$. The auxiliary plane is used to avoid any possible warping of the interface facets. The elimination of mortar side warping is important since the mortar integrals are evaluated on this geometrically approximated area.

2. Projecting the mortar segment and the non-mortar segment nodes onto the auxiliary plane along the normal $\underline{\boldsymbol{n}}$.

3. To construct the mortar domain, we need to clip the overlapping 2D segments on the auxiliary plane.

4. The resulting intersection polygon is triangulated to obtain the integration cells for the overlap area of the two segments. Each integration cell consists of three vertices.

5. The Gauss integration points have to be projected from the auxiliary plane space along the normal $\underline{\boldsymbol{n}}$ onto the mortar and non-mortar segments to obtain the Gaussian integration point in the parametric spaces of mortar ($\underline{\boldsymbol{\xi}}^1$) and non-mortar ($\underline{\boldsymbol{\xi}}^2$) segments.

The numerical integration is performed over each cell (triangle). For a pair of overlapping 2D segments the entries of $\mathbf{D}$ and $\mathbf{M}$ matrices are

$$D_{lm} = \sum_{\text{cell}=1}^{N_{\text{cell}}} \left(w_G \Phi_l(\underline{\boldsymbol{\xi}}_G^1) N_m^1(\underline{\boldsymbol{\xi}}_G^1) J_{\text{cell}}(\underline{\boldsymbol{\xi}}_G^1) \right), \tag{3.51}$$

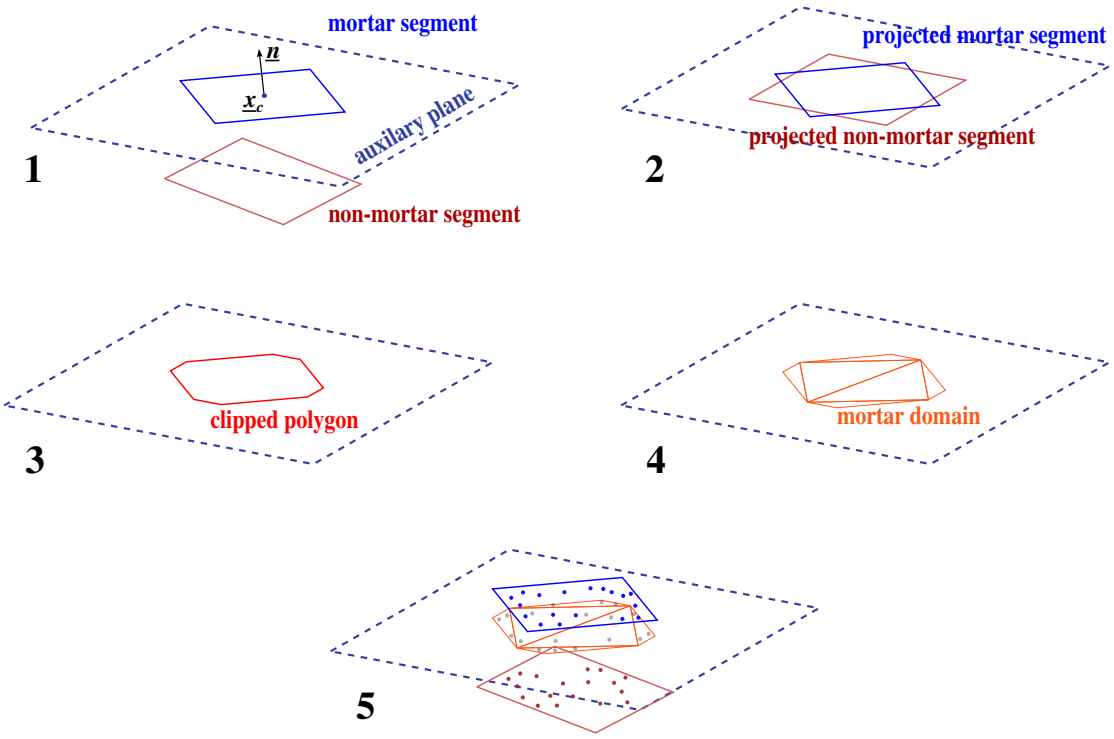


Figure 3.3: Algorithmic steps in numerical evaluation of mortar integrals for 3D.

$$M_{li} = \sum_{\text{cell}=1}^{N_{\text{cell}}} \left(w_G \Phi_l(\underline{\boldsymbol{\xi}}_G^1) N_i^2(\underline{\boldsymbol{\xi}}_G^2) J_{\text{cell}}(\underline{\boldsymbol{\xi}}_G^1) \right), \tag{3.52}$$

where J_{cell} is the Jacobian of the integration point.

Linearization

In line with the 2D linearizations, the 3D mortar integral evaluations are derived as follows:

$$\begin{aligned}
\Delta D_{lm} = & \sum_{\text{cell}=1}^{N_{\text{cell}}} w_G \Delta\Phi_l(\underline{\boldsymbol{\xi}}_G^1)\, N_m^1(\underline{\boldsymbol{\xi}}_G^1)\, J_{\text{cell}}(\underline{\boldsymbol{\xi}}_G^1) + \\
& + \sum_{\text{cell}=1}^{N_{\text{cell}}} w_G \Phi_l(\underline{\boldsymbol{\xi}}_G^1)\, \Delta N_m^1(\underline{\boldsymbol{\xi}}_G^1)\, J_{\text{cell}}(\underline{\boldsymbol{\xi}}_G^1) + \\
& + \sum_{\text{cell}=1}^{N_{\text{cell}}} w_G \Phi_l(\underline{\boldsymbol{\xi}}_G^1)\, N_m^1(\underline{\boldsymbol{\xi}}_G^1)\, \Delta J_{\text{cell}}(\underline{\boldsymbol{\xi}}_G^1).
\end{aligned} \tag{3.53}$$

$$\begin{aligned}
\Delta M_{li} = & \sum_{\text{cell}=1}^{N_{\text{cell}}} w_G \Delta\Phi_l(\underline{\boldsymbol{\xi}}_G^1)\, N_i^2(\underline{\boldsymbol{\xi}}_G^2)\, J_{\text{cell}}(\underline{\boldsymbol{\xi}}_G^1) + \\
& + \sum_{\text{cell}=1}^{N_{\text{cell}}} w_G \Phi_l(\underline{\boldsymbol{\xi}}_G^1)\, \Delta N_i^2(\underline{\boldsymbol{\xi}}_G^2)\, J_{\text{cell}}(\underline{\boldsymbol{\xi}}_G^1) + \\
& + \sum_{\text{cell}=1}^{N_{\text{cell}}} w_G \Phi_l(\underline{\boldsymbol{\xi}}_G^1)\, N_i^2(\underline{\boldsymbol{\xi}}_G^2)\, \Delta J_{\text{cell}}(\underline{\boldsymbol{\xi}}_G^1).
\end{aligned} \tag{3.54}$$

For the linearizations of the mortar integrals (3.49), (3.50), (3.53) and (3.54), the reader is referred to [Puso, 2004, Yang et al., 2005, Popp, 2012].

3.4.3 Clipping

The clipping of overlapping polygons is an important algorithmic aspect of the mortar based discretization scheme. It is essential to capture the true area of contact or tying between two overlapping discretized segments for accurate evaluation of mortar integrals. For the employed linear elements within the thesis, the two dimensional faces of the contact surfaces are always convex. For the case of quadratic elements, the faces are split into linear elements for the clipping procedures [Puso et al., 2008]. This simplifies the complexity of the polygon clipping algorithms required. Among many clipping algorithms, the ones proposed by Weiler and Atherton [Weiler and Atherton, 1977], Liang and Barsky [Liang and Barsky, 1983] and Sutherland and Hodgman are well known [Sutherland and Hodgman, 1974].

Here we employ the Sutherland-Hodgman algorithm as it is a simpler and straightforward algorithm to implement. The algorithm steps are given in Algorithm 1. Here two polygons are the input for the algorithm. One of the them is called the clipper which clips the other polygon called the subject. It works by infinitely extending each edge of the clipper in turn and selecting only vertices from the subject polygon that are on the visible side. A new intermediate polygon is created at the end of each iteration, with the vertices lying on the visible side along with the intersections between the extended edge and the subject polygon. After looping through all the edges of the clipper, the resulting polygon represents the overlap region of the two polygons [see Fig. 3.4].

Algorithm 1 Sutherland-Hodgman clipping algorithm

```
1: procedure CLIP_POLYGON(clipper,subject)          ▷ subject clipped with clipper.
2:     for each edge in clipper do
3:         clippedpolygon.clear()                   ▷ reset the clipped polygon for the next edge
4:         for (i=0 ; i<subject.length - 1 ; i++) do
5:             P_i = subject.vertex[i]
6:             P_{i+1} = subject.vertex[i+1]
7:             if P_i in clipper then               ▷ verify if a vertex is inside the clipper (polygon)
8:                 if P_{i+1} in clipper then
9:                     clippedpolygon.add(P_{i+1})
10:                else
11:                    clippedpolygon.add(intersectionpoint(P_i, P_{i+1}, edge)
12:                end if
13:            else
14:                if P_{i+1} is inside clipper then
15:                    clippedpolygon.add(intersectionpoint(P_i, P_{i+1}, edge)
16:                    clippedpolygon.add(P_{i+1})
17:                end if
18:            end if
19:        end for
20:        subject = clippedpolygon                 ▷ The clipped polygon becomes the subject for the
    subsequent iterations.
21:    end for
22: end procedure
```

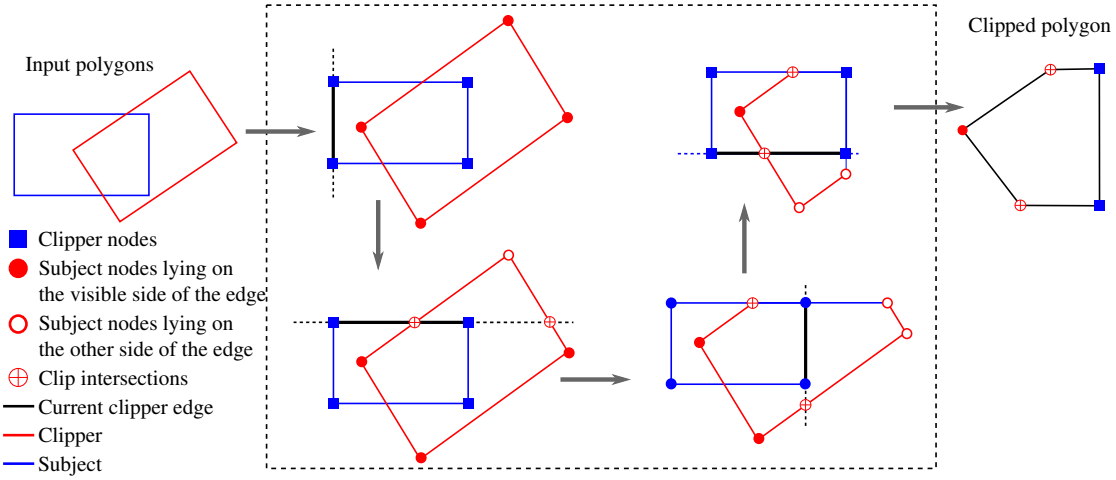


Figure 3.4: Illustration of Sutherland-Hodgman algorithm for clipping of two overlapping convex polygons.

3.4.4 Triangulation

The intersection of convex polygons always results in convex polygon. The resulting polygon has arbitrary number of edges. In order to be able to use the standard numerical quadrature rules of integration for the evaluation of the mortar integrals, the clipped polygon is triangulated. For the clipped convex polygon, the triangulation is straight forward. It involves selection of a vertex and creating a triangle fan outward [see Fig. 3.5]. The steps are listed in Algorithm 2. Different triangulation techniques can be used [Bykat, 1976, Lee and Schachter, 1980]. Alternatively, the polygons can be split into quadrilateral elements.

Algorithm 2 Convex polygon triangulation algorithm

```
1: procedure TRIANGULATE POLYGON(vertex_list) ▷ vertices stacked in clockwise order
2:     base_vertex = vertex_list[0]              ▷ the vertex is deleted from vertex_list
3:     mid_vertex = vertex_list[0]               ▷ the vertex is deleted from vertex_list
4:     while vertex_list not empty do
5:         temp_vertex = vertex_list[0]          ▷ the vertex is deleted from vertex_list
6:         CREATE_TRIANGLE(base_vertex, mid_vertex, temp_vertex)
7:         mid_vertex = temp_vertex
8:     end while
9: end procedure
```

Each triangle acts as an integration cell, for the mortar integral evaluation. Each vertex of the cells is either a projection of the mortar and non-mortar segments onto the auxiliary plane or an intersection of the polygon edges. This classification of the integration cell vertices is necessary as the underlying elementary linearizations in 3D case, requires linearization of the positions of these vertices [Puso and Laursen, 2004, Popp, 2012].

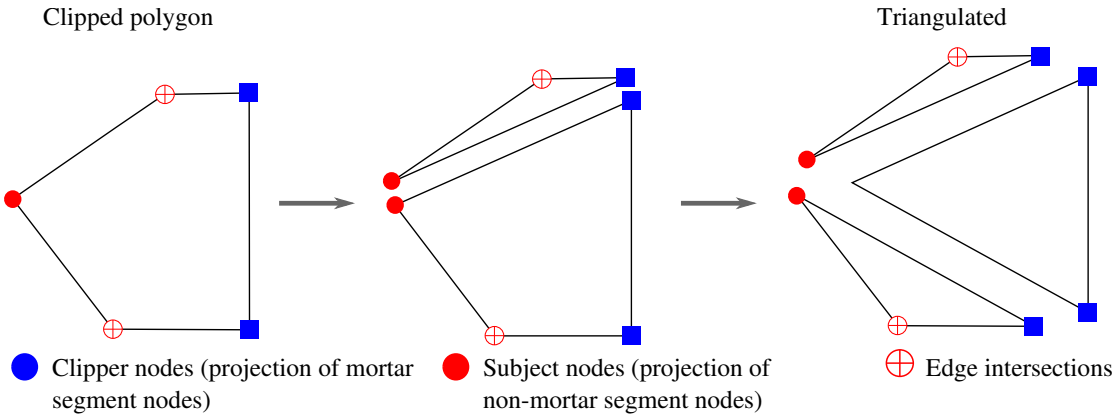


Figure 3.5: Illustration of convex polygon triangulation.

3.5 Aspects of detection and parallel resolution

3.5.1 Efficient contact detection

The contact detection is a crucial ingredient of the numerical framework for treating contact problems. It is a starting point that is followed by the discretization and resolution steps. The lack of adequate detection schemes could potentially have a cascading effect on the subsequent steps leading to e.g. unrealistic interpenetrations between contacting bodies when a potential contact is not detected. Numerical algorithms for detection are quite complex in nature as the contact detection has to be done on flight at every iteration or load step for elements which are already in contact and those which can potentially come in contact [Plimpton et al., 1998, Williams and O'Connor, 1999, Fujun et al., 2000, Yastrebov et al., 2011a, Kopačka et al., 2016]. Here we briefly outline our approach based on a partition of space using a binary search tree (type "KD-Tree") and the definition of a "Bounding box".

Bounding box

Here we define a simple offset box enveloping each non-mortar surface. For this a "warning distance" is defined. The algorithm treats a pair of contacting segments as potential contact pair, when the distance between them is below the warning distance. An offset distance is defined using the warning distance. The key here is to define an optimum warning distance which determines the offset surface. Other complex and optimized variants of bounding box like the axis-aligned bounding box, oriented bounding box and k faced discrete orientation polytopes can be found in the works of Yang and Laursen [Yang and Laursen, 2008]. In the case of a bounding box containing a mortar side node, the corresponding non-mortar and mortar segments are considered as potential contact pairs and processed further (clipping and triangulation).

KD-tree

In order to accelerate contact recognition, we use a variant of the data structure "KD-tree" (tree has K dimensions). This structure is constructed by looking for the median of the centers of segments along each axes (x, y, z) to create a partition

of the space. Each partition is in turn under-cut along the median of the following axis until a single segment is obtained per partition. Special attention is given to the search algorithm of the median of a set of nodes. A simplistic approach would consist in sorting out segments along one of the axes and then divide the whole into two parts of equal number of constituents. We optimized this approach by deriving a "quick select" algorithm from the original "quick search" algorithm [Knuth, 1998], this algorithm focuses on the objective of finding the median element along an axis without sorting all the constituents of a subspace. This approach allows to reach a complexity of $\mathcal{O}(n \log(n))$ for the construction of the binary tree. Once the individual segment bounding boxes are created, the tree is crossed from the bottom up by merging the bounding boxes of individual segments then those of the following levels. This allows to create enclosing boxes for each partition of the space up to that encompassing the entire non-mortar surface. The search procedure is simple: if a mortar node is inside the bounding box of a level of the tree, we test the lower levels attached to it until the segments potentially in contact with the node are determined. If the node is outside of any segment bounding box, the search is stopped. The search algorithm is also very efficient with a complexity of $\mathcal{O}(n)$ in the worst case.

Contact patch test

The primary issue addressed here is to confine the number of potential non-mortar segments for each mortar segment to a minimum. The confined set of non-mortar segments are considered for further numerical procedures like the clipping and triangulation to precisely evaluate the contact kinematic quantities like gap and formulate corresponding contact elements, residuals and tangent operators. To illustrate the performance of the implemented KD-tree algorithm, we consider a contact patch test setup [Fig. 3.6(a)]. This consists of two square plates of side $L = 1$ mm and thickness $h = 0.01$ mm, with flat contacting surfaces. The flat surfaces imply maximum contact area. A vertical displacement component u_y is applied on the top surface of the upper plate. This results in a uniform stress p MPa in the bodies [Fig. 3.6(b)]. In case of brute detection scheme, every mortar segment is checked against every non-mortar segment for projections. On the other hand, with a preliminary segregation of non-mortar segments based on bounding boxes and KD-tree formation confines the potential non-mortar segments for projections. This fact is reflected in gains achieved in the detection time shown in Fig. 3.7.

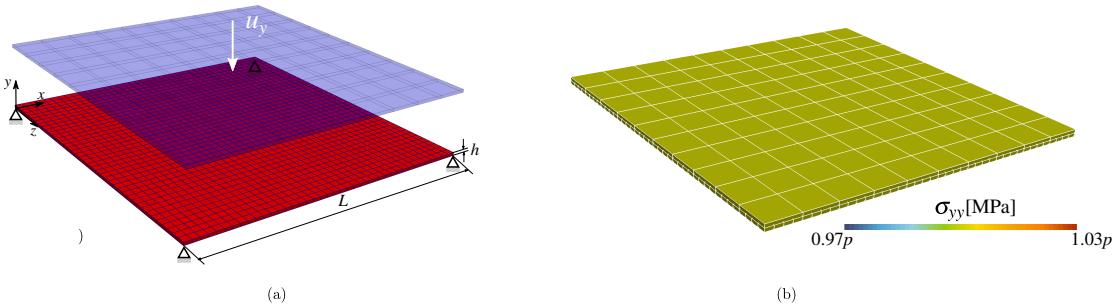


Figure 3.6: (a) Contact patch setup; (b) Uniform stress distribution for the contact patch test.

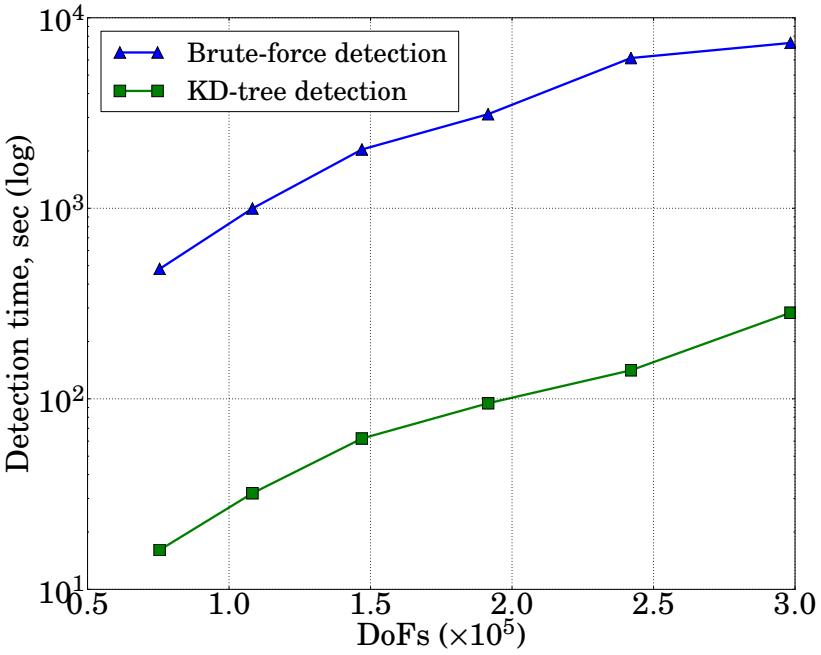


Figure 3.7: Comparison of detection time for the brute force and KD-tree scheme for increasing number of DoFs.

3.5.2 Parallel resolution

The mortar algorithms presented in the earlier sections are computationally intensive, as they involve operations like projections, clipping and triangulation, which need performed at every iteration of each increment. Within the generalized Newton-Raphson scheme employed, these operations could be invoked several times resulting in too long computation time on a single-processor based architecture. Within the FE formulations different parallel computing paradigms exist. These are classified based on the type of concurrency imposed: node/element-wise [Ortiz and Nour-Omid, 1986, Abel et al., 1991], column-wise [Farhat and Wilson, 1988, Farhat and Crivelli, 1989] and domain-wise [Farhat and Wilson, 1987, Sun and Mao, 1988, Yagawa et al., 1991]. With the domain decomposition being the underlying philosophy of the mortar methods, it becomes the natural choice of parallel implementation of the mortar methods for both mesh tying and contact problems.

The domain decomposition method (DDM) paradigm can be applied as a "continuous" version (at the level of PDE problem) or "discrete" version (at the level of underlying discretization). In the continuous approach the PDE is subdivided into a number of smaller coupled PDE problems. Within the discrete setting the mesh is subdivided and the corresponding computations are assigned to different nodes/cores/processors on distributed or shared memory architectures.

In the geometry partitioning strategy, the discretized FEM domain Ω_0 is subdivided into P overlapping or non-overlapping subdomains sharing interface nodes [Farhat, 1988, Simon, 1991, Gupta and Ramirez, 1995]. Here, we exclusively focus on non-overlapping domain decomposition. Few important criteria should be respected in this subdivision to make the parallel architectures efficient (balanced work between nodes, minimal communication):

- the subdomains shall have approximately the same number of elements,
- the number of nodes in the interface between subdomains should be as small as possible,
- the number of adjacent subdomain to each subdomain should be kept minimal,
- the communication requirements of the underlying computation on a given architecture (processor interconnection graph) should be minimized.

Typically, the DDM is combined with iterative solvers. Within such a framework the finite element analysis is performed separately on each processor, while ensuring displacement continuity and thus force balance between subdomains is achieved in an iterative manner. The monolithic augmented Lagrangian scheme employed here results in a typical saddle point operator structure that is not positive definite. This makes it a poor candidate for iterative solvers. This leaves us with the option of distributed sparse direct solver. The distributed sparse direct solver decomposes a large sparse matrix into smaller sub-matrices, and then sends these sub-matrices to multiple nodes on either shared-memory (e.g., server) or distributed-memory (e.g., cluster) hardware [Amestoy et al., 2000]. Here, in the considered DDM paradigm each sub matrix represents the finite element sub domain. During the matrix factorization phase, each process factorizes its sub-matrices simultaneously and communicates the information as necessary [see Fig. 3.9(a)]. This communication is ensured by message passing, normally using MPI. Avoiding the use of iterative based DDM, circumvents the need to ensure minimum number of iterations required to enforce continuity across subdomains and related convergence issues. However, it comes at a price of additional MPI calls initiated by the distributed solver to assemble the global matrix on a single processor.

Set-up

In order to parallelize the computationally intensive mortar algorithms, the mortar data structures have to be adapted to accommodate the Message Passing Interface (MPI) communication. The parallelization of the contact resolution enables the following algorithm executions in parallel:

- the geometrical mortar procedures of projection, clipping and triangulation along interface are performed simultaneously on each sub-domain,
- linearization of contact residuals for each sub-domain,
- factorization of the local sub-domain operators (internal MPI calls are initiated by the distributed solver MUMPS [Amestoy et al., 2000] for the interface factorizations).

The first step towards achieving a parallelization is to partition the domain into sub-domains. Graph methods are widely used for domain partitioning. The RGB (recursive graph bisection) and RGL (recursive graph labeling) are the two commonly used methods [Simon, 1991, Nikishkov et al., 1999]. The primary requirement of a DDM procedure is to ensure optimal balancing of computations among

the processors and minimal communication. The computational time is a direct reflection of the quality of the partitioning algorithm [Schloegel et al., 2000]. Here we use a regular user defined split domain partitioning technique. For the examples considered, this gives a better control on ensuring an optimal load balance among the various nodes.

Data exchange

The mortar MPI communication involves packing of non-mortar interface data for sharing among processes [see Fig. 3.9(b)]. The data includes the non-mortar surface nodes, their ranks and DoFs and nodal connectivity information from each non-mortar sub-domain. The data exchange is categorized into pre and post MPI exchanges. Each sub-domain has a set of preliminary data set-up steps, to enable it to share its data with other sub-domains. This includes:

- creating a data structure of vectors, that define uniquely each segment of the discretized surface.
- creating a list of integers (IDs, global ranks) and vectors (DoFs).

The post MPI data exchange involves:

- assembly of the received data into separate global containers, to be used for the mortar algorithms.

For each sub-domain containing the non-mortar segments, the following data is packed:

- A unique nodal ID that recognizes the node in the global unpartitioned mesh. This is essential to track the node being shared across different sub-domains.
- The initial position of a node is also shared, as this is essential for obtaining the position of node at any given instance during the simulation (BEGINNING-OF-INCREMENT, END-OF-INCREMENT, BEGINNING-OF-PROBLEM). The position at any of the these instances is obtained by appending the initial position with the displacement DoF value.
- The segment definitions are required by the receiving sub-domain in order to construct an internal geometric surface on which the mortar algorithms operate. Unlike the node-to-segment scheme where only the node data is shared, the mortar method requires complete geometric surface information. This requirement increases the volume of data exchanges done over MPI.

Contact patch test

The same contact patch set-up considered for detection schemes [see Fig. 3.6(a)] is used for the illustrating the parallel performance. For this particular problem, the non-mortar side Ω^1 of the domain is not partitioned, where it is treated as a single sub-domain [see Fig. 3.8]. This choice is is partially justified by the fact that the non-mortar side is coarser compared to the mortar side. Also this particular configuration, could result in a minimum exchange of MPI data of the non-mortar

side quantities and maximum parallelization of mortar algorithms (projections, clipping, triangulations, linearization etc). However, the implementation can handle arbitrary partitioning in which both mortar and non-mortar are partitioned into multiple subdomains.

The bottom plate is considered as the mortar side of the interface and is divided into 4 sub-domains. The mortar side is approximately 4 times finer than the non-mortar side. Also there is only one element in the thickness direction. Fig. 3.10(b), shows the CPU time comparison for the cases of sequential and parallel contact resolution for increasing number of DoFs (increased proportionally for the mortar and non-mortar sides). On the other hand, Fig. 3.11 shows the effect of domain partitioning on the CPU time. Interestingly, the gain in CPU time stabilizes beyond a certain number of sub-domains into which the domain is partitioned. Note that in the figures, the number of mortar side sub-domains are referred to as "cores".

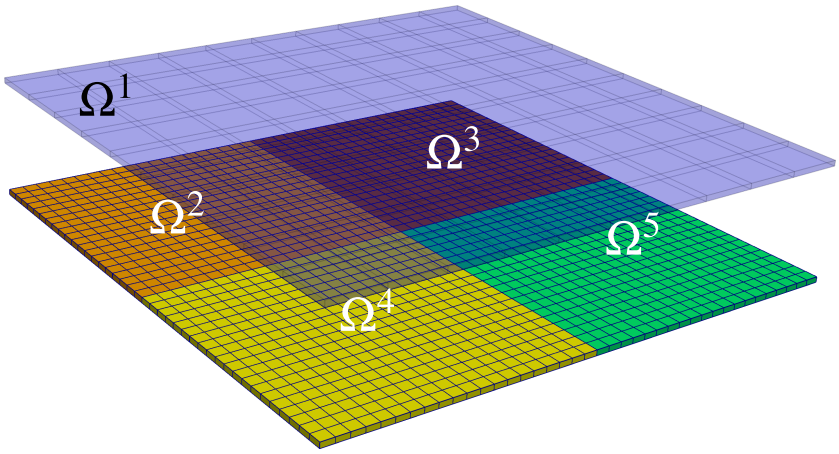


Figure 3.8: Ω partitioned into Ω^i $(i = 1, 2, 3, 4, 5)$.

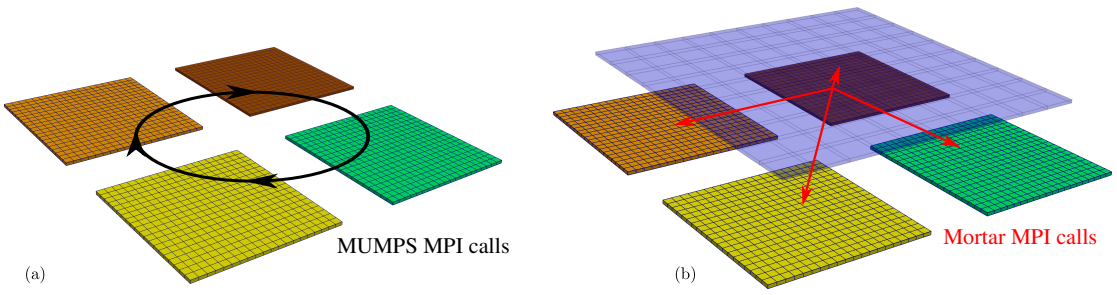


Figure 3.9: (a) MPI calls initiated between sub-domains by MUMPS, for matrix factorization; (b) MPI calls to transfer non-mortar interface definitions between all the sub-domains.

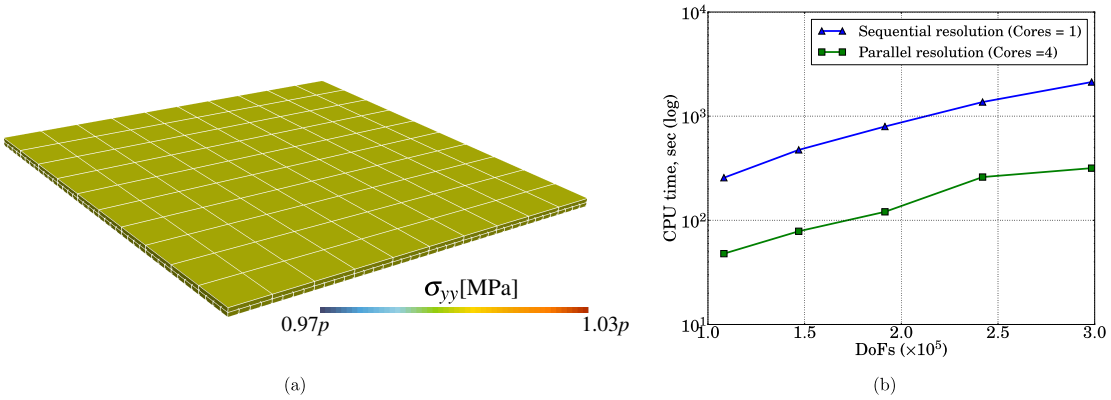


Figure 3.10: (a) Uniform stress distribution of the contact patch test; (b) comparison between sequential and parallel resolution of the contact problem.

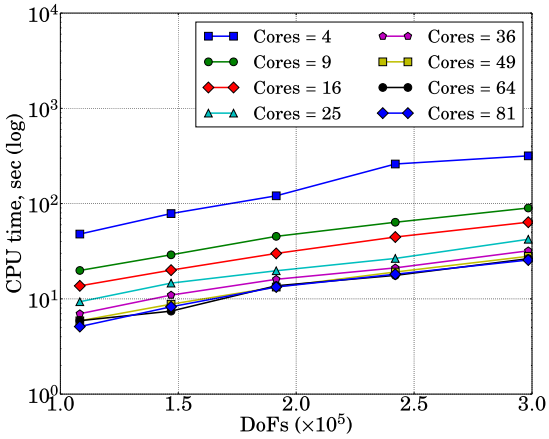


Figure 3.11: Effect of the number sub-domains into which Ω is partitioned on: (a) CPU time; (b) MPI exchange time.

Rough surface in contact with rigid plane

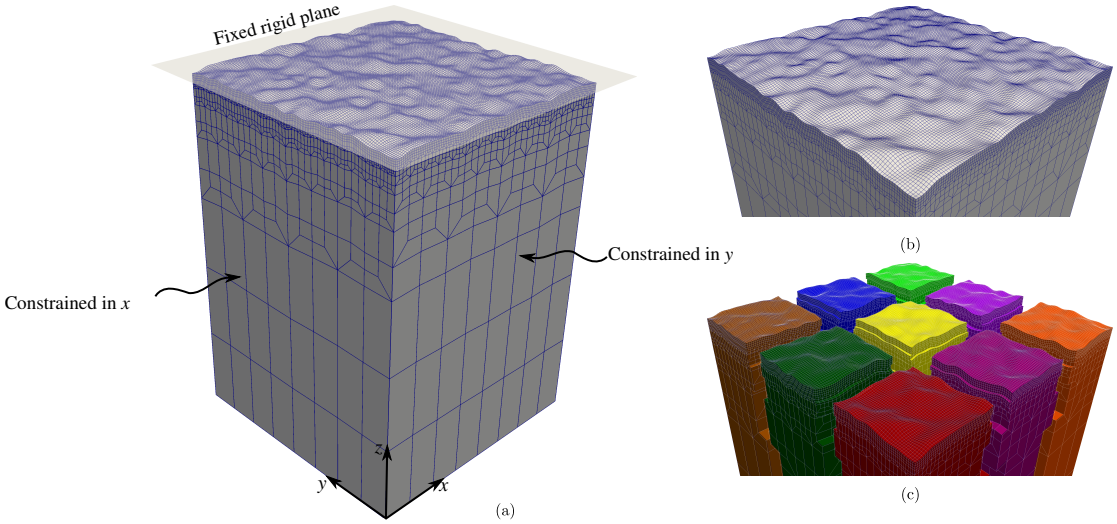


Figure 3.12: (a) the FE mesh of a block with rough contact surface (roughness scale factor = 2) along with the boundary conditions; (b) closer view of the discretized rough contact surface; (c) mortar side partitioned into 9 sub-domains.

Here, we consider a frictionless normal contact problem setup between a rough surface and a rigid plane. The block has a square cross section of side 1 mm and height 1.478 mm. As shown in Fig. 3.12, a fine mesh discretization with 256×256 elements is used for the contacting surface. The bottom face is fixed in the x and y directions and vertical displacement of $u_z = 0.02$ mm is applied on the bottom surface, in order to make contact with the rigid surface. The rough surface is considered as the mortar side of the interface. A linear elastic material with $E = 1$ MPa and $\nu = 0.3$ is used for the rough surface block. This set-up is used to illustrate the effect of number of partitions on the CPU time. The mortar side of the problem is divided into $[4, 9, 16, 25, 36]$ domains. Exemplarily, the domain partitioned into 9 sub domains is shown in Fig. 3.12(c). The contour stress plots of the stress component σ_{zz} Fig. 3.13(a) , demonstrates both the peaks of the rough surface getting into contact with the rigid plane resulting in a negative σ_{zz} stress, and the no-contact regions with zero stress. The reduction in CPU time as a function of the number of cores is shown on Fig. 3.13(b).

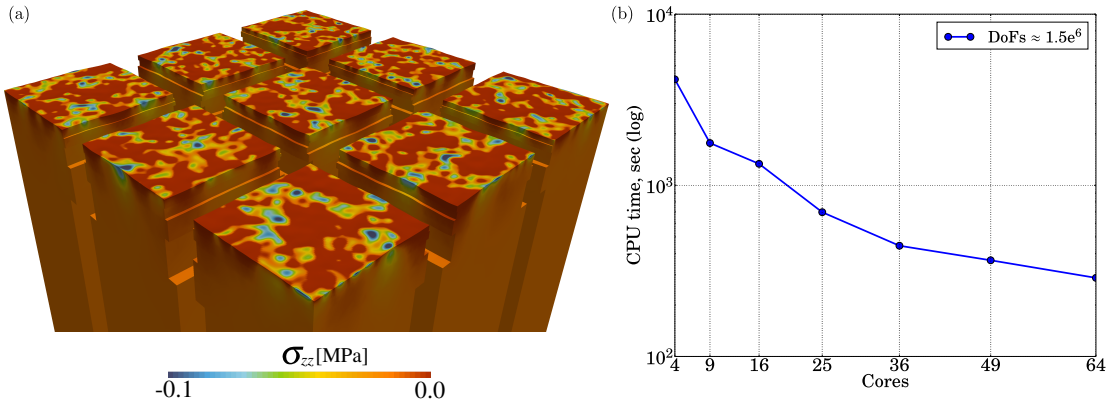


Figure 3.13: (a) Contour stress plot of contact stress σ_{zz}; (b) the effect of partitioning on the CPU time.

3.6 Case studies

3.6.1 Incremental/Iterative update of mortar elements

Within the generalized Newton-Raphson scheme employed for solving the contact non-linear problem, the mortar contact interface elements are reinitialized at the beginning of each load step. The initialization is based on the stable equilibrium configuration obtained from the previous converged load increment. However, this kind of incremental update of the mortar interface contact elements could compromise the accuracy of the solution in certain cases and might not converge in others. Here we illustrate both the cases of loss of accuracy and lack of convergence (for frictionless contact problem). The main steps involved in resolution of frictionless contact problem are listed in Algorithm 3. Note that separate residual norms for the primal $||\underline{\boldsymbol{r}}||_{\text{primal}}||$ and duals $||\underline{\boldsymbol{r}}||_{\text{dual}}$ DoFs are checked against tolerances within the Newton-Raphson iteration scheme.

Algorithm 3 Frictionless contact problem resolution of an increment

Update boundary conditions
Update λ_n from previous converged increment
CREATE_MORTAR_ELEMENTS( ) ▷ involves projections, clipping, triangulation.

procedure NEWTON_RAPHSON_LOOP( )
  while ($||\underline{\boldsymbol{r}}||_{\text{primal}} > \text{tol1}$ and $||\underline{\boldsymbol{r}}||_{\text{dual}} > \text{tol2}$) **do**
    if (iterative-update) **then**
      CREATE_MORTAR_ELEMENTS( ) ▷ For iterative update, the mortar interface elements are created at the beginning of each Newton iteration.
    end if
    for `el in mortar_elements` **do** ▷ loop over each mortar interface element.
      $r^{\text{el}} = 0$
      D_{lm}, M_{li} ←EVALUATE_INTEGRALS( ) (3.14), (3.15) ▷ involves projections, clipping, triangulation
      for (l=0 ; l<L ; l++) **do** ▷ Loop over each mortar side node
        $\bar{g}_n^l$ ←COMPUTE_GAP( )(3.17)
        $\hat{\lambda}_n^l$ ←COMPUTE_AUG_LM( )(3.35)
        if ($\hat{\lambda}_n^l \leq 0$) **then**
          r^{el}+ =EVALUATE_RESIDUAL( ) (3.30)
        else
          r^{el}+ =EVALUATE_RESIDUAL( ) (3.31)
        end if
      end for
      if ($\hat{\lambda}_n^l \leq 0$) **then**
        EVALUATE_K^{el}( )(3.38)
      else
        EVALUATE_K^{el}( )(3.39)
      end if
    end for
    if (iterative-update) **then**
      REINTIALIZE_INTERFACE( ) ▷ For iterative update, the mortar interface elements are deleted at end of each Newton iteration.
    end if
  end while
end procedure

Case 1: Accuracy

The problem under consideration is the frictionless Hertzian contact between two cylinders [see Fig. 3.14]. The cylinders are of equal radius $R = R_1 = R_2 = 8$ mm. Linear elastic materials are used for both the domains ω^1 (E_1, ν_1) and ω^2 (E_2, ν_2). A material contrast is introduced by choosing $E_2/E_1 = 100$ ($E_1 = 10$ MPa). The problem is considered under plane strain assumption. The same Poisson's ratio of $\nu_1 = \nu_2 = 0.3$ is used for both the domains. The volumes in the vicinity of the contact zone are meshed with bilinear quadrilateral elements, with comparable mesh densities for both the mortar (ω^1) and non-mortar (ω^2) domains. The bottom surface of the lower half cylinder is fully fixed. A downward vertical displacement $u_y = 0.01$ mm is applied on the surface of the top cylinder, while

fixing the center of the top surface in x direction. This results in a total reaction force of $P = 0.00341678$ N, in the y direction. The analytical solution for this

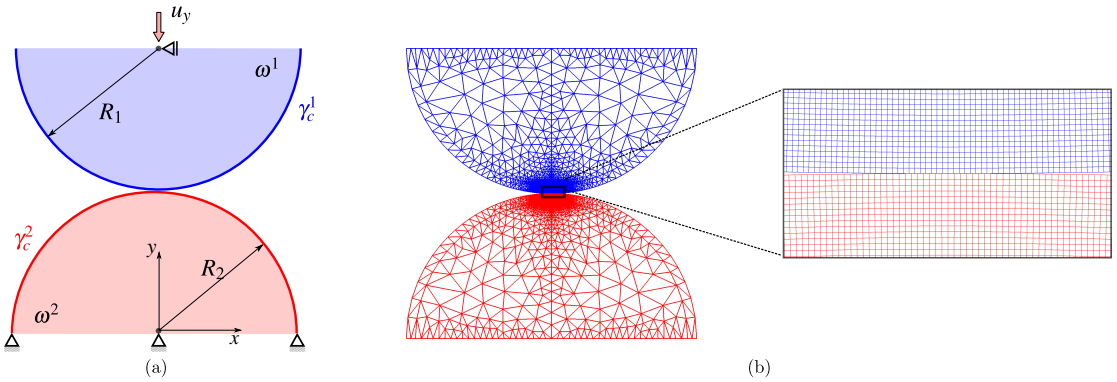


Figure 3.14: Hertzian contact: (a) cylinder on cylinder contact setting; (b) Discretization of the cylinders with comparable mesh densities (zoom at the interface mesh).

problem is derived from the Hertzian contact formulae for two cylinders, which defines the maximum contact pressure (p_0), the semi contact width a and the contact pressure distribution p along the x coordinate [Johnson, 1985].

$$p_0 = \sqrt{\frac{PE^*}{\pi R^*}}, \quad (3.55) \qquad a = \sqrt{\frac{4PR^*}{\pi E^*}}, \quad (3.56) \qquad p = p_0\sqrt{1 - \left(\frac{x}{a}\right)^2}. \quad (3.57)$$

The effective elastic modulus E^* is introduced as below:

$$E^* = \frac{E_1 E_2}{E_1(1-\nu_2^2) + E_2(1-\nu_1^2)}, \quad (3.58)$$

and the effective radius R^*, is evaluated as:

$$R^* = \frac{R_1 R_2}{R_1 + R_2}. \quad (3.59)$$

Here we consider the two sub cases which differ in the number of load steps, within which the total displacement u_y is applied. The Lagrange multipliers (λ_n) are plotted for both the sub cases at the end of load step, for the incremental and iterative update of contact elements. For the case of one single load increment, the incremental update of contact elements results in oscillatory λ_n profile, with a poor approximation of the analytical Hertz solution. In contrast the iterative update of contact elements, has a smoother λ_n profile and approximates better the analytical Hertz solution [Fig. 3.15(a)].

This behaviour results from the fact that the incremental update, uses only the initial set of contact elements created at the beginning of increment. This has two consequences:

- during the iterative process a possible loss of contact between existing contact pairs is not taken into account.

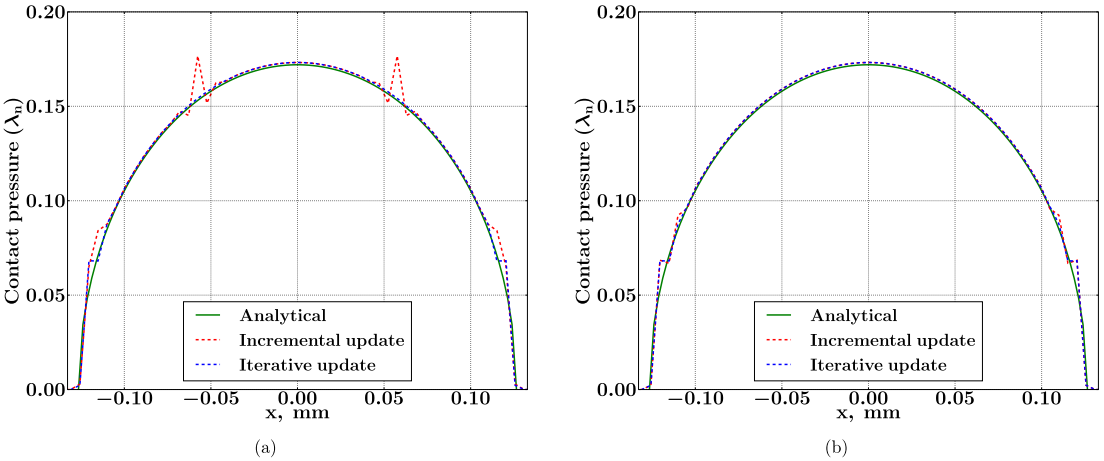


Figure 3.15: Contact tractions comparison at $t = 1$ seconds: (a) one load increment; (b) two load increments.

- the parts newly entering in contact during the convergence are not taken into account.

Both of the above consequences lead to a faulty active set of interface segments at the current iteration. The iterative update is more accurate as the contact elements are recreated based on the current configuration, and the above two issues are addressed. Displacements of the order of the interface element sizes can experience the above issues within the incremental update. To ensure accuracy in case of incremental update, the load step size should be small. This is illustrated in second sub case, where the displacement is applied in two load increments [Fig. 3.15(b)]. This results in a smoother profile for the incremental update as well, which is however, still outperformed by the iterative update.

Case 2: Convergence

Here, we consider a simple setting involving two bodies discretized by one element each. The problem setting and the boundary conditions are shown in Figure 3.16(a). The dimensions of the bodies are: $L_1 = 2$, $L_2 = 1$, $L_3 = 0.7$, $L_4 = 0.5$ mm respectively. The boundary conditions are intended to bring them in normal contact during the first sequence and to loose the contact as a result of sliding at the end of second sequence. Each load sequence is applied in 10 load steps. The first load sequence applies a vertical displacement u_y ($t = [0, 1.0\,\mathrm{s}]$) followed by the second sequence of horizontal displacements u_x ($t = [1.0, 2.0\,\mathrm{s}]$), maintaining the vertical displacements. .

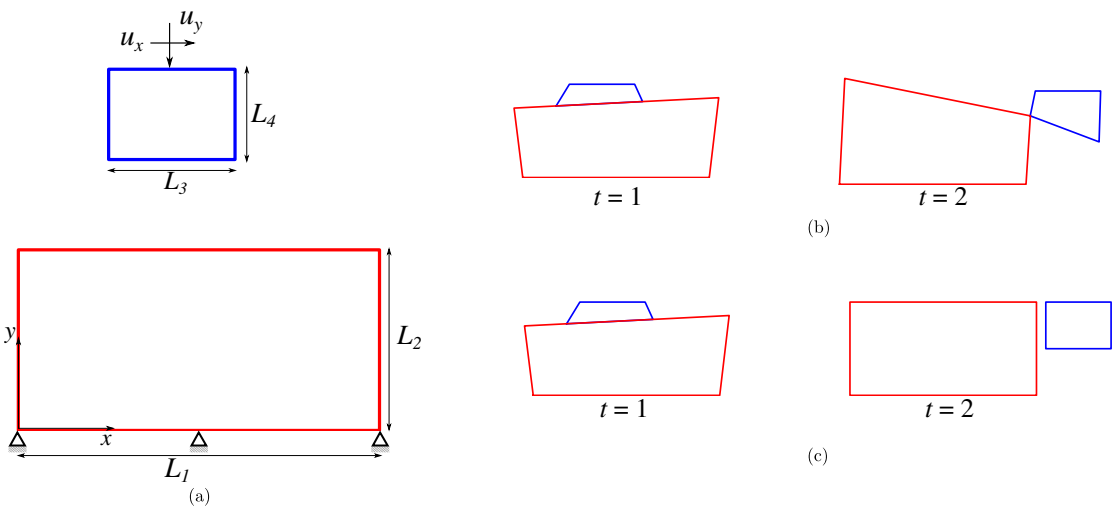


Figure 3.16: (a) Problem setting of two bodies coming into contact and loosing contact; (b) configurations at $t = 1, 2$ sec for the incremental update; (c) configurations at $t = 1, 2$ sec for the iterative update.

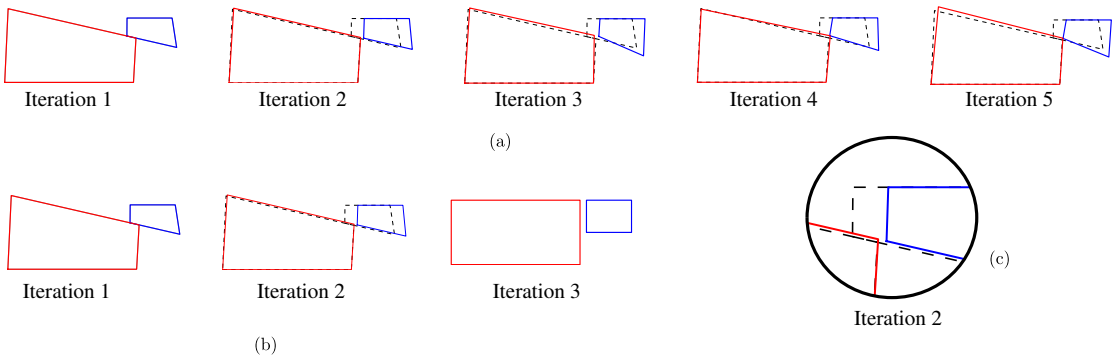


Figure 3.17: Configurations for load increment 10 at the end of each Newton-Raphson iteration (the dotted lines indicate the equilibrium configuration of the previous load increment): (a) incremental update; (b) iterative update; (c) closeup look at the interface for Iteration 2.

Fig. 3.16(b), indicates that at the end of the load sequence 2, the bodies are still in contact for the incremental update contrary to the iterative update in Fig. 3.16(c) where the bodies lose contact. Having a closer look at the intermediate configurations of the Newton-Raphson scheme provides us with plausible explanations. Fig. 3.17 shows these configurations during the Newton-Raphson procedure for the incremental and iterative updates of the contact interface elements. The incremental update fails to converge within the limit of 5 iterations. The dotted lines in these figures indicate the equilibrium configuration of the previous converged increment. This is the configuration on which the mortar algorithms of projections are performed. It is evident from Fig. 3.17(c), that the application of boundary condition leads to loss of contact. However, in case of incremental update the mortar element formed in the first iteration cannot be “broken”, while the iterative update allows to reinitialise the contacting pairs, rendering the loss of contact possible. The iterative update on the other hand, reconstructs the mortar element on the actual configuration.

The price paid for the accuracy and convergence is in terms of the computational time of the iterative update. The reconstruction of mortar interface contact elements invokes the underlying operations of projections, clipping and triangulations during every Newton-Raphson iteration.

3.6.2 Tangential slip history

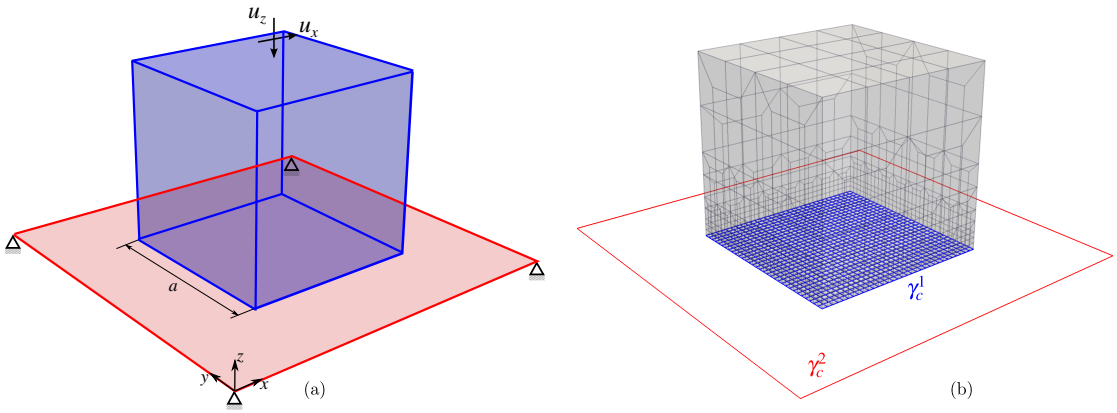


Figure 3.18: (a) Problem setting of the sliding cube problem; (b) FE discretization of the mortar and non-mortar sides.

In case of frictional problems, the incremental slip value $\underline{\mathring{\boldsymbol{g}}}_{\tau}^{l}$ of a mortar node l is stored, between each load step, i.e. at the end of each converged increment $\underline{\mathring{\boldsymbol{g}}}_{\tau}^{l}$ are stored. The evaluation of tangential augmented Lagrange multiplier $\underline{\hat{\boldsymbol{\lambda}}}_{\tau}^{l}$ (3.36) requires the incremental slip value. This evaluation of $\underline{\hat{\boldsymbol{\lambda}}}_{\tau}^{l}$ is very crucial, as the tangential contact status of slip and stick within the ALM method is decided by it. During the first load step increment, all the mortar nodes that are in contact have a default stick status. In the subsequent load steps, the history of slip for a mortar element from the previous converged increment has to be stored and used for initializing the slip. The following problem setting is used to emphasize on the necessity to store the tangential slip history of an interface element to accelerate the simulation. A deformable cube of side $a = 1$ mm is pressed vertically and slided horizontally on a rigid plane surface. A linear elastic material with $E = 1$ MPa and $\nu = 0.3$ is used for the cube. In a first load sequence ($t = [0, 1]$ s) a vertical displacement of $u_z = 0.05$ mm is applied on the top surface of the cube in 10 load steps, while preserving the vertical displacement. During the second load sequence ($t = [1, 2]$ s) a horizontal displacement of $u_x = 0.2$ mm is applied on the top surface in 100 load steps [see Fig. 3.18(a)]. The rigid plane surface γ_c^2 is chosen as non-mortar side and is discretized with a single element. The surface of the cube γ_c^1 is chosen as the mortar side and has a finer discretization [see Fig. 3.18(b)]. Here we consider two cases:

- Case 1: the nodal incremental slip values $\underline{\mathring{\boldsymbol{g}}}_{\tau}^{l}$ are not stored.
- Case 2: the nodal incremental slip values $\underline{\mathring{\boldsymbol{g}}}_{\tau}^{l}$ are stored.

The Newton convergence summary is presented in Table (3.1) for selected load increment steps. The residual norm considered here is given as below:

$$||\underline{\boldsymbol{r}}|| = \begin{cases} \frac{||\underline{\boldsymbol{r}}^{\text{int}} - \underline{\boldsymbol{r}}^{\text{ext}}||_2}{||\underline{\boldsymbol{r}}^{\text{ext}}||_2}, & ||\underline{\boldsymbol{r}}^{\text{ext}}|| > 0 \\ ||\underline{\boldsymbol{r}}^{\text{int}} - \underline{\boldsymbol{r}}^{\text{ext}}||_\infty, & ||\underline{\boldsymbol{r}}^{\text{ext}}|| = 0. \end{cases} \tag{3.60}$$

	increment 1		increment 2		increment 50	
Iteration	Case 1	Case 2	Case 1	Case 2	Case 1	Case 2
1	2.444e-02	2.444e-02	1.249e-02	2.476e-04	8.068e-03	1.398e-08
2	1.846e-02	1.846e-02	9.391e-03	8.672e-08	3.169e-03	-
3	8.751e-03	8.751e-03	4.573e-03	-	7.005e-04	-
4	3.421e-03	3.421e-03	1.693e-03	-	9.152e-08	-
5	1.833e-03	1.833e-03	8.942e-04	-	-	-
6	8.791e-04	8.791e-04	3.678e-04	-	-	-
7	1.698e-04	1.698e-04	1.623e-04	-	-	-
8	4.890e-05	4.890e-05	2.004e-05	-	-	-
9	9.223e-08	9.223e-08	6.365e-08	-	-	-

Table 3.1: The Newton-Raphson convergence loop for various load steps: 1, 2 and 50 (of load sequence two i.e. at $t = 1.5$ seconds).

During the first increment of vertical displacement sequence, all the mortar side nodes are in the stick status. As seen in Fig. 3.19, for both the cases the evolution of the slip and stick status of the frictional problem are the same. However, in the following load step, for the Case 1 where the nodal slip incremental values are not stored, it takes again nine Newton-Raphson iterations to converge, compared to only two iterations for Case 2 (see Fig. 3.20, Table 3.1). The greater number of iterations in Case 1 can be attributed to the fact that in the absence of previously converged values of nodal incremental slips, the current load step is initialized with faulty contact statuses. However, both of the simulations converge to the same solution.

Also, we consider the second load sequence. The problem setting involving tangential displacement boundary conditions results in eventual full sliding mode of the cube when ($\mu|P| > |T_x|$). Here P is the total vertical reaction scaled by the friction coefficient $\mu|P|$ and the $|T_x|$ is the tangential reaction force [see Fig. 3.22]. Once the Coulomb threshold is reached, the subsequent load steps would need minimum iterations to converge as the contact status change between stick and slip is no longer present here. This can be seen for the Case 2, as seen in Fig. 3.21(b). However, for Case 1 the convergence is longer as previously observed, the lack of initialization of nodal incremental slip values from the previous converged load step, results in faulty contact status that manifests itself in additional Newton iterations for convergence.

3.6.3 Friction vs Frictionless: incremental and iterative updates

It was illustrated in Section 3.6.1, that the iterative update of the interface elements has some advantages compared to the incremental update. The purpose of

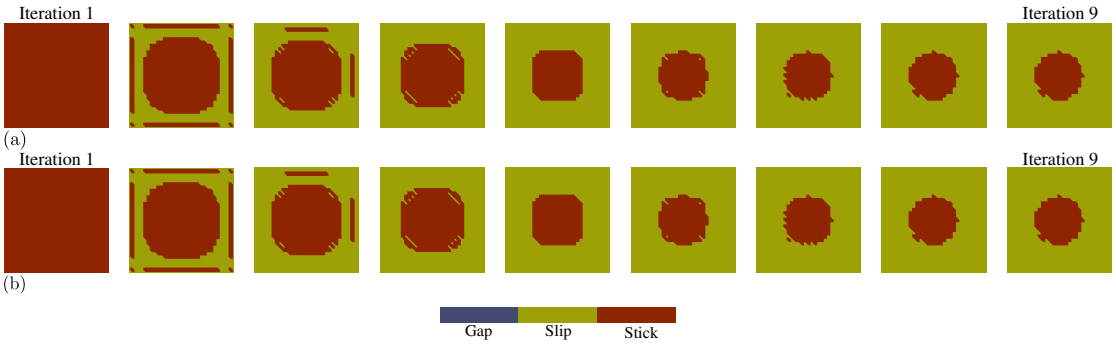


Figure 3.19: The evolution of stick and slip status during the Newton-Raphson iterations of a load increment 1: (a) Case 1; (b) Case 2.

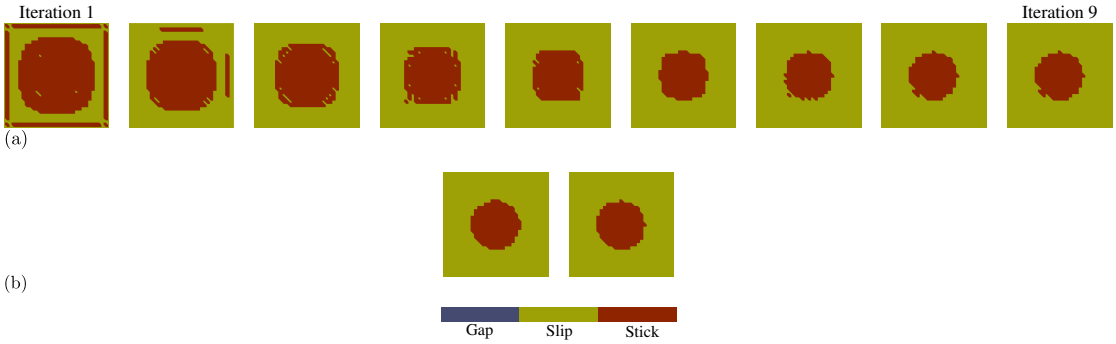


Figure 3.20: The evolution of stick and slip status during the Newton-Raphson iterations of a load increment 2: (a) Case 1; (b) Case 2.

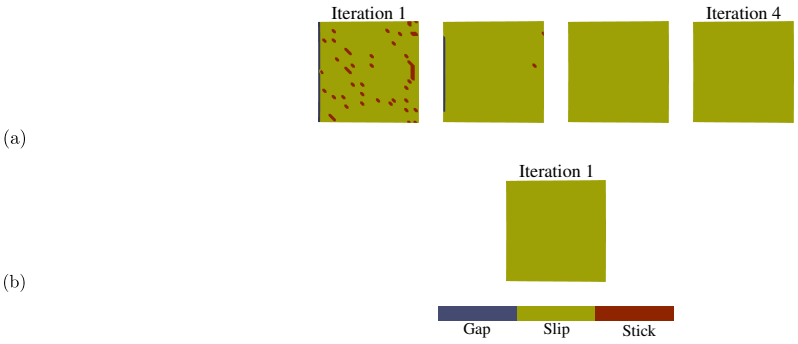


Figure 3.21: The evolution of stick and slip status during the Newton-Raphson iterations of a load increment 50: (a) Case 1; (b) Case 2.

this case study is to understand the additional numerical complexities that manifest under a frictional contact ($\mu = 0.2$) setting compared to the frictionless one. See Algorithm 4 for the various steps involved in resolution of frictional contact problem.

To demonstrate it we consider a cube on a rigid plane similar to the one presented in Section 3.6.2 (the dimensions of the cube and material properties are preserved). However, the rigid surface has more elements along the sliding direction x in order to place the problem under a more realistic numerical setting, i.e. to allow formation of new interface elements as the cube slides along the rigid surface. The cube is represented by a single element [see Fig. 3.23].

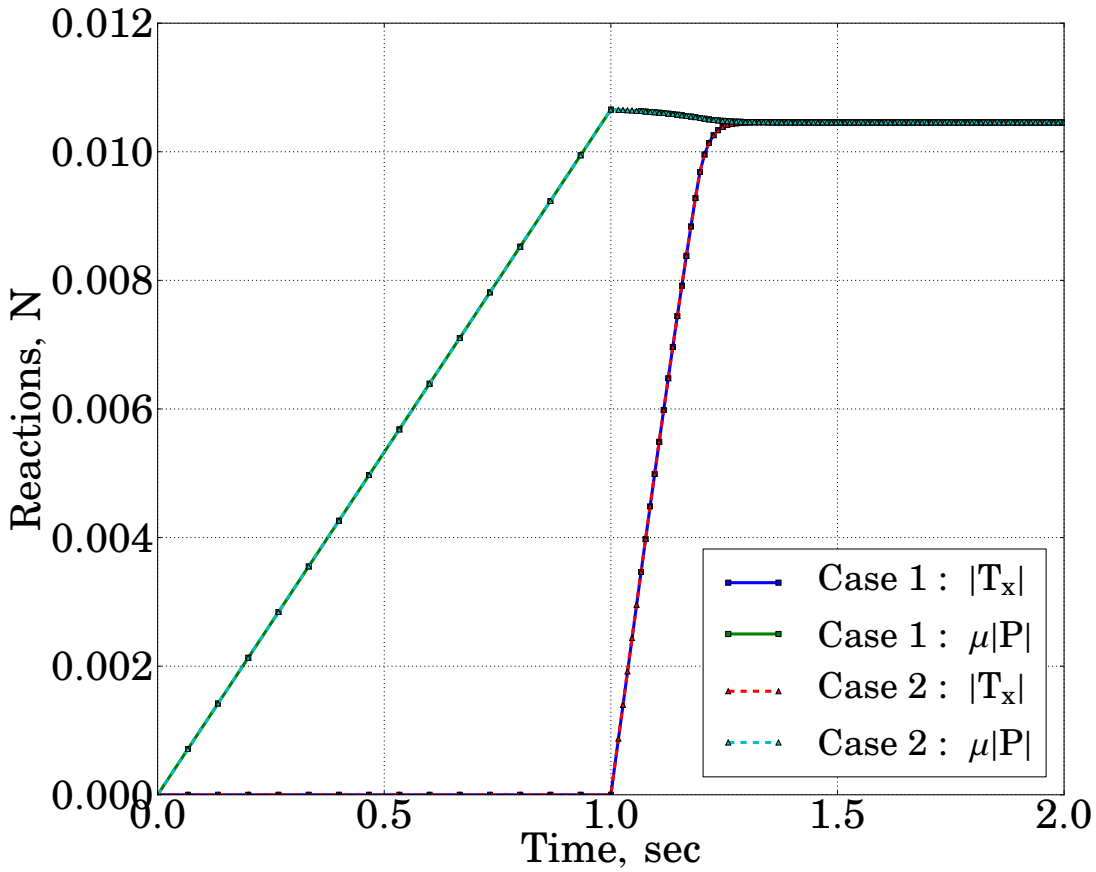


Figure 3.22: Evolution of the scaled normal reaction $\mu|P|$ and tangential reaction $|T_x|$.

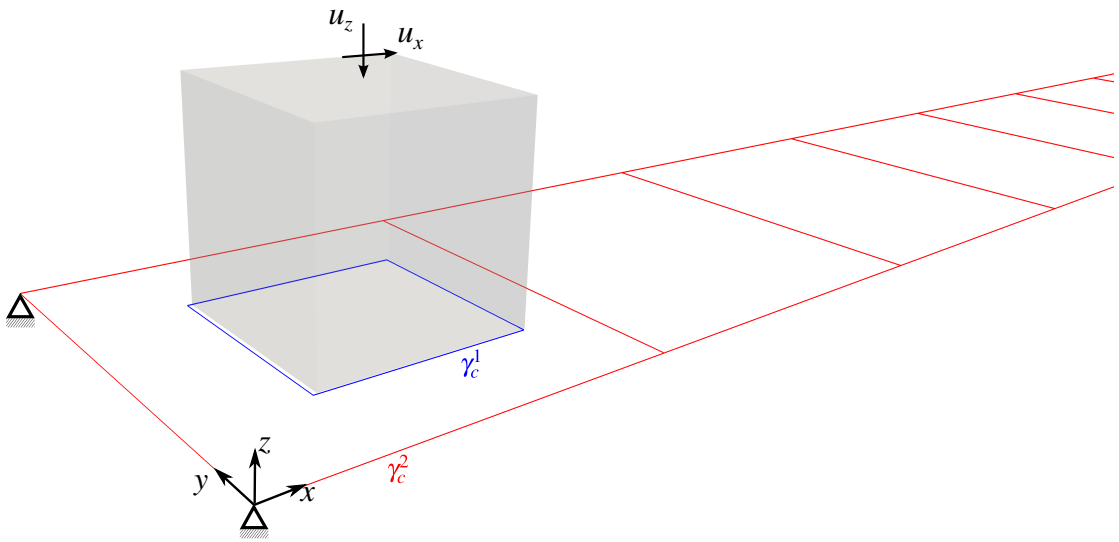


Figure 3.23: FE discretization of the cube and rigid surface along with boundary conditions.

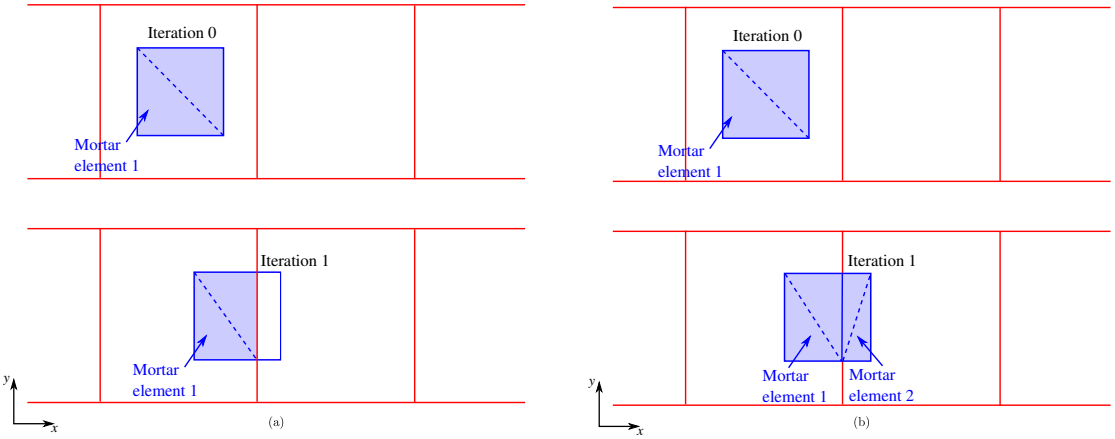


Figure 3.24: Intermediate configurations for load increment n the dotted lines indicate the boundaries of the integration cells formed by triangulation, and the shaded region denotes the mortar domain for the current iteration: (a) incremental update; (b) iterative update.

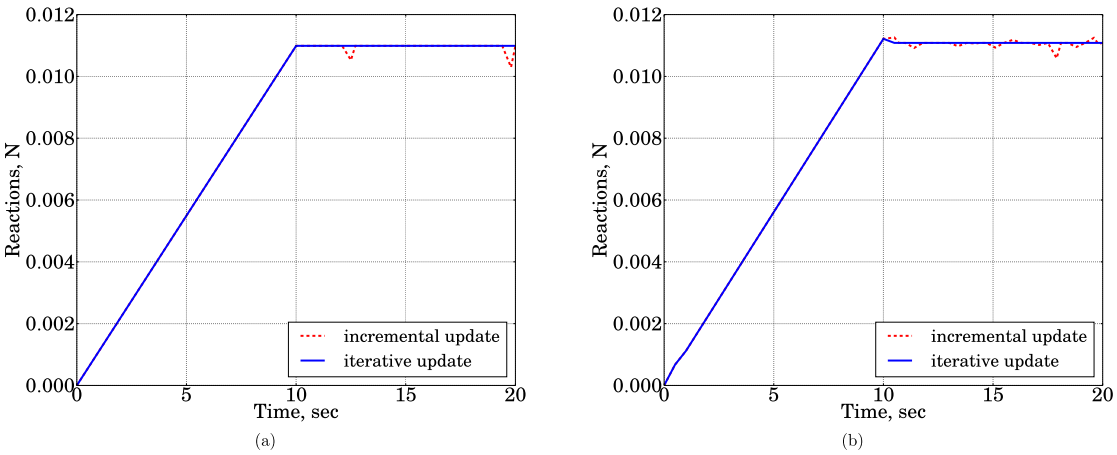


Figure 3.25: Normal reaction forces R_n on the cube for 25 load steps (sequence 2): (a) frictionless case; (b) frictional case.

In a first load sequence ($t = [0, 10]$ s) a vertical displacement of $u_z = 0.01$ mm is applied on the top surface of the cube in 20 load increments and maintained for the subsequent load steps. During the second load sequence ($t = [10, 20]$ s) a horizontal displacement of $u_x = 10$ mm is applied on the top surface in 25 load step increments. The rigid plane surface γ_c^2 is chosen as non-mortar side and γ_c^1 as the mortar side. Fig. 3.24, shows intermediate configurations (iteration 0-configuration at the beginning of the increment, iteration 1-configuration after the application of boundary conditions) of the cube in sliding motion for the load increment n. As shown in Fig. 3.24(a), for the incremental update, the mortar algorithms are performed on the equilibrium configuration of the previous increment represented as Iteration 1. The incremental update avoids formation of a new mortar interface element for the subsequent Newton iterations. However, the mortar integrals are computed on the updated configuration, but inaccurate contact area which is also smaller in this case (shaded region in Fig 3.24(a)). Therefore the reaction drops as seen in Fig. 3.25 and recovers in the next increment. This is reflected in Fig. 3.24(a), where a new mortar interface element is not formed

and equilibrium is achieved on a faulty set of mortar elements. On contrary, for the iterative update, the mortar algorithms of clipping and triangulation are performed iteratively capturing accurately the contact area by forming two mortar elements in this case. The lack of accuracy in capturing the true contact area is reflected in the form of oscillatory and smooth reaction profiles for the incremental and iterative updates, respectively [see Fig. 3.25].

Algorithm 4 Frictional contact problem resolution of an increment

Update boundary conditions
Update λ_n and $\underline{\boldsymbol{\lambda}}_\tau$ from previous converged increment
CREATE_MORTAR_ELEMENTS( ) ▷ involves detection, projections, clipping, triangulation.

procedure NEWTON_RAPHSON_LOOP( )
  while ($||\underline{\boldsymbol{r}}||_{\text{primal}} > \text{tol1}$ and $||\underline{\boldsymbol{r}}||_{\text{dual}} > \text{tol2}$) **do**
    if (iterative-update) **then**
      CREATE_MORTAR_ELEMENTS( ) ▷ For iterative update, the mortar interface elements are created at the beginning of each Newton iteration.
    end if
    for `el in mortar_elements` **do** ▷ loop over each mortar interface element.
      $r^{\text{el}} = 0$
      D_{lm}, M_{li} ←EVALUATE_INTEGRALS( ) (3.14), (3.15)
      for (l=0 ; l<L ; l++) **do** ▷ Loop over each mortar side node
        $\bar{g}_n^l$ ←COMPUTE_GAP( )(3.17)
        if (el is NEW) **then** ▷ verify if the pair of mortar, non-mortar segments have formed a mortar element in the previous converged configuration
          $\underline{\overset{\circ}{\boldsymbol{g}}}_\tau^l = 0$
        else
          $\underline{\overset{\circ}{\boldsymbol{g}}}_\tau^l$ ←RETRIEVE_INCREMENTAL_SLIP( ) (3.27)
        end if
        $\hat{\lambda}_n^l$ ←COMPUTE_AUG_LM( )(3.35)
        $\underline{\hat{\boldsymbol{\lambda}}}_\tau^l$ ←COMPUTE_AUG_LM( )(3.36)
        if ($\hat{\lambda}_n^l \leq 0$) **then**
          if ($||\underline{\hat{\boldsymbol{\lambda}}}_\tau^l|| \leq -\mu\hat{\lambda}_n^l$) **then**
            r^{el}+ =EVALUATE_RESIDUAL( ) (3.32)
          else
            r^{el}+ =EVALUATE_RESIDUAL( ) (3.33)
          end if
        else
          r^{el}+ =EVALUATE_RESIDUAL( ) (3.34)
        end if
      end for
      if ($\hat{\lambda}_n^l \leq 0$) **then**
        if ($||\underline{\hat{\boldsymbol{\lambda}}}_\tau^l|| \leq -\mu\hat{\lambda}_n^l$) **then**
          EVALUATE_K^{el}( ) (3.40)
        else
          EVALUATE_K^{el}( ) (3.41)
        end if
      else
        EVALUATE_K^{el}( ) (3.42)
      end if
    end for
    if (iterative-update) **then**
      REINITIALIZE_INTERFACE( ) ▷ For iterative update, the mortar interface elements are deleted at end of each Newton iteration.
    end if
  end while
end procedure

3.7 Numerical examples

In this section we present few selected numerical examples, to demonstrate the robustness and performance of the implemented mortar discretion method. Few examples and applications are shown in Fig. 3.26; among them 2D and 3D patch tests and sub-structuring simulations for mesh tying; and for contact problems we solved 2D and 3D Charpy fracture tests simulated for highly non-linear elasto-viscoplastic anistropic pipeline steel with integrated Gurson-type ductile fracture model, wheel-tyre-road rolling contact, and various modifications of 2D and 3D ironing problems. Those are only a few examples which were used to validate and test the elaborated mortar tying and contact framework. Below we present different selected tests.

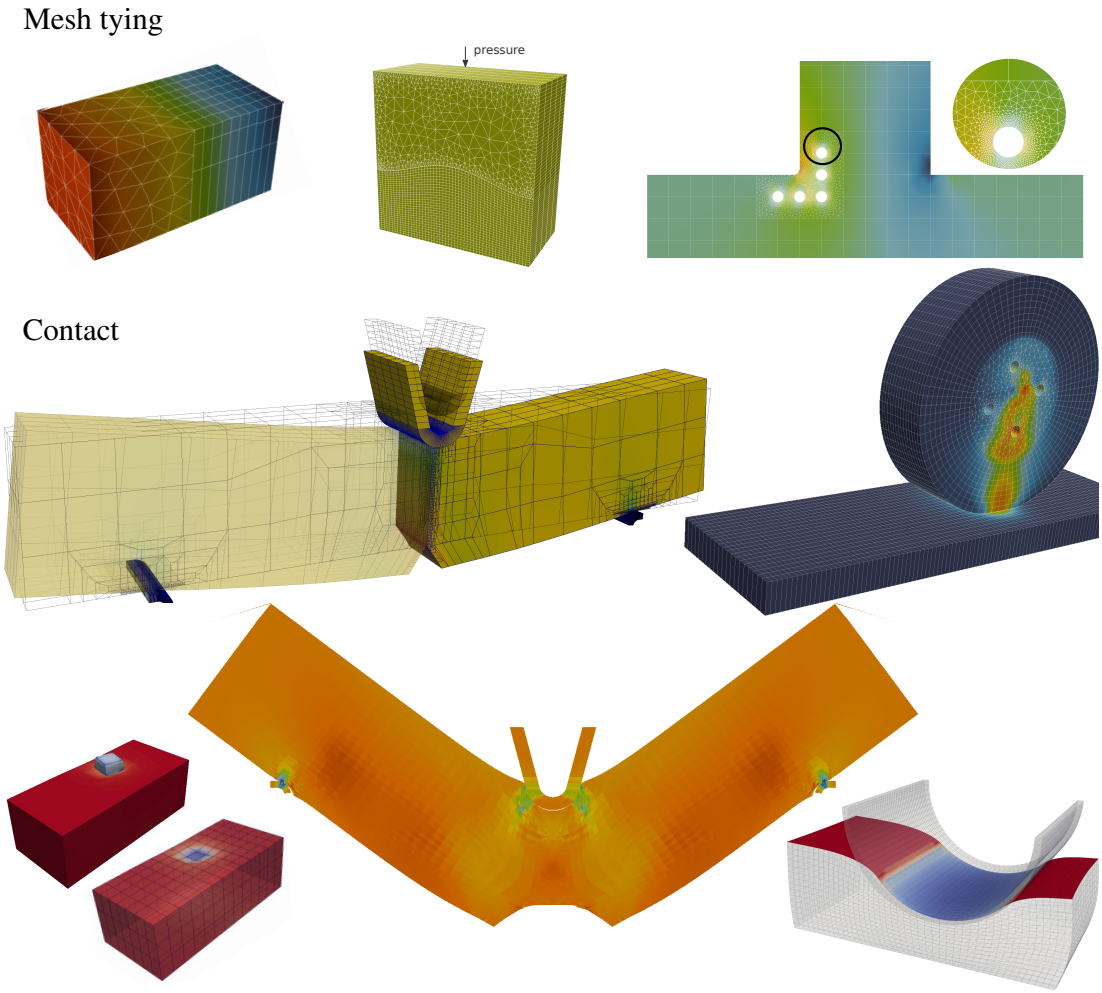


Figure 3.26: Few interface problems solved within Z-set finite element suite using mortar method.

3.7.1 Tying of multiple domains

In this example, we illustrate the mesh tying capabilities of the mortar methods. The problem set-up involves multiple domains tied along the common shared interfaces. The boundary conditions are chosen in such way as to result in a constant stress field in the domain. Such a set-up is often referred to as patch test

[see Fig. 3.27(a)]. A displacement u_y is applied on the top surface. The sides are fixed laterally and the bottom is fully fixed. The applied displacement results in a constant stress field σ. The solutions for both the 2D and 3D cases using the mortar implementation are demonstrated in Fig. 3.27(b).

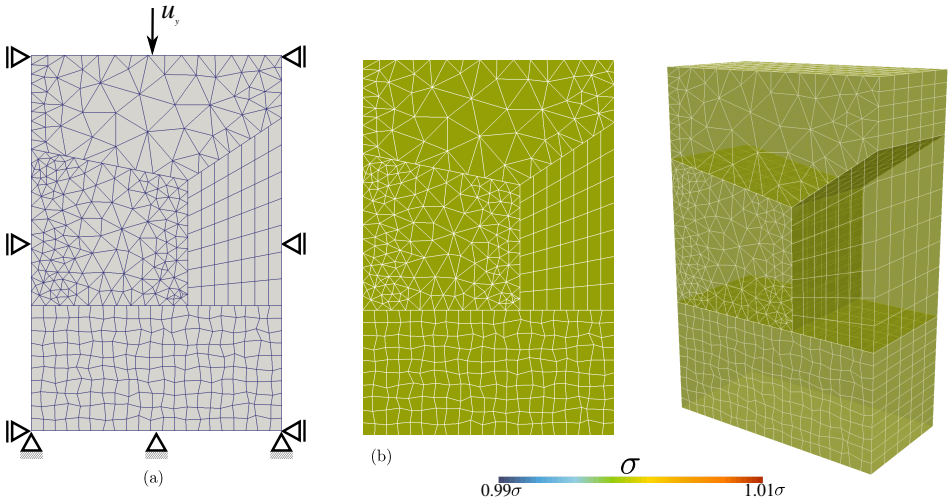


Figure 3.27: Patch test: (a) problem set-up; (b) constant stress field σ.

3.7.2 3D cylinder plate tying

In this example we consider a square plate of side $L = 40$ mm and cylinder (outer radius $R = 8$ mm, thickness $t = .5$ mm, height $h = 20$ mm) with its axis perpendicular to the plate [see Fig. 3.28(a)]. Same linear elastic material properties $E = 100$ MPa and $\nu = 0.3$ are used for the cylinder and plate. The lateral surfaces of the plate are fixed and an axial displacement $u_z = 10$ mm is applied on the top surface of the cylinder. The tying constraints are imposed between the bottom surface of the cylinder and top surface of the plate. Strict adherence to the imposed tying constraints is reflected in the smooth axial displacement and stress fields shown in Fig. 3.28(b) and (c), respectively.

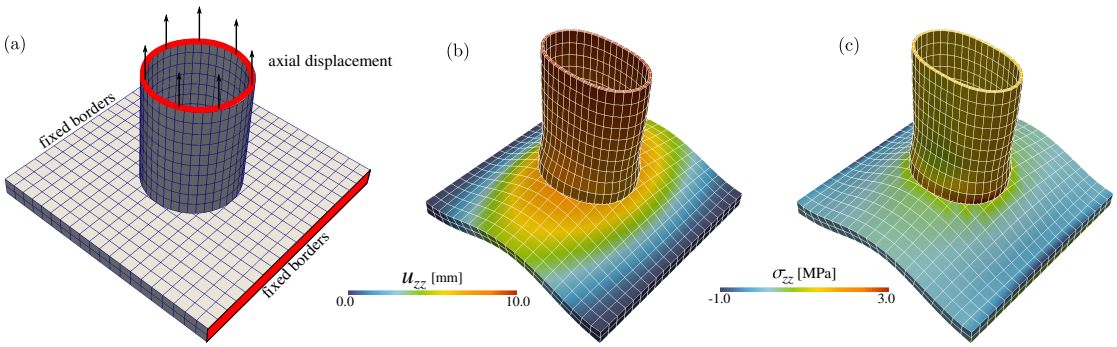


Figure 3.28: Plate and cylinder tied at the interface: (a) problem set-up with boundary conditions; (b) u_z field; (c) σ_{zz} field.

3.7.3 Periodic boundary conditions

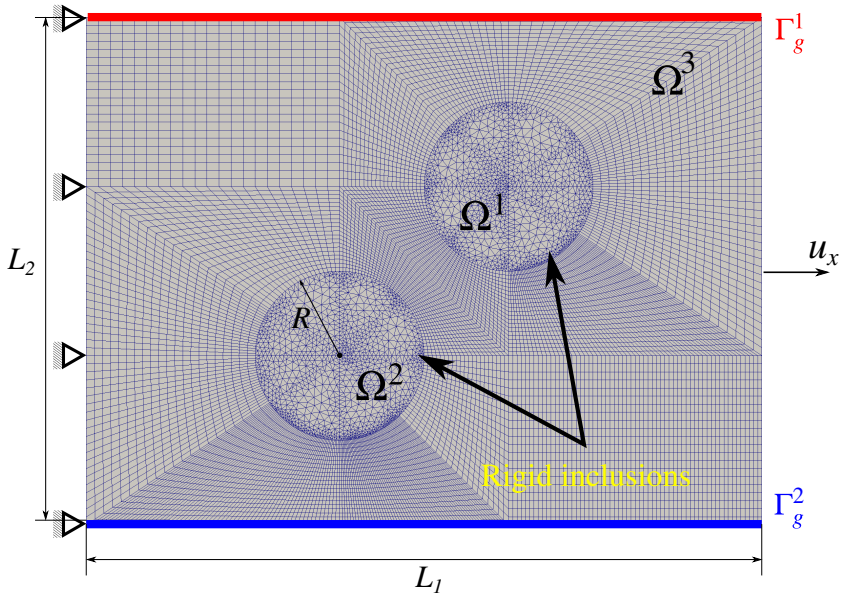


Figure 3.29: Discretized problem setup for periodic boundary conditions applied along Γ_g^1 and Γ_g^2.

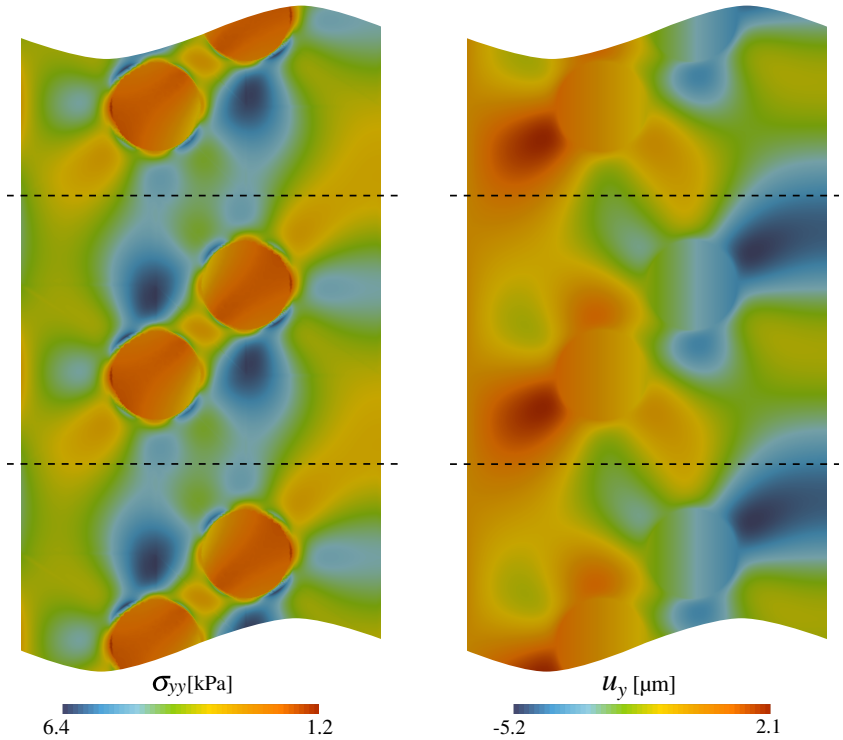


Figure 3.30: Smooth stress σ_{yy} and displacement u_y fields across periodic boundary regardless non-conformal discretizations.

In this example we demonstrate the applicability of the mortar mesh tying algorithms to impose periodic boundary conditions. For this we consider a plate, with two stiff circular inclusions [see Fig. 3.29]. Linear elastic material properties are

assigned to both the inclusions ($E_1 = E_2 = 1000$ MPa, $\nu_1 = \nu_2 = 0.31$) and the plate ($E_3 = 1$ MPa, $\nu_3 = 0.31$). The geometric dimensions are: $L_1 = 8$ mm, $L_6 = 8$ mm and $R = 1$ mm. The periodic boundary conditions are imposed between the boundaries Γ_g^1 and Γ_g^2 separated by a distance L_2. Smooth and continuous fields of displacements and stresses across the periodic boundary can be seen in Fig. 3.30.

3.7.4 Frictional beam contact

In this example we demonstrate the robustness of the proposed mortar contact algorithm within the context of large deformations with finite frictional sliding. The setup consists of a curved beam ($E = 2250$ MPa, $\nu = 0.125$) and a straight beam ($E = 2700$ MPa, $\nu = 0.35$), see Fig. 3.31 [Yang et al., 2005, Gitterle et al., 2010]. The straight beam is simply supported. The top left of the curved beams is fixed in y direction, and right top of the beam is subjected to a vertical displacement of $u_y = 1.2t$. Simultaneously, a horizontal displacement of $u_x = 2.0t$ is applied on both left and right top surfaces of the beam. At time $t_{\max} = 8$ s, the maximal displacements are $u_{x\max} = 16$ mm and $u_{y\max} = 9.6$ mm. Coulomb friction with friction coefficient $\mu = 0.5$ is used. The configurations at $t = 2.0, 4.0, 6.0$ and 8.0 s, are shown in Fig. 3.32. The horizontal reaction forces at the left corner of the flat beam are compared for both the cases of curved beam and flat beam considered as mortar side with the original results obtained in [Yang et al., 2005] [see Fig. 3.33]. The comparison shows good agreement and also the choice of mortar and non-mortar sides is negligible as expected.

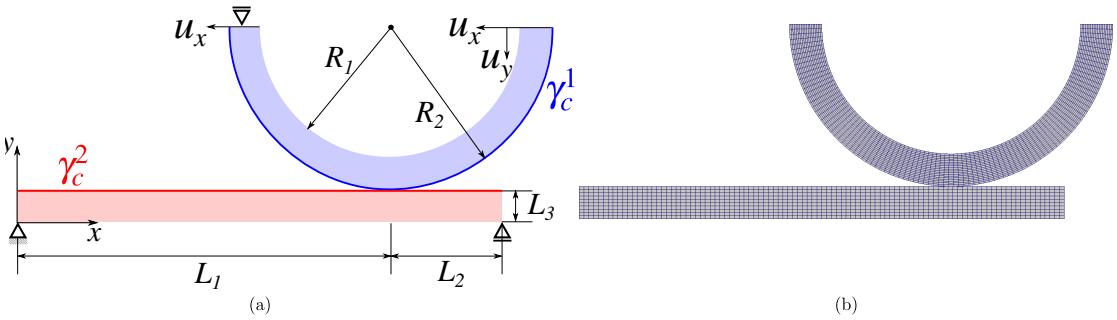


Figure 3.31: (a) Problem setup of the frictional beam contact ($R_1 = 8$, $R_2 = 10$, $L_1 = 23$, $L_2 = 7$, $L_3 = 2$ mm); (b) FE mesh of the beams.

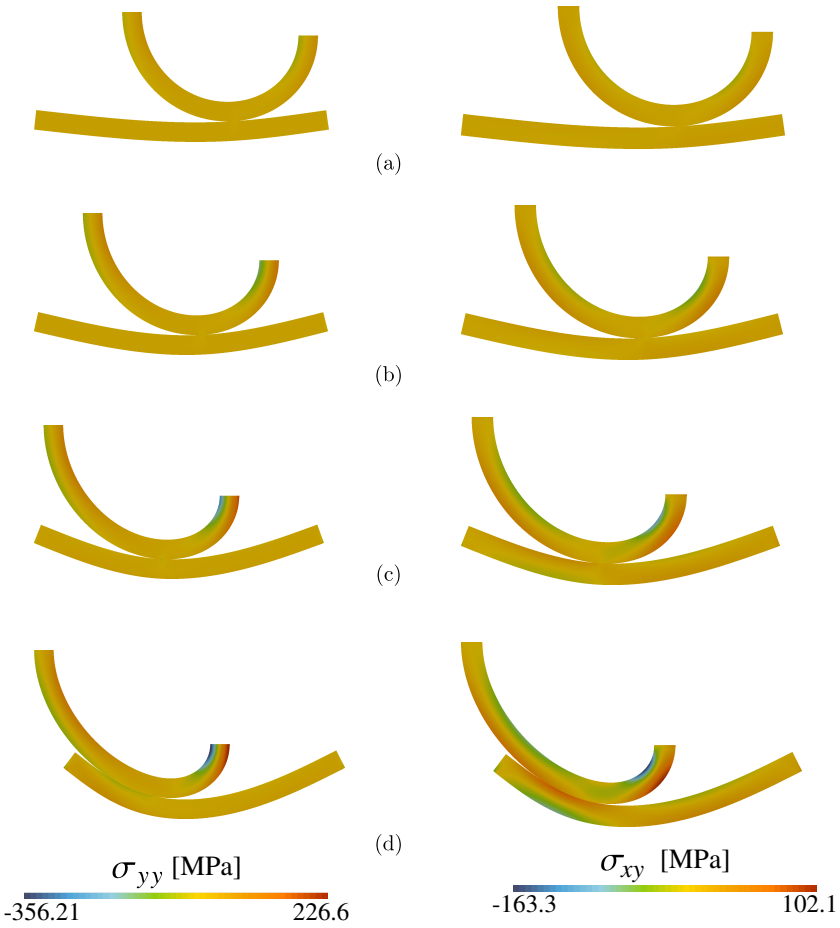


Figure 3.32: Deformed configurations: (a) $t = 2$; (b) $t = 4$; (c) $t = 6$; (d) $t = 8$ s.

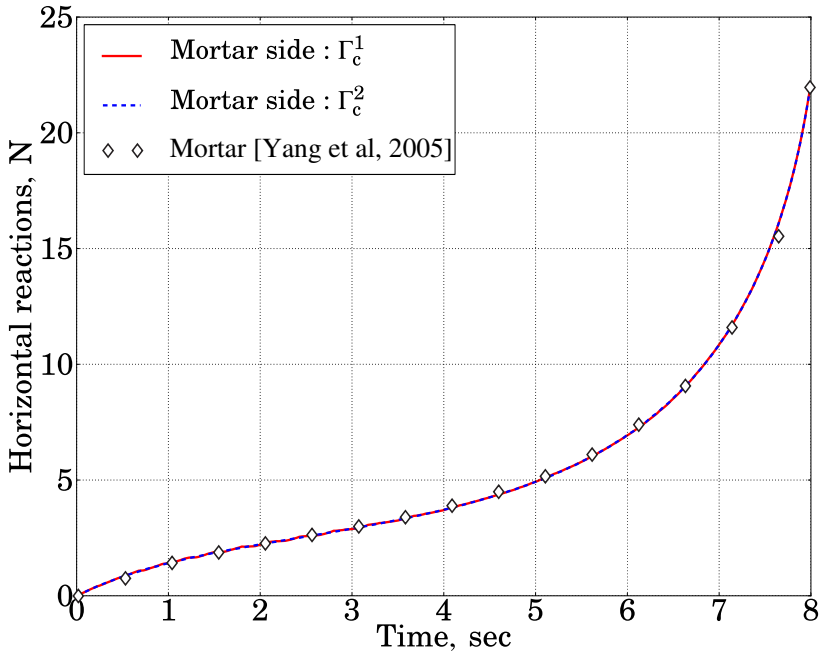


Figure 3.33: Comparison of the horizontal reaction forces at the left corner of the flat beam with solution from [Yang et al., 2005]: for both the cases of curved beam and flat beam considered as mortar side.

3.7.5 Axisymmetric deep cup drawing

The underlying forming process is contact driven, and is numerically challenging as it involves all the non-linearities: geometric (large deformations within the updated Lagrangian framework), material (elasto-visco-plastic model with exponential hardening) and frictional contact. We illustrate the method's robustness in terms of handling all the non-linearities occurring simultaneously. This particular problem was also considered in [Rousselier et al., 2009, Yastrebov, 2013]. The problem setting and the FE mesh are shown in Fig. 3.34 and Fig. 3.35, respectively. The geometric dimensions are: die diameter $d = 97.46$ mm, fillet radius $R = 12.7$ mm, die opening $w = 101.48$ mm, sheet diameter $L = 158.76$ mm and sheet thickness $h = 1.6$ mm. A mixed force-displacement type boundary conditions are employed: a pressure $p = 1.86e^{-2}$ MPa is applied on the holding surfaces during $t \in [0, 100]$ s and maintained for the complete duration of the simulation. A vertical displacement $u_y = 42.0$ mm is applied on the die during $t \in [0, 420]$ s. This is then followed by unloading, where the die is pulled back until $u_y = 35$ mm during $t \in [420, 500]$ s.

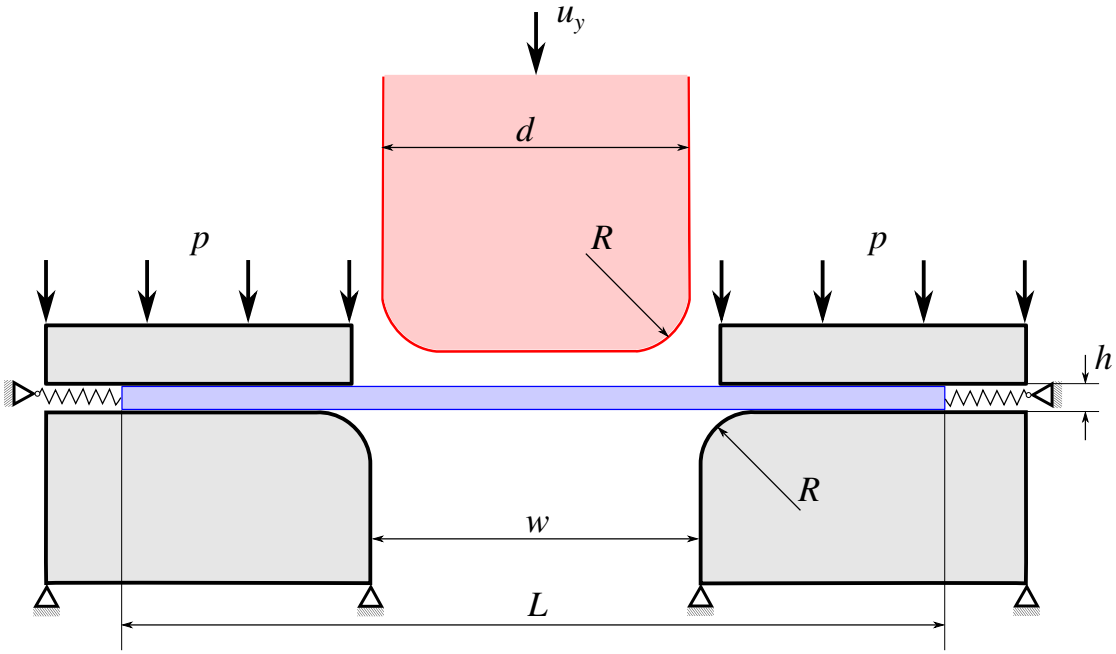


Figure 3.34: Problem setting and boundary conditions for the deep draw simulation [Yastrebov, 2013].

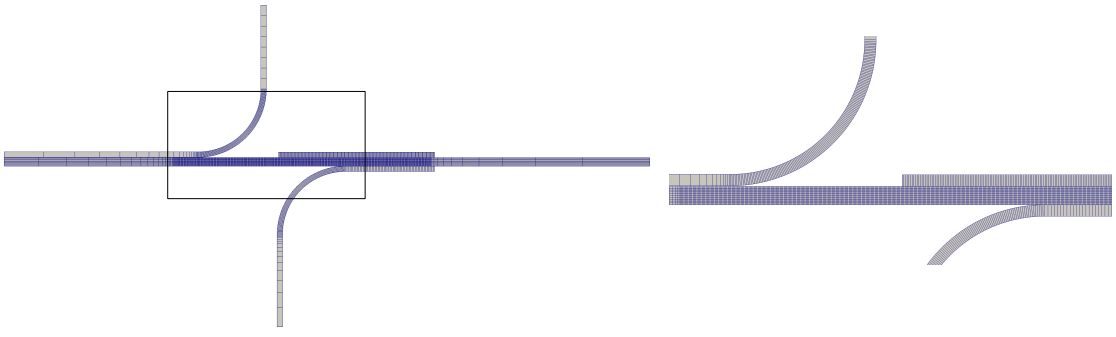


Figure 3.35: Finite element mesh for the axisymmetric deep cup drawing problem.

In order to stabilize the solution a soft spring (700 times softer than the material of the sheet) is attached to the edge of the sheet. A friction coefficient of $\mu = 0.1$ is used. For this simulation we use local tuning of augmentation parameters according to (3.37) and also use different augmentation parameters for the normal and tangential directions ($\varepsilon_\tau = .5 * \varepsilon_n$). An elasto-visco-plastic material model with exponential hardening is used for the sheet.

Young's modulus $E = 69.\,\text{GPa}$
Poisson's ratio $\nu = 0.33$
Yield criterion: von Mises $R_0 = 0.22\,\text{GPa}$
Norton creep power law: $\dot{\lambda} = \langle f/K \rangle$, $K = 0.5\,\text{GPa}$, $n = 7$
Isotropic power hardening law $R = R_0 + K(e_0 + p)^n$, $K = 0.99\,\text{GPa}$, $e_0 = 7e-4$, $n = 7$

Fig. 3.36, shows the accumulated plastic strain at various time steps. As can be seen, the necking of the sheet happens for the drawing depth greater than 33.5 mm $u_y \in [33.5, 42.0]$ mm. Fig. 3.37 shows the reaction forces in the vertical direction on the die. On an average each load step takes 5 iterations (for a tolerance of 0.1) to converge.

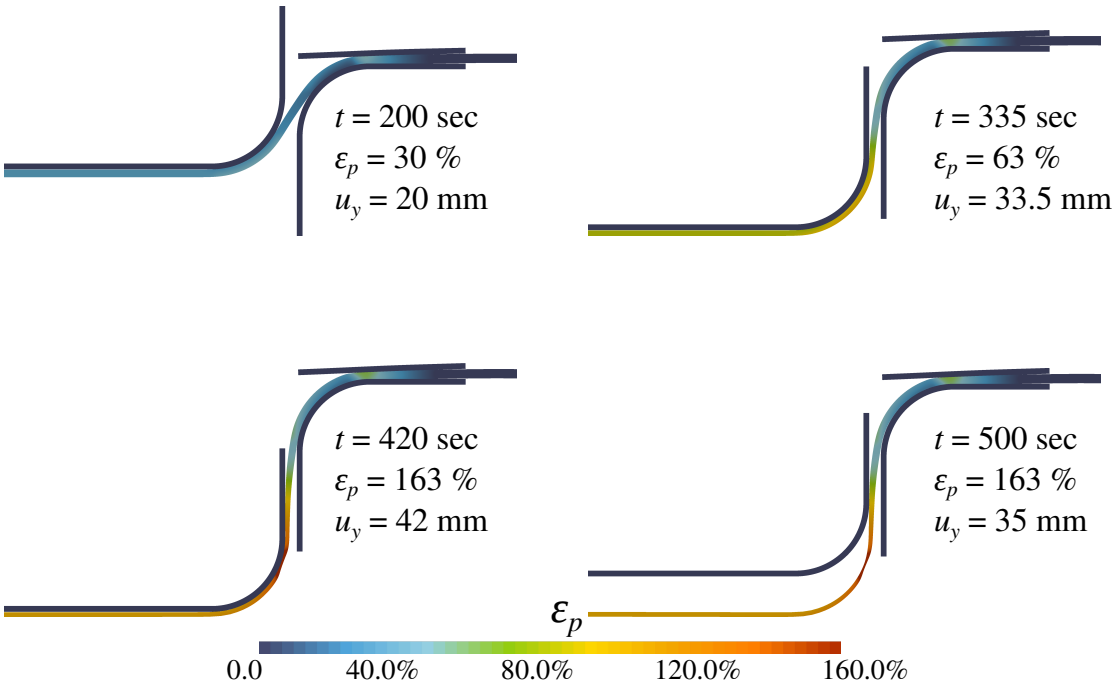


Figure 3.36: Distribution of accumulated plastic strain at different time steps.

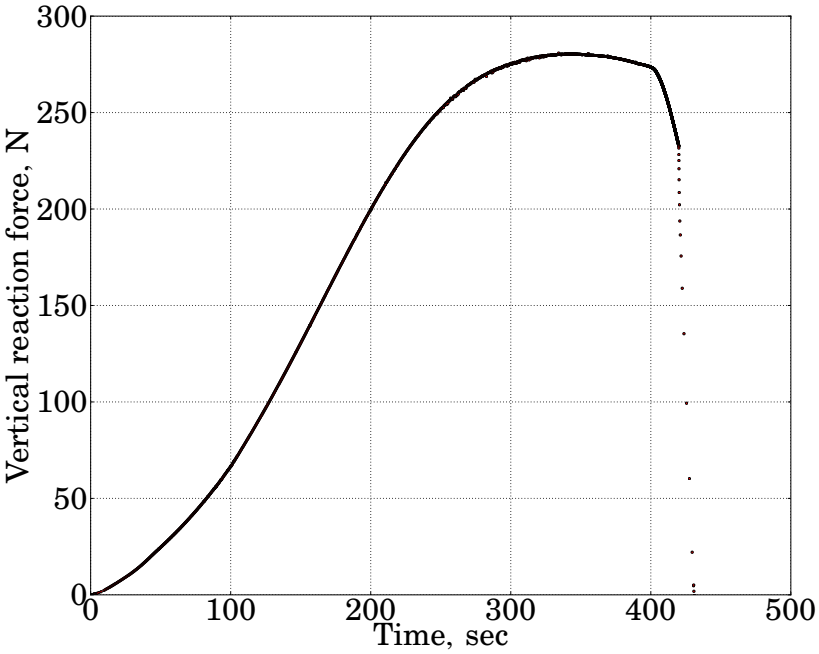


Figure 3.37: Evolution of the reaction on the punch with time.

3.7.6 3D Hertzian contact

A Hertz contact problem with a sphere pressed into a flat deformable support is considered here. The analytical solution to this problem is derived under few assumptions, that include small deformations, infinite parabolic contacting solids, and a very small radius of contact area compared to the radius of sphere [Hertz, 1882]. The geometry settings for the problem are shown in Fig. 3.38 ($L_1 = L_5 = 400$, $L_2 = 250$ and $R = 300$ mm). The sphere radius R is truncated when its radius reaches $L_4 = 75$ mm in the yz plane. This additional approximation is introduced to reduce the problem size.A layer of thickness $L_3 = 20$ mm is added on the top of the sphere's segment. A linear elastic material with Young's modulus $E = 100$ GPa and Poisson's ration $\nu = 0.3$ is used for both the sphere and the support. Augmentation parameter $\varepsilon_n = 1e^6$ is used for the simulation. We use this set-up to carry out a mesh convergence study. For this purpose we generate seven meshes with average mesh sizes in the contact zone of : $h \in [3, 2, 1.6, 1.2, 1.0, 0.8, 0.6]$ mm. The coarsest mesh ($h = 3.0$ mm) has 18180 primal DoFs and the finest mesh ($h = 0.6$ mm) has 422766 primal DoFs. The mesh refinements are shown in Fig. 3.39.

A pressure $p = 5.659$ MPa, is applied on the surface of the truncated sphere. This results in a force $P = 100$ kN. The motion of the bottom face of the block is restricted in all directions. For a qualitative comparison of the obtained numerical results we use the analytical solution. For this purpose, the effective elastic modulus E^* is introduced as below:

$$E^* = \frac{E}{2(1-\nu^2)} \approx 54.9\,\text{GPa}. \tag{3.61}$$

The resulting contact surface is a circle with radius given as:

$$a = \left(\frac{3PR}{4E^*}\right)^{\frac{1}{3}} = 7.43\,\text{mm}. \tag{3.62}$$

The contact pressure distribution for this circle centered at $(x, y) = (0, 0)$ is give by:

$$p(x, y) = p_0 \sqrt{1 - \left(\frac{x^2 + y^2}{a^2}\right)}, \tag{3.63}$$

where the maximum contact pressure p_0 is given as:

$$p_0 = \left(\frac{6PE^{*2}}{\pi^3 R^2}\right)^{\frac{1}{3}} \approx 865.85\,\text{MPa}. \tag{3.64}$$

The Fig. 3.40, shows the results obtained using the fine mesh ($h = 0.6$ mm). The numerically obtained contact area radius and the maximum stress σ_{zz} bear an error of $\approx 1.5\%$ in comparison to the analytical solution. The error can be attributed to the geometrical assumptions made for the numerical set-up. A rather good convergence to the analytical solution for p_0 is demonstrated in Fig. 3.41.

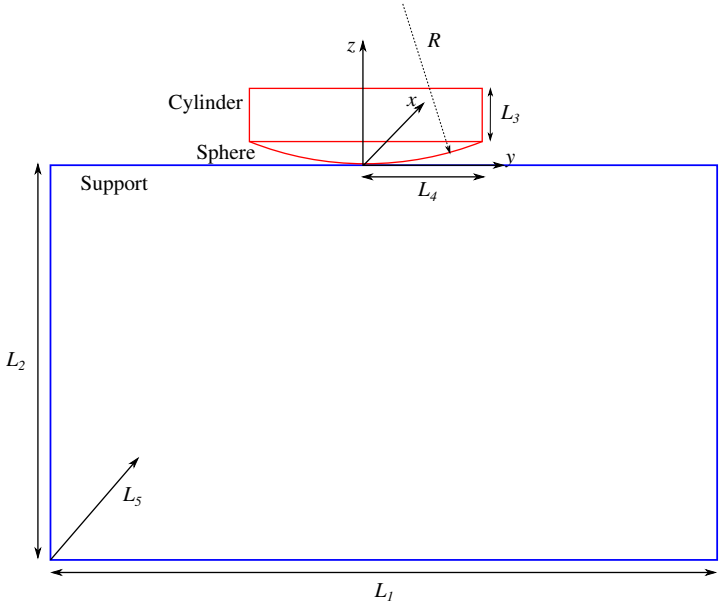


Figure 3.38: Geometry view in plane parallel to zy coordinate plane.

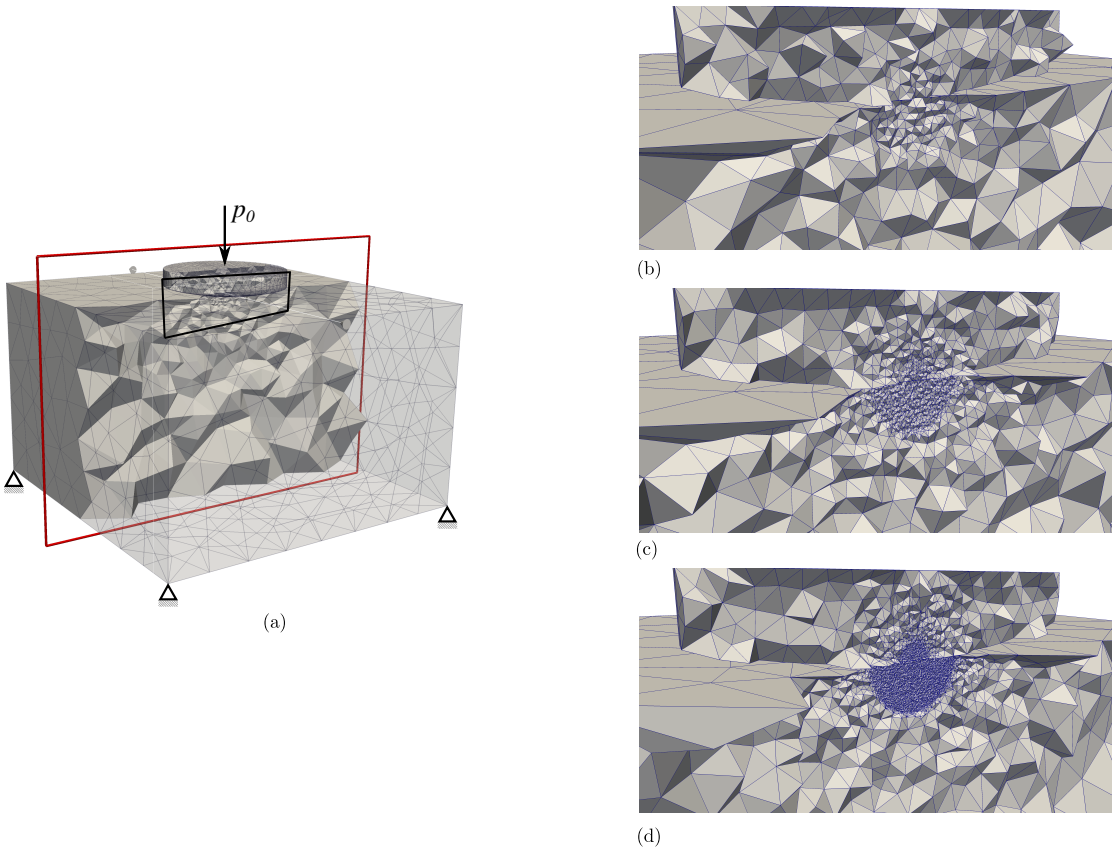


Figure 3.39: (a) Finite element mesh with applied boundary conditions; (b) coarse mesh refinement near the contact zone ($h = 3.0$ mm); (c) intermediate mesh refinement near the contact zone ($h = 1.6$ mm); (d) very fine mesh refinement near the contact zone ($h = 0.6$ mm).

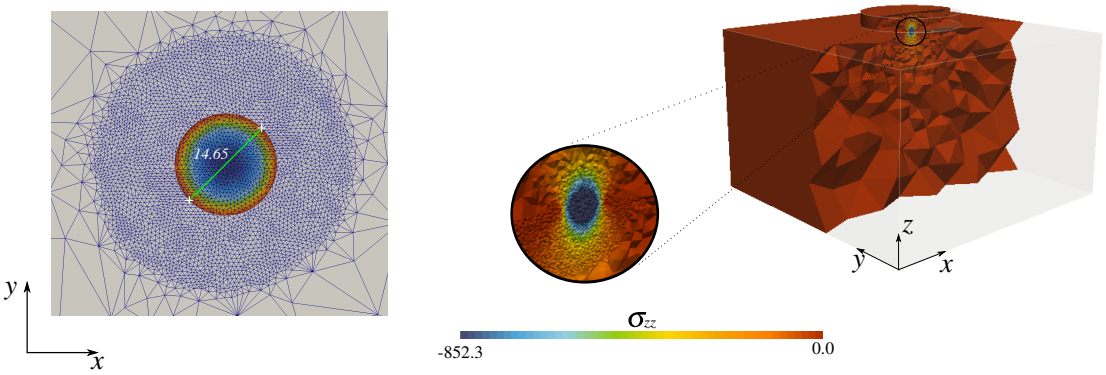


Figure 3.40: 3D Hertzian problem results: (a) radius of the contact area; (b) the contour stress plot of σ_{zz}.

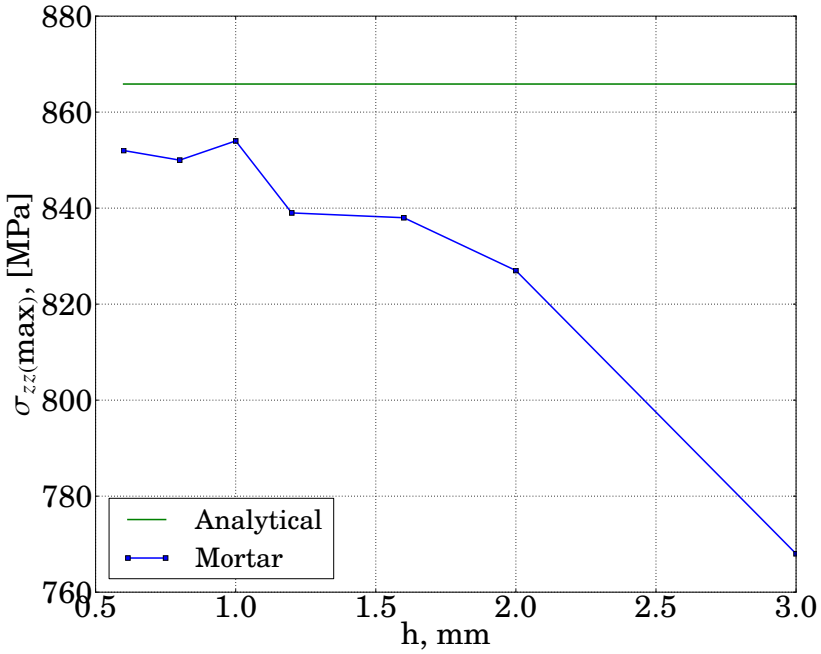


Figure 3.41: The mesh convergence study for the maximum contact stress $\sigma_{zz}(\max)$.

3.7.7 Frictional sliding of a cube on a rigid plane

Here we revisit the frictional sliding of a cube on a rigid plane problem, considered in Section 3.6.2 [see Fig. 3.18(a)]. Additionally, to the coefficient of friction $\mu = 0.2$ we consider a case of higher coefficient of friction $\mu = 0.7$, between the cube and rigid plane. The case with $\mu = 0.7$ leads to partial detachment of contacting surfaces during the sliding motion. These two cases are used to illustrate the nodal transitions between the three contact statuses namely: gap, slip and stick, during the course of the simulation. The material, geometrical and FE discretization set-up remains the same as in Section 3.6.2. However the boundary conditions are slightly modified. In a first load sequence ($t = [0, 1]$ s) a vertical displacement of $u_z = 0.05$ mm is applied on the top surface of the block in 10 increments. During the second load sequence ($t = [1, 2]$ s) a horizontal displacement of $u_x = 0.33$ mm is applied on the top surface in 100 increments.Augmentation parameters $\varepsilon_n = \varepsilon_\tau = 10000$ are used for the simulation.

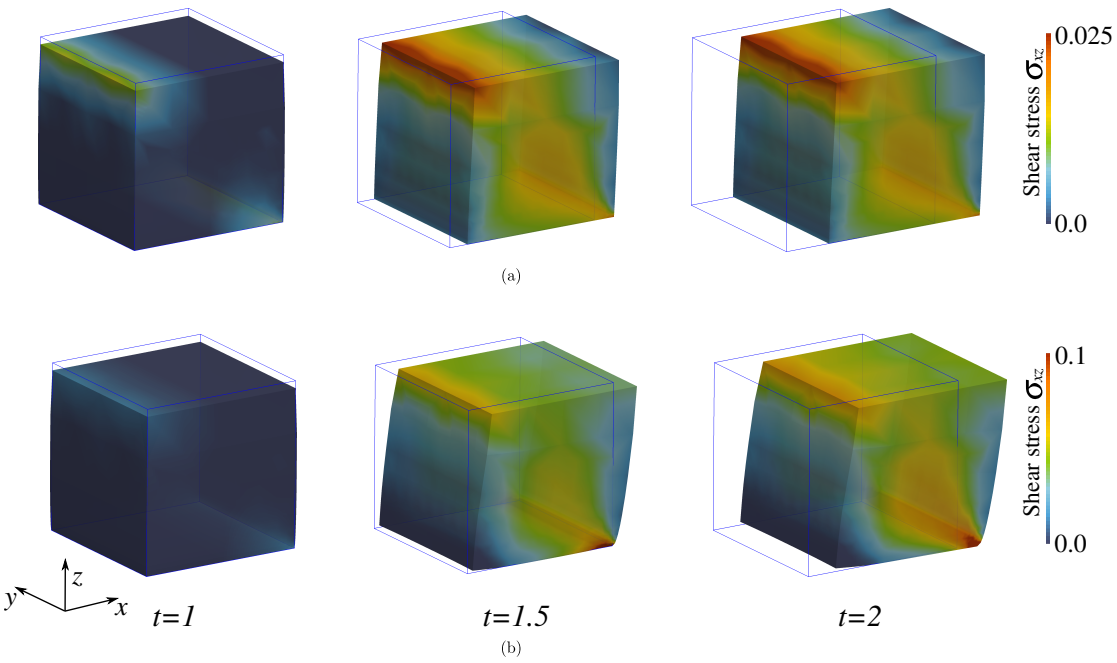


Figure 3.42: Initial configuration (blue line) and the contour stress plots of shear stress σ_{zz} at $t = 1, 1.5, 2$ s for: (a) $\mu = 0.2$; (b) $\mu = 0.7$.

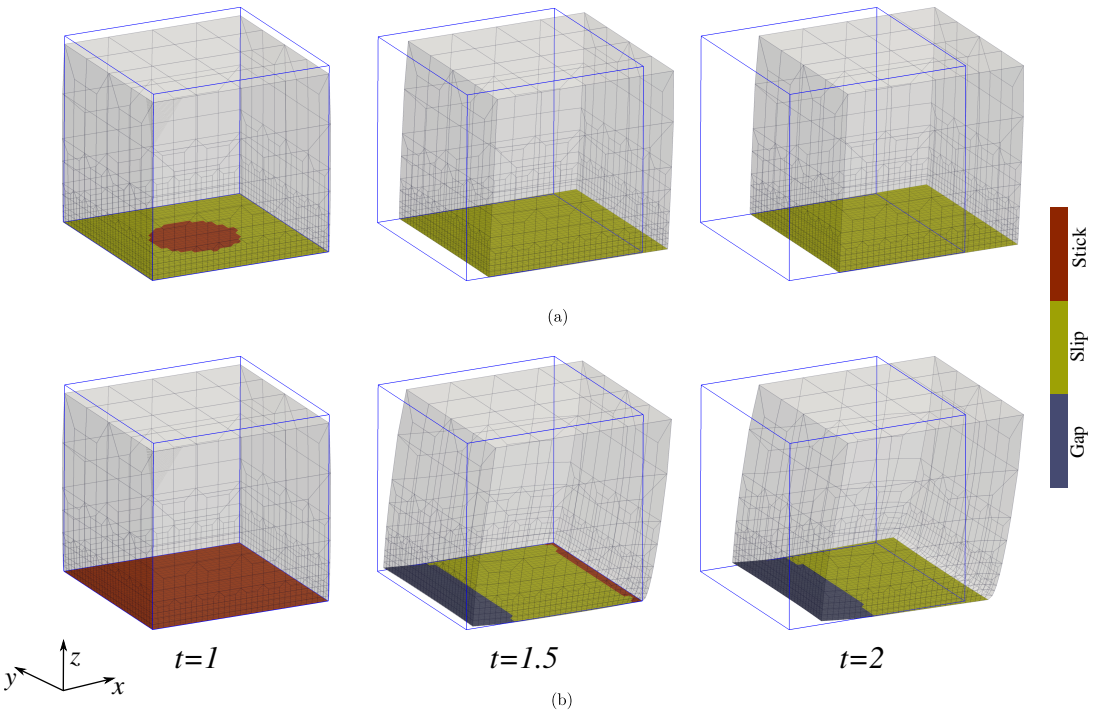


Figure 3.43: Initial configuration (blue line) and the contact status of the mortar side nodes at $t = 1, 1.5, 2$ s for: (a) $\mu = 0.2$; (b) $\mu = 0.7$.

Fig. 3.42, shows the shear stress distribution on the deformed configurations at various time intervals of the simulation. The case with $\mu = 0.7$ is qualitatively different from the one with $\mu = 0.2$, as it leads to detachment in the contact zone. As a consequence, there is some stress redistribution in the contact interface and an decrease (for the case with $\mu = 0.2$) in the normal reaction due to the coupling

between normal and tangential reactions, see Fig. 3.44. However, for the case with $\mu = 0.7$, as the trailing edge looses contact results in an increase in the contact pressure (due to redistribution over a smaller contact area).

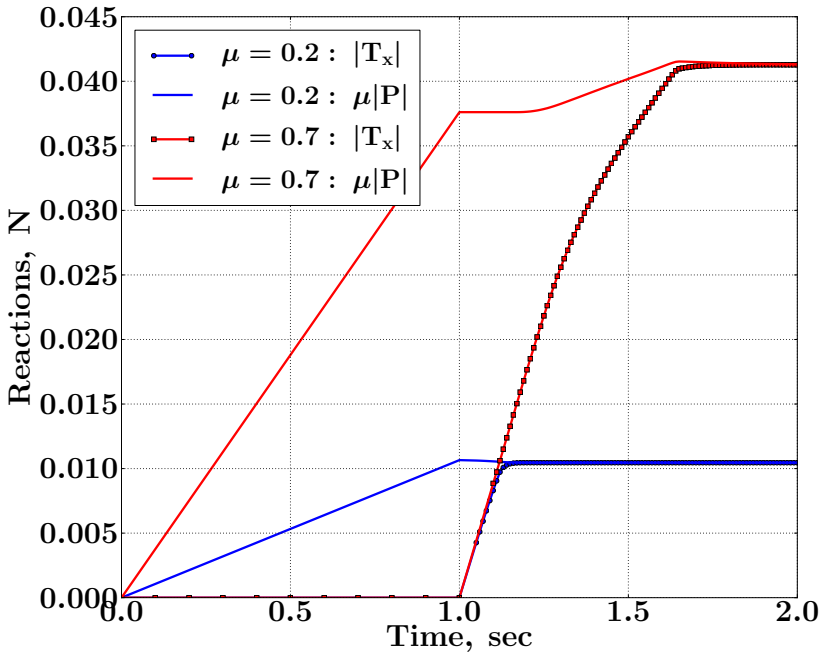


Figure 3.44: Evolution of the scaled normal reaction $\mu|P|$ and tangential reaction $|T_x|$.

3.7.8 Frictional contact: Turbine blade disk

In this example we consider a frictional contact between a model blade disk assembly, a critical component of the aircraft engine. Accurate analysis of the stress levels induced due to the complex loading cycles is very crucial for efficient design of these components and their life cycle assessment. Here, we consider the loading conditions shown in Fig. 3.45. There are a total of four loading sequences. Displacements are applied on the top surface of the blade. During the first loading sequence $t \in [0, 1]$ sec, a vertical displacement $u_z = 0.35$ mm is applied. For the second load sequence $t \in [1, 2]$ sec, a vertical displacement $u_z = -0.70$ mm is applied and maintained for the rest of the load sequences. For the third load sequence $t \in [2, 3]$ sec, a displacement $u_y = -0.75$ mm is applied. For the fourth load cycle $t \in [3, 4]$ sec, a displacement $u_y = 1.50$ mm is applied. Through out the load sequences, the outer surfaces of the disk's part are fixed as shown in Fig. 3.45, while symmetry boundary conditions are applied on the center surface of the blade and the disk. Same linear elastic material properties are assigned for both the disk and blade ($E = 1$ GPa, $\nu = 0.3$). A coefficient of friction $\mu = 0.2$ is used. The geometric dimensions are: $L^1 = 35$ mm, $L^2 = 12$ mm, $L^3 = 14$ mm, $L^4 = 10$ mm and $L^5 = 5$ mm. Augmentation parameters $\varepsilon_n = \varepsilon_\tau = 10000$ are used for the simulation.

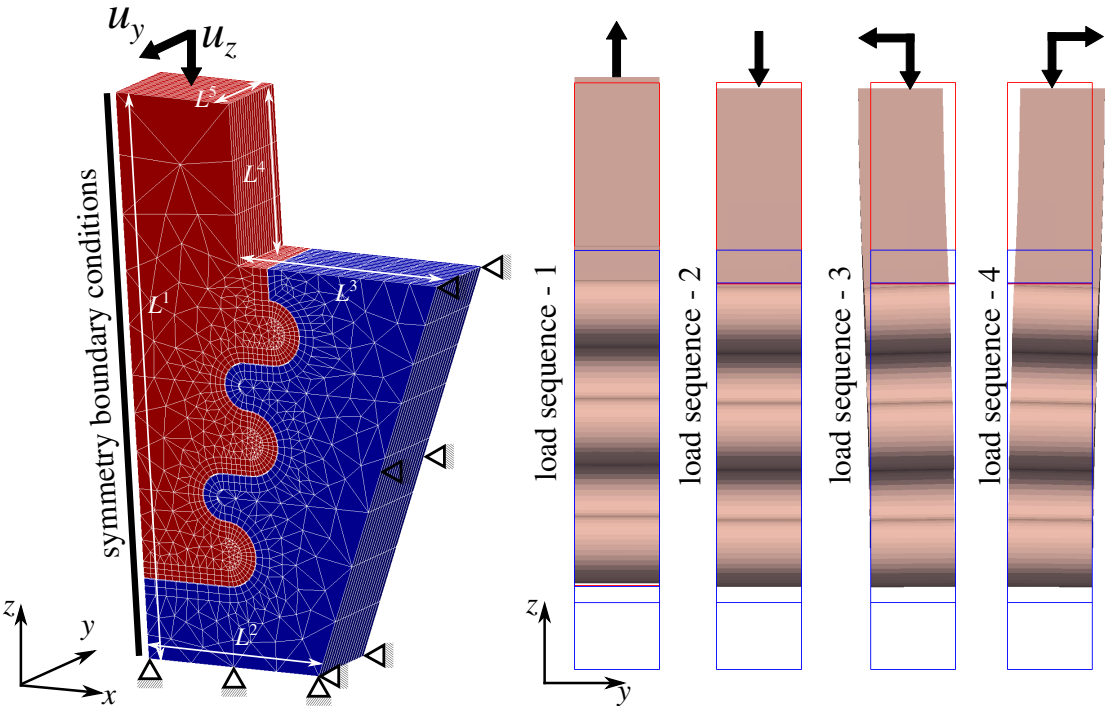


Figure 3.45: Disk blade contact problem set-up; the deformed configurations at various load sequences.

The following aspects of this problem set-up are interesting to test the mortar implementation:

- handling multiple separate contact zones.
- handling status change from no-contact to contact and in turn alternating from slip and stick statuses.

- handling curved interfaces in 3D is particularly challenging in terms of ensuring accurate clipping and triangulation on auxiliary planes.

The stress profiles achieved are smooth, which is important for the analysis of such crucial components. This is achieved in spite of the computational challenges in terms of problem resolution and continuously evolving contact surfaces in response to loading. The contour stress plots for σ_{yy} and σ_{yz} are shown in Fig. 3.46 and Fig. 3.47, respectively.

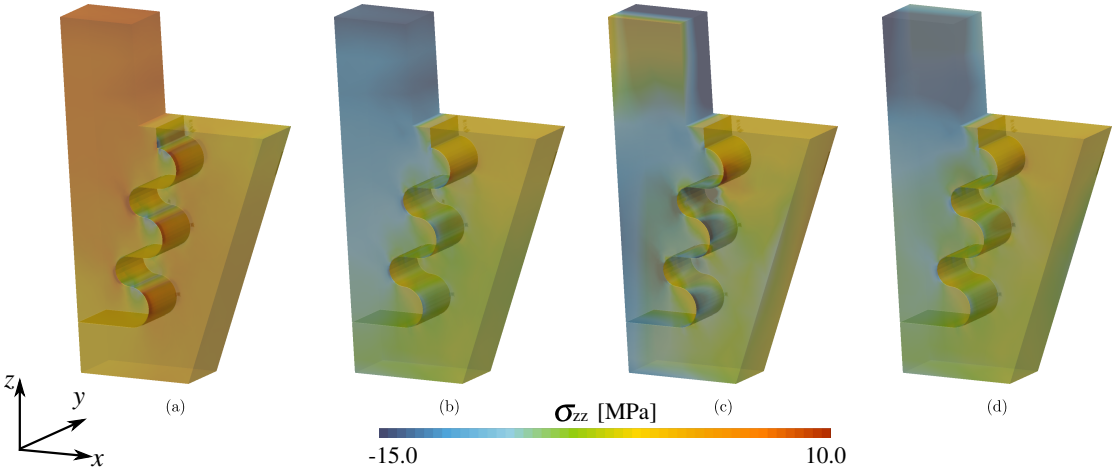


Figure 3.46: σ_{yy} contour stress plots at: (a) $t = 1$; (b) $t = 2$; (b) $t = 3$; (d) $t = 4$ seconds.

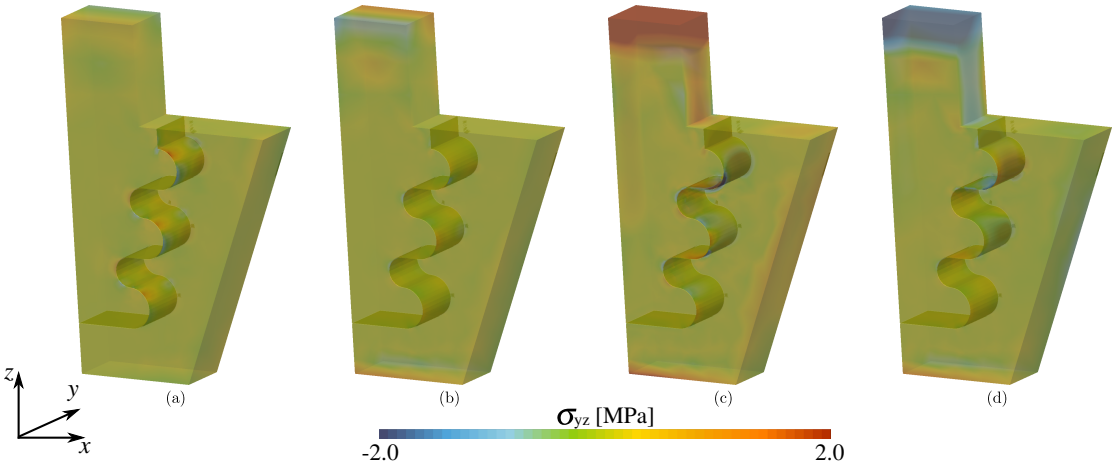


Figure 3.47: σ_{yz} contour stress plots at: (a) $t = 1$; (b) $t = 2$; (b) $t = 3$; (d) $t = 4$ seconds.

3.8 Summary

We presented the mortar finite element discretization framework for the mesh-tying of non-overlapping domains and the contact problems. The numerical algorithms of projection, clipping and triangulation essential for accurate integration of the mortar integrals are discussed. Particularly for the contact problems,

a fully linearized discrete monolithic augmented Lagrangian scheme was implemented within the mortar framework. Apart from the discretization, other numerical aspects of the contact problem like the detection, parallel resolution are also presented. Optimized variants of the "quick search" algorithm have been adapted for the KD-tree construction, which enables to achieve an algorithmic complexity of $\mathcal{O}(n \log(n))$ [Section 3.5.1]. The efficiency of the KD-tree scheme was demonstrated on contact patch test set-up, with good gains in terms of overall search time as a function of the system size. The aspects of parallel resolution were discussed and tested on the contact patch test set-up and the rough surface problems (with$\approx$ 1.5 million DoFs), with satisfactory gains in simulation times [Section 3.5.2]. Emphasis was laid on algorithmic details of resolving contact inequalities with the aid of case studies presented in Section 3.6. The importance of iteratively constructing the contact interface discretization for both frictionless and frictional cases is demonstrated. The robustness of the implementation is demonstrated in the form of numerical examples spanning various aspects of interface problems with and without friction in 2D and 3D [Section 3.7].

Chapter 4

Extended mortar methods for tying

Résumé: Dans ce chapitre, nous présentons une nouvelle méthode numérique que nous appellerons MorteX. Cettee méthode est developée pour le couplage de maillages entre des domaines qui se chevauchent. Le cadre MorteX nécessite des composants numériques supplémentaires tels que la méthode des éléments finis étendus (X-FEM) afin de prendre en compte la présence des surfaces immergées dans le volume. Cette méthode nécessite également quelques modifications des schémas classiques de la méthode Mortar introduits dans le chapitre 3, afin de pouvoir appliquer des contraintes sur des surfaces immergées.

In this chapter we present the numerical framework MorteX that is specifically developed to address the embedded class of interface problems. The MorteX framework, requires additional numerical ingredients such as the extended finite element methods (X-FEM) to account for the presence of embedded surfaces. This also requires few adaptations of the classical mortar schemes introduced in Chapter 3, to be able to enforce constraints along embedded surfaces.

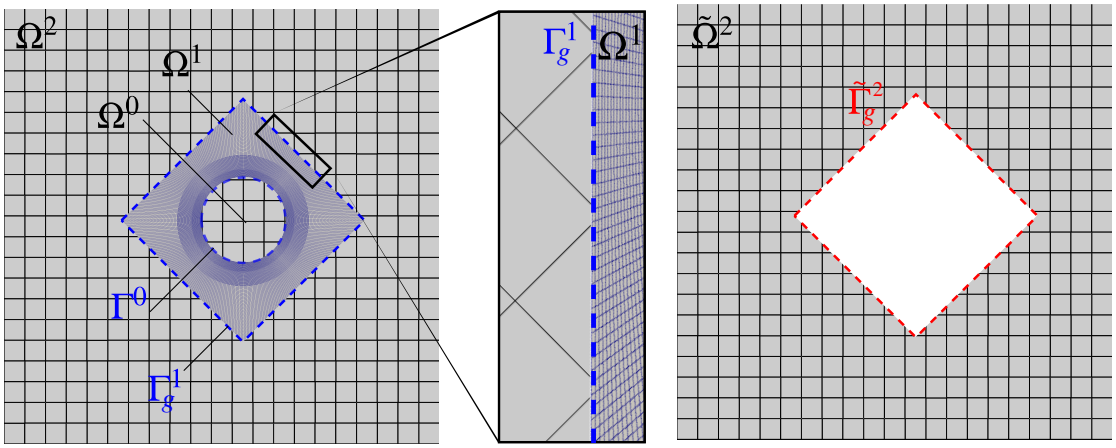
The MorteX framework is presented for the problem setting of tying between overlapping domains (see Section 2.2.3). Various aspects of the framework and involved complexities are elaborated using the same problem setting.

4.1 MorteX framework

The evaluation of the internal virtual work restricted to the effective volume of the host solid $\tilde{\Omega}^2$ is accomplished with the X-FEM method. The mortar method is extended to enforce the displacement equality constraint over the interface between the overlapping domains, i.e. between the boundary of the embedded domain (patch) Γ_g^1 and the corresponding virtual surface $\tilde{\Gamma}_g^2$ of the host solid.

The main features of the proposed method are illustrated on an example shown in Fig. 4.1. It represents the discretized finite-element setting for the overlapping domains. A discretized square patch with a circular hole $\Omega^0 \cup \Omega^1$ with surfaces $\Gamma^0 = \partial\Omega^0$ and $\Gamma_g^1 = \partial\left\{\Omega^1 \cup \bar{\Omega}^0\right\}$ is embedded into a host mesh Ω^2. As before the bar notation is used to denote the open domain united with its closure, here $\bar{\Omega}^0 = \Omega^0 \cup \partial\Omega^0$. Note that the necessity to consider explicitly the hole as an extra domain comes from the particularities of the problem and was introduced for the sake of avoidance of misinterpretation: the physics inside the contour Γ_g^1 is fully

determined by the patch with a circular hole. The intersection of the patch's boundary Γ_g^1 and the host domain represents the virtual surface $\tilde{\Gamma}_g^2 = \Gamma_g^1 \cap \Omega^2$. The X-FEM is used to account for the virtual work only in the effective domain volume $\tilde{\Omega}^2 = \Omega^2 \setminus \{\bar{\Omega}^0 \cup \bar{\Omega}^1\}$. The mortar method is is brought into play to tie together the two domains Ω^1 and $\tilde{\Omega}^2$ along the interface made of Γ_g^1 and $\tilde{\Gamma}_g^2$. Note that in the presented example, the tying boundary is fully embedded.



(a) (b) (c)

Figure 4.1: (a) Two overlapping meshes: the host Ω^2 and the patch Ω^1 with a circular hole Ω^0 are tied along interface Γ_g^1; (b) zoom on the interface between the host and patch meshes; (c) effective volume of the host mesh $\tilde{\Omega}^2 = \Omega^2 \setminus \{\bar{\Omega}^0 \cup \bar{\Omega}^1\}$.

4.1.1 Extended finite element method

The virtual surface $\tilde{\Gamma}_g^2$ of the host domain is treated as an internal discontinuity. This is modeled within the X-FEM framework, thereby nullifying the presence of the overlap region $\left(\bar{\Omega}^1 \cup \bar{\Omega}^0\right) \cap \Omega^2$ in the domain Ω^2 [see Fig. 4.1(c)]. The X-FEM relies on enhancement of the FEM shape functions used to interpolate the displacement fields. Here the enrichment functions describing the field behavior are incorporated locally into the finite element approximation. This feature allows the resulting displacement to capture discontinuities. The subdivision of the host mesh is defined by indicator function $\phi(\underline{\boldsymbol{X}}) : \mathbb{R}^{\dim} \to \{0, 1\}$ (where $\underline{\boldsymbol{X}}$ is the spatial position vector in the reference configuration in domain Ω^2) [Sethian, 1999]. The indicator function is non-zero only in the non-overlapping part of domain Ω^2:

$$\phi(\underline{\boldsymbol{X}}) = \begin{cases} 1, & \text{if } \underline{\boldsymbol{X}} \in \tilde{\Omega}^2; \\ 0, & \text{elsewhere.} \end{cases}$$

The discontinuity surface $\tilde{\Gamma}_g^2$ can be seen as a level-set defined as follows:

$$\tilde{\Gamma}_g^2 = \left\{ \underline{\boldsymbol{X}} \in \Omega^2 : \ \nabla\phi(\underline{\boldsymbol{X}}) \neq 0 \right\}$$

As a result, the indicator function $\phi(\underline{\boldsymbol{X}})$ partitions the elements of the host domain Ω^2 into three distinct categories [Fig. 4.2(a)], namely standard elements, blending elements and discarded elements.

In practice, the enrichment of shape functions in case of void/inclusion problem can be simply replaced by a selective integration scheme [Sukumar et al., 2001]. For the standard elements, there is no change in volume of integration and the discarded elements are simply excluded from the volume integration procedure. In order to obtain the effective volume of integration for each blending element, we perform the clipping of the blending elements by the discretized surface $\tilde{\Gamma}_g^2$ [Fig. 4.2(b)]. The clipping of a single element could result in one or several various polygons[1] both convex, and non-convex, which represent the effective volumes of integration.

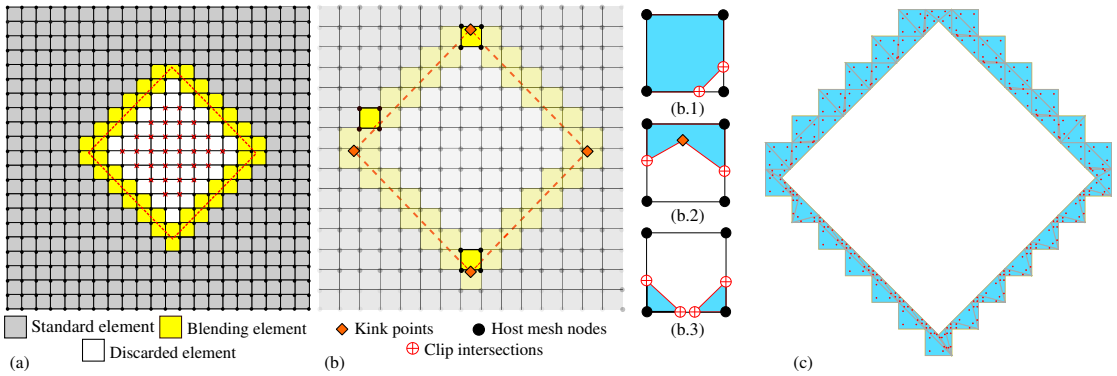


Figure 4.2: (a) Element classification in X-FEM framework; (b) clipping of blending elements by $\tilde{\Gamma}_g^2$, the volume colored in blue in (b.1-3) is the effective volume of integration ($\tilde{\Omega}^e$): (b.1) a convex polygon, (b.2) a non-convex polygon, (b.3) disjoint polygons; (c) selective integration is carried out over re-triangulated blending elements with reinitialized Gauss integration points (shown in red).

To selectively integrate the internal virtual work in the effective volume only, the resulting polygons are virtually remeshed into standard convex elements (for example, triangles). Note that this remeshing is merely performed to use a Gauss quadrature for integration [Fig. 4.2(c)], and does not imply the creation of additional degrees of freedom, and as such does not change the topological connectivity of nodes. The displacement field is evaluated using the standard shape functions and the original DoFs; only the integration is changed. To carry out this remeshing, we applied the ear clipping triangulation algorithm [Mei et al., 2013] to the polygons. The DoFs associated with the elements outside the integration domain $\tilde{\Omega}^2$ [marked with red crosses in Fig. 4.2(a)] are removed from the global system of equations.

4.1.2 MorteX discretization

Within the mortar discretization framework, the tied domains are classified into mortar and non-mortar sides. The superscript "1" refers to the mortar side of the interface and "2" to the non-mortar side; the former stores the Lagrange multipliers (dual DoFs) in addition to displacement degrees of freedom (primal DoFs). If the host is selected as a mortar side, the context of the problem becomes

[1]Hereinafter, we assume that all elements use first order interpolation, therefore all edges of elements are straight. It enables us to assume that an intersection or difference of elements can be always represented as one or several polygons.

similar to the one considered in [Moës et al., 2006, Béchet et al., 2009, Hautefeuille et al., 2012], where it was shown that strong restrictions apply on the choice of Lagrange multiplier spaces in order to fulfill the inf-sup condition. The algorithm for construction of such spaces is not straightforward. Therefore, to avoid these difficulties, we select the patch side as the mortar surface, which provides us with a more flexible setting. This choice was already reflected in the fact that the host boundary was chosen as the integration side for tying conditions (2.72). However, under specific problem settings, the choice of employing standard interpolation functions for the Lagrange multipliers on the embedded interface still leads to spurious oscillations of interfacial tractions. Remedies for this problem will be discussed in Section 4.4.

Displacements on the mortar side Γ_g^1 are given by classical one-dimensional shape functions with the interpolation order equal to that of the underlying mesh:

$$\underline{\boldsymbol{u}}^1(\xi^1) = N_m^1(\xi^1)\underline{\boldsymbol{u}}^m, \quad m \in [1, \mathrm{M}], \tag{4.1}$$

where M is the number of nodes per mortar element's edge and $\xi^1 \in [-1; 1]$ is the parametric coordinate of the mortar side. The displacements along the virtual surface $\tilde{\Gamma}_g^2$ running through the host mesh elements are characterized by the two dimensional host mesh interpolations, and can be expressed as follows:

$$\underline{\boldsymbol{u}}^2(\zeta) = N_i^2\left(\mu^2(\zeta), \eta^2(\zeta)\right)\underline{\boldsymbol{u}}^i, \quad i \in [1, \mathrm{N}], \tag{4.2}$$

where ζ is the one-dimensional parametric coordinate of integration segment of non-mortar side, $\mu^2, \eta^2 \in [-1; 1]$ are the classical two-dimensional parametric coordinates of the host element, and N is the number of nodes of this element. The use of two-dimensional interpolation on the non-mortar side marks the difference between the classical mortar and the presented MorteX frameworks. The Lagrange multipliers (defined on the mortar side) are interpolated using shape functions Φ:

$$\underline{\boldsymbol{\lambda}}(\xi^1) = \Phi_l(\xi^1)\,\underline{\boldsymbol{\lambda}}^l, \quad l \in [1, \mathrm{L}], \tag{4.3}$$

where L can be less than or equal to M. It enables to select shape functions for dual variables independently of the primal shape functions.

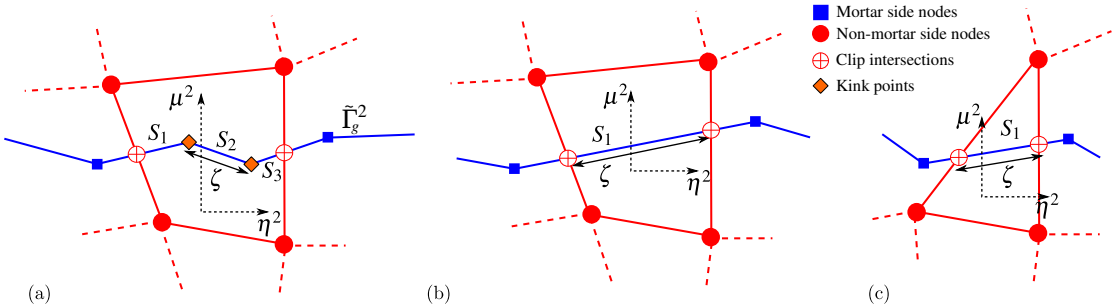


Figure 4.3: Illustration of a single host element intersected by the virtual surface $\tilde{\Gamma}_g^2$: (a) quadrilateral host element intersected by several mortar side segments; (b) quadrilateral host element intersected by a single mortar segment; (c) triangular host element intersected by a single mortar segment.

Few remarks could be made here. In the absence of a real non-mortar side, we redefine the notion of segment: it is a straight line whose vertices can either

be a clip intersection[2] or a kink point, the latter being a node of the mortar side lying inside the host element. This 1D non-mortar segment also represents the mortar domain S^{el}. Since a host element can be intersected by several patch segments S_i [Fig. 4.3(a)], the functions $\mu^2(\zeta)$, $\eta^2(\zeta)$ can be piece-wise smooth, which implies that the underlying displacement (and coordinate) can also be piece-wise smooth. Second remark: if the host elements are quadrilateral, then each segment interpolation is given by $p_1 \times (p_2+1)$, where p_1 is the mortar interpolation order, and p_2 is the non-mortar interpolation order; the later is augmented by one since the virtual interface passes inside the element, where the interpolation order in quadrilateral 2D elements is one order higher than along the edges. For a triangular host mesh, the interpolation order is simply the product of interpolation orders of the host and patch meshes $p_1 \times p_2$. To give an example, let us assume that the patch mesh is linear and the host mesh is linear quadrilateral; then let us imagine that a host element is cut into two parts by a (straight) patch segment. Then the displacement in the host element along this straight segment will be second order polynomial function of the parameter ζ [see Fig. 4.3(b)]. However, the displacement along such a cut would remain linear for triangular linear host elements [see Fig. 4.3(c)].

MorteX interface element

A mortar element is formed with a single mortar segment (on the patch side) and a single non-mortar element (on the host side). Each tying element consists of (M+N) nodes, M from the mortar segment, and N from non-mortar element, and stores (L × dim) Lagrange multipliers, where (dim) is the spatial dimension. The choice of L is guided by the inf-sup condition requirement of the discrete Lagrange multiplier spaces (usually L ≤ M).

Substituting the interpolations (4.1), (4.2), (4.3) into the weak form (2.72) and extracting only the terms related to the mesh tying of a single mortar element, we obtain

$$\delta\tilde{W}_g^{\text{el}} = \left(\underline{\boldsymbol{\lambda}}^l \cdot \delta\underline{\boldsymbol{u}}^m + \delta\underline{\boldsymbol{\lambda}}^l \cdot \underline{\boldsymbol{u}}^m\right) D_{lm} - \left(\underline{\boldsymbol{\lambda}}^l \cdot \delta\underline{\boldsymbol{u}}^i + \delta\underline{\boldsymbol{\lambda}}^l \cdot \underline{\boldsymbol{u}}^i\right) \tilde{M}_{li}, \qquad (4.4)$$

$$l \in [1, \text{L}],\ i \in [1, \text{N}],\ m \in [1, \text{M}]$$

where D_{lm} and $\tilde{M}_{li}$ are the mortar integrals evaluated over the mortar domain $S^{\text{el}} \subset \left(\Gamma_g^1 \cap \tilde{\Omega}_i^e\right)$, where $\tilde{\Omega}_i^e$ is the current host element forming the mortar element. The integral D_{lm} on the mortar side is the 2D variant of integral in Eq. 3.5 .

$$D_{lm} = \int_{S^{\text{el}}} \Phi_l^1(\xi^1) N_m^1(\xi^1)\, d\Gamma, \qquad (4.5)$$

$$\tilde{M}_{li} = \int_{S^{\text{el}}} \Phi_l^1(\xi^1) N_i^2\left(\mu^2(\zeta^2), \eta^2(\zeta^2)\right)\, d\Gamma. \qquad (4.6)$$

The nodal blocks of the mortar matrices denoted as $\mathbf{D}$ (L × M) and $\mathbf{M}$ (L × N) can be expressed as:

$$\mathbf{D}(l, m) = D_{lm}\underline{\underline{\boldsymbol{I}}}, \qquad (4.7)$$

[2]The clip intersection points are located on the edges of blending elements intersected by the virtual surface $\tilde{\Gamma}_g^2$.

$$\tilde{\mathbf{M}}(l,i) = \tilde{M}_{li}\underline{\underline{\boldsymbol{I}}}, \tag{4.8}$$

where $\underline{\underline{\boldsymbol{I}}}$ is the identity tensor of the spatial dimension of the problem. Using matrix notations, Eq. (4.4) reads

$$\delta\tilde{W}_g^{\mathrm{el}} = \begin{bmatrix} \mathbf{D}^\intercal \cdot \mathbf{L} \\ -\mathbf{M}^\intercal \cdot \mathbf{L} \\ \mathbf{D}\cdot\mathbf{U}^1 - \mathbf{M}\cdot\mathbf{U}^2 \end{bmatrix}^\intercal \cdot \begin{bmatrix} \delta\mathbf{U}^1 \\ \delta\mathbf{U}^2 \\ \delta\mathbf{L} \end{bmatrix}, \tag{4.9}$$

where arrays $\mathbf{U}^1, \mathbf{U}^2, \mathbf{L}$ store current values of associated nodal primal (on mortar and non-mortar sides) and dual (mortar) DoFs:

$$\mathbf{U}^1 = \left[\underline{\boldsymbol{u}}^1, \ldots, \underline{\boldsymbol{u}}^N\right]^\intercal, \quad \mathbf{U}^2 = \left[\underline{\boldsymbol{u}}^1, \ldots, \underline{\boldsymbol{u}}^M\right]^\intercal, \quad \mathbf{L} = \left[\underline{\boldsymbol{\lambda}}^1, \ldots, \underline{\boldsymbol{\lambda}}^L\right]^\intercal$$

whereas their variations are denoted $\delta\mathbf{U}^1, \delta\mathbf{U}^2, \delta\mathbf{L}$. The tangent operator for the mortar interface element is obtained by taking the derivatives of (4.9) with respect to its DoFs:

$$\mathbf{K} = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{D}^\intercal \\ \mathbf{0} & \mathbf{0} & -\tilde{\mathbf{M}}^\intercal \\ \mathbf{D} & -\tilde{\mathbf{M}} & 0 \end{bmatrix}. \tag{4.10}$$

Evaluation of integrals

The mortar integrals are evaluated over the mortar domain S^{el} [see Figs. 4.3, 4.4]. In comparison to the evaluation of integrals in classical mortar framework, the evaluation within MorteX framework has the following differences:

- the non-mortar side integral $\tilde{M}_{li}$ (4.6) involves second order interpolation functions $N_i^2\left(\mu^2(\zeta^2), \eta^2(\zeta^2)\right)$ compared to the first order in (3.6).
- the operator $\tilde{\mathbf{M}}$ (4.8) in the MorteX framework is always rectangular, compared to the square operator (3.8).
- the evaluation of the integrals (4.5) (4.6) in MorteX framework does not require a projection step (3.43) to determine the mortar domain S^{el}. It is readily available as a complementary output of the clipping (selective integration) step. This also circumvents the need for projecting the Gauss points as done in (3.45).
- the limits of mortar domain S^{el} are clip intersections or kink points (mortar nodes) [see Fig. 4.4], while in classical mortar schemes they are projections of non-mortar nodes or mortar nodes [see Fig. 3.2].

The mortar domain is parametrized by $\chi \in [-1, 1]$ [Fig. 4.4], which needs to be linked with the parametrization of the mortar side, which is given by:

$$\xi^1(\chi) = \frac{1}{2}(1-\chi)\xi_a^1 + \frac{1}{2}(1+\chi)\xi_b^1, \tag{4.11}$$

where ξ_a^1 and ξ_b^1 define the limits of the integration on the mortar side as shown in Fig. 4.4. To evaluate the integrals using the Gauss quadrature, we need to find the location of Gauss points χ_G in terms of mortar (ξ_G^1) and non-mortar (μ_G^2, ν_G^2)

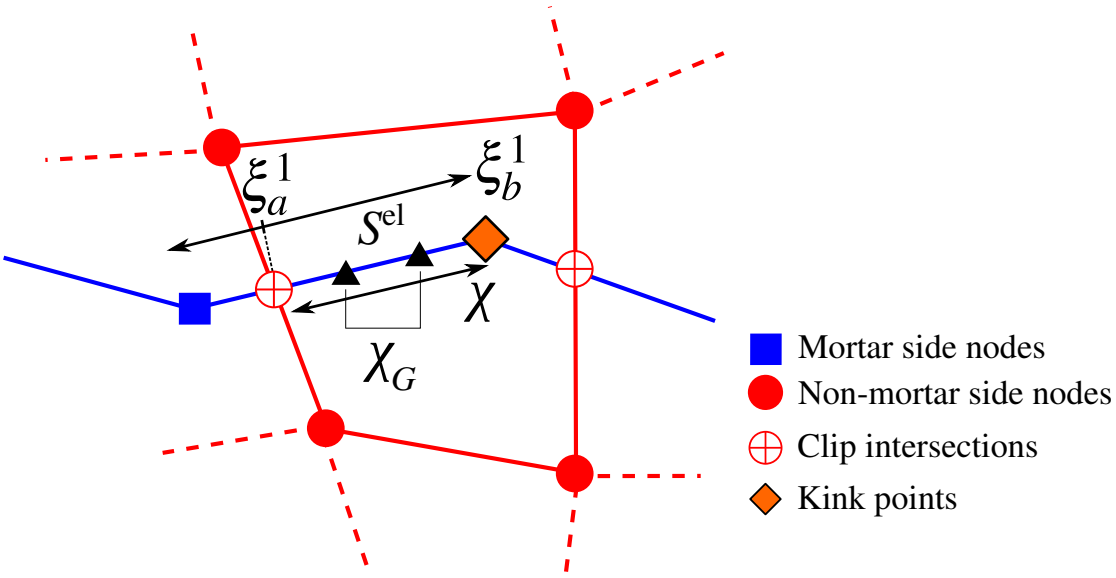


Figure 4.4: Example of mortar domain S^{el} with a clip intersection ξ^1_a and kink point ξ^2_b as its limits (Gauss points are shown as triangles).

parametrization. While the former is straightforward using (4.11), the latter can be done by solving the following equation:

$$N^1_m(\xi^1_G)\underline{\boldsymbol{X}}^1_m = N^2_i(\mu_G, \eta_G)\underline{\boldsymbol{X}}^2_i, \tag{4.12}$$

where the physical location of the Gauss point is given by $\underline{\boldsymbol{X}}_G = N^1_m(\xi^1_G)\underline{\boldsymbol{X}}^1_m$. With these notations, the mortar integrals can now be evaluated using the Gauss quadrature rule as:

$$D_{lm} = \int_{S^{\text{el}}} \Phi_l(\xi^1) N^1_m(\xi^1)\, d\Gamma = \sum_{G=1}^{N_G} w_G \Phi_l(\xi^1_G) N^1_m(\xi^1_G) J_{\text{seg}}(\xi^1_G), \tag{4.13}$$

$$\tilde{M}_{li} = \int_{S^{\text{el}}} \Phi_l(\xi^1) N^2_i(\mu^2, \eta^2)\, d\Gamma = \sum_{G=1}^{N_G} w_G \Phi_l(\xi^1_G) N^2_i\left(\mu^2_G, \eta^2_G\right) J_{\text{seg}}(\xi^1_G), \tag{4.14}$$

where as previously $l \in [1, L]$, $m \in [1, M]$, $n \in [1, N]$ and N_G is the number of Gauss integration points, w_G are the Gauss weights, J_{seg} is the Jacobian of the mapping from the parent space ξ^1 to the real space including the adjustment of the integral limits:

$$J_{\text{seg}}(\xi^1) = \left| \frac{\partial N_m}{\partial \xi^1} \frac{\partial \xi^1}{\partial \chi} \underline{\boldsymbol{X}}_m \right|. \tag{4.15}$$

In the classical mortar framework, the numerical procedure to evaluate the integrals remained same for both the classes of tying and contact problems. Within MorteX framework, the above numerical procedure cannot be used as is for the contact problems within the MorteX framework. Details of the procedure are presented in Chapter 5.

4.2 Intra-element interpolation of displacements in the host mesh

For the overlapping domains the coupling is made between the patch mesh boundary and virtual surface running through the host mesh elements. This is reflected in the mortar matrix $\mathbf{M}$ (4.14) that contains the integral of a product of volumetric (in the host mesh) and surface (patch mesh) shape functions. To demonstrate the effect of the interpolation choice, we use the set-up shown in Fig. 4.5(a). The following dimensions are used: $h^1 = 1$ mm, $h^2 = 1.25$ mm, $h^* = 0.25$ mm and $l = 1.5$ mm. A uniform pressure of $\sigma_0 = 1$ MPa is applied on the top surface of Ω^1. The meshes Ω^1 and Ω^2 are tied along the interface Γ_g^1.

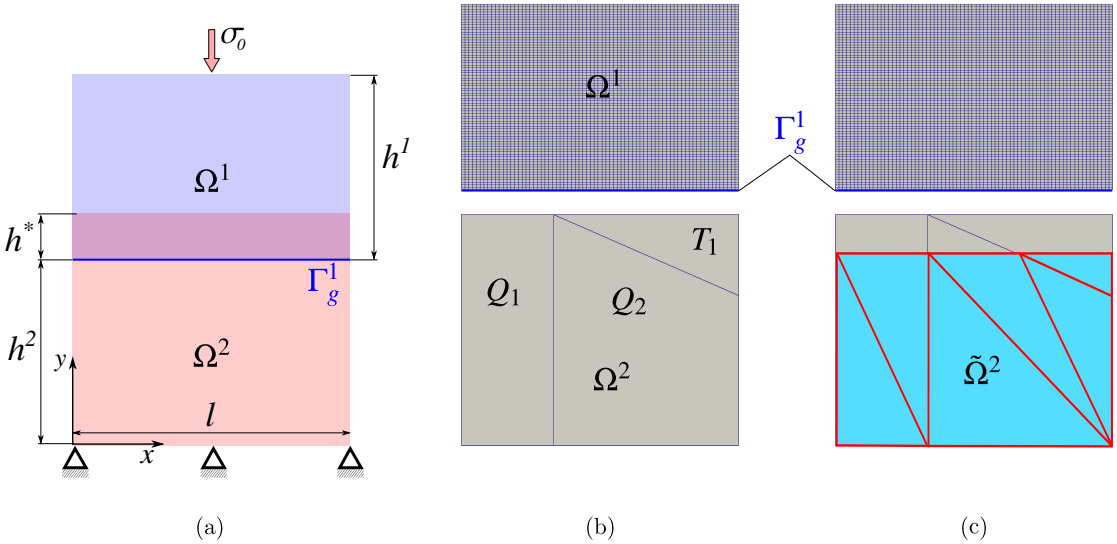


Figure 4.5: Illustration of the effect of underlying mesh interpolations: (a) problem setting: tying of two rectangular overlapping domains; (b) discretized patch (upper solid) and host (lower solid) domains; (c) triangulated effective volume of the host domain $\tilde{\Omega}^2$; note that in (b,c) the two meshes are shown separated only for convenience.

The domain Ω^2 is discretized with a triangular (T1) and two quadrilateral (Q1, Q2) elements, all elements use first order interpolation, and the patch domain is discretized into a rectangular elements [Fig. 4.5(b)]. The two domains are made of the same linearly elastic material ($E = 1$ GPa, $\nu = 0$); the reference solution for the selected boundary conditions is a uniform stress field ($\sigma_{xx} = \sigma_{xy} = 0$ and $\sigma_{yy} = \sigma_0$) whether it be under plane strain or plane stress formulation. Of course, the displacement along the tying line is uniform. However, as shown in Fig. 4.6 the selected discretization does not allow to obtain the reference solution. The solid line in Fig. 4.6 shows the resulting vertical displacement along the tying line Γ_g^1; it consists of a combination of linear and non-linear portions. We hypothesize that the inability to reproduce the reference solution is somehow related to the interpolation order of displacement in host elements. As seen from the figure, the order of the solution is matching the maximum available interpolation order of displacements $p_{\max}$. This maximum interpolation order is one ($p_{\max} = 1$) along any straight line inside a linear triangle (T1). However, this order raises to two ($p_{\max} = 2$) along straight lines inside quadrilateral elements, as long as both

parent coordinates μ and η change along these lines (Q2). This is not the case in Q1 ($p_{\max} = 1$), where one of parent coordinates remains constant along the virtual interface, i.e. $\mu^2(\zeta) = \text{const}$ or $\eta^2(\zeta) = \text{const}$.

This observation motivates us to test triangulation of blending element to limit the maximum interpolation order in host element to one ($p_{\max} = 1$). This operation does not change the number of DoFs and is easy to handle in practice. This procedure enables us to obtain the reference solution in the considered case as demonstrated in Fig. 4.6. The general applicability of this method will be tested in the following sections.

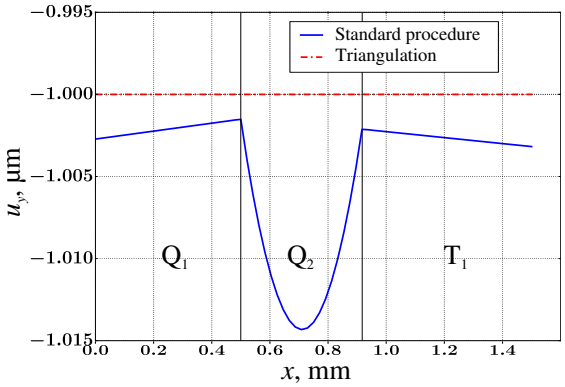


Figure 4.6: Displacement u_y profile along nodes of $\Gamma^1_{g,h}$ for the set-up shown in Fig. 4.5: direct tying using the MorteX method results in non-linear displacement field (solid line); triangulation of blending elements results in a perfect match (dash-dotted line) with the reference solution $u_y = -1$ μm.

4.3 Coarse Grained Interpolation of Lagrange multipliers

The stability of the proposed mixed formulation is guided by the requirement to satisfy the inf-sup condition [Babuška, 1973], which is not a trivial task. For example, the use of Lagrange multipliers with standard interpolation leads to non-physical oscillations along the interface when used to enforce Dirichlet boundary conditions [Barbosa and Hughes, 1991]. Having a much stiffer patch than the host material represents a case approximately similar to the imposition of Dirichlet boundary conditions, therefore in most presented examples stiffer patch will be used. The technique presented in [Moës et al., 2006, Béchet et al., 2009] for the X-FEM framework, involves coarsening of the Lagrange multipliers to avoid spurious oscillations. This is achieved by algorithmically selecting nodes referred to as "winner nodes" along the Dirichlet boundary. The algorithm favors nodes close to the boundary, or nodes from which many edges intersecting the boundary emanate. The Lagrange multiplier space is built using the winner nodes. The size of the multiplier space is the size of the winner nodes set. Inspired from this technique, we suggest to use coarse grained interpolation (CGI) for Lagrange multipliers (dual DoFs) to avoid spurious oscillations. In Fig. 4.7, for example, the number of mortar nodes (each of which carries Lagrange multipliers in the standard approach) per host element considerably outnumbers the associated DoFs to

render the associated constraints independent of physical deformations the host elements allow [Sanders et al., 2012]. Therefore the system shown in this figure is overconstrained and requires an appropriate stabilization. Coarse graining of Lagrange multiplier interpolation functions enables to reduce the number of constraints and thus improves the problem stability. In this approach, not every mortar node is equipped with a Lagrange multiplier. Therefore, the interpolation functions become non-local, i.e. they span more than one patch segment. For this purpose, we choose a 1D parametric space $\xi^{CG} \in [-1, 1]$, spanning multiple mortar-side segments. Such parametrization can be chosen such that length L^i of the corresponding super-segment in the physical space is comparable to the size of host elements. As shown in Fig. 4.7(a), the mortar-surface is segmented into three super-segments of lengths L^1, L^2 and L^3. The end nodes of these segments are termed the "master" nodes (they carry the dual DoFs $\underline{\boldsymbol{\lambda}}$), other mortar-nodes are termed "slave" nodes. We introduce the local coarse-graining parameter κ that determines the number of segments contained in a super-segment, and thus $(\kappa - 1)$ determines the number of slave nodes per super-segment. In Fig. 4.7(a), the coarse-graining parameter takes the values $\kappa = 4, 9, 5$, for the super-segments of lengths L^1, L^2 and L^3, respectively.

In theory, the coarse graining is achieved through defining dual DoFs only on master nodes. In practice, this can also be done by keeping the Lagrange multipliers at all mortar nodes and using a multi-point constraint (MPC) to enforce a linear interpolation between the master nodes. Hence, for a given slave node, the dual DoFs are given by:

$$\underline{\boldsymbol{\lambda}}_{\text{slave}}(\xi^{CG}) = \Phi_l^{CG}(\xi^{CG})\underline{\boldsymbol{\lambda}}^l_{\text{master}}, \quad l = 1, 2. \tag{4.16}$$

The parametrization of the super-segment can be chosen such that $\forall i$: $(\xi^{CG}_{i+1} - \xi^{CG}_i)/L_{i+1,i} = \text{const}$, where the numerator represents the spacing between two mortar (patch) nodes in the coarse-grained parametric space and $L_{i+1,i}$ corresponds to the physical length of the corresponding segment. An average ratio of number of mortar segments per number of blending elements, which is termed "mesh contrast" m_c, can be used to guide the selection of the coarse-graining parameter κ: for example, $m_c = 6$ (18 mortar segments per 3 host elements) in the example shown in Fig. 4.7. The coarse-graining parameter κ, for an open mortar surface, can take values in the range $\kappa \in [1, N_m]$, where N_m is the number mortar segments; for a closed mortar surface the upper limit is one less.

The optimal choice of coarse-graining parameter κ is studied on particular problem settings in Sections 4.4 and 4.5 for open and closed mortar surfaces. The limit case $\kappa = 1$ corresponds to the standard Lagrange interpolation (SLI). For the case of approximately regular discretization on both sides, a global coarse-graining parameter can be chosen, and its value is set to be approximately equal to the mesh contrast parameter $\kappa \approx m_c$ as shown in Fig. 4.7(b). In case of non-regular mesh discretizations on mortar or/and on host sides, the coarse-graining parameter should be selected element-by-element according to the local mesh contrast as shown in Fig. 4.7(a). However, in all the examples considered below, we use regular discretizations, and thus the global coarse-graining parameter will be used.

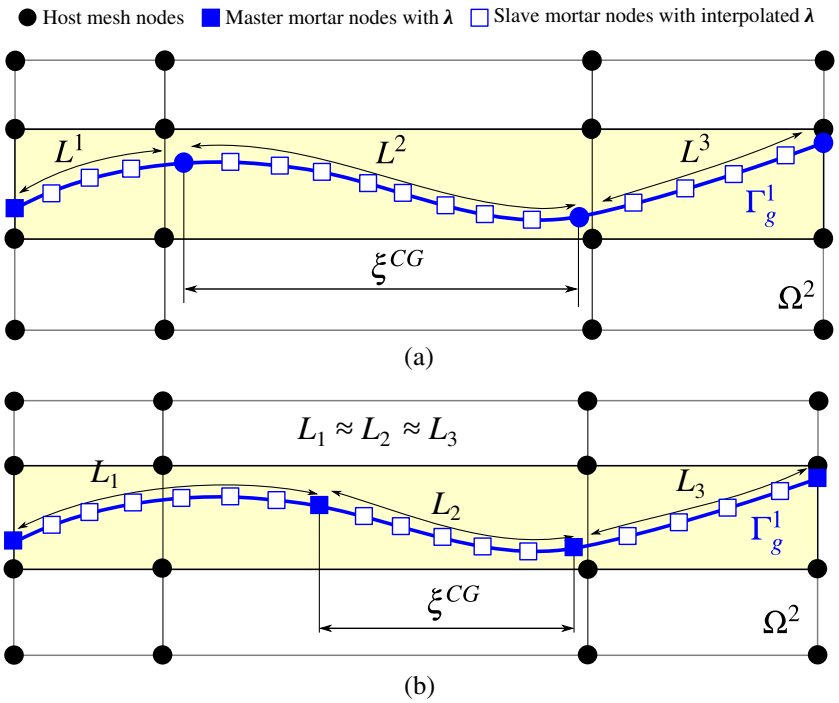


Figure 4.7: An example illustrating the embedded surface Γ_g^1 cutting through the blending elements (shaded in yellow) of the coarser host mesh. The coarse graining of Lagrange multipliers can be implemented with respect to the local (a) or global; (b) contrast in mesh densities.

4.4 Patch tests

In this section, the algorithms introduced in the previous sections, are tested on simple problems of tied overlapped domains of different discretizations and different material contrasts subject to bending or tensile/compressive boundary conditions: the patch mesh is Ω^1 and the host mesh is Ω^2. Linear elastic material properties are used for both the patch (E^1, ν^1) and the host (E^2, ν^2). The geometric set-up of the patch and host domains are illustrated in [Fig. 4.8(a)]. The following two extreme cases will be considered: **Case 1.** a finer and stiffer patch mesh is superposed onto the host mesh, and **Case 2.** a coarser and stiffer patch mesh is superposed onto the host mesh.

These two particular cases are chosen for the validation since they are prone to severe manifestations of the mesh locking [Sanders et al., 2012] as in the case of enforcing Dirichlet boundary conditions using Lagrange multipliers along embedded surfaces [Moës et al., 2006, Béchet et al., 2009, Hautefeuille et al., 2012]. Additionally, the host domain is meshed with "distorted" quadrilaterals which is classical in patch test studies to exacerbate potential anomalies. Moreover, as was shown by a simple example Section 4.2 the tying along distorted quadrilateral elements is prone to considerable errors if the mortar-type tying is used directly. The material contrast is introduced by choosing $E^1/E^2 = 1000$, and $E^1 = 1$ GPa, both domains have the same Poisson's ratio $\nu^1 = \nu^2 = 0.3$. The discretizations for the two cases are shown in Fig. 4.8(b, c). The mesh contrast $m_c \approx 11$ and $m_c \approx 0.1$ and the number of mortar segments $N_m = 191$ and $N_m = 35$ are used for Case 1 and 2, respectively. Note that all the stress fields σ_{xx}, σ_{yy} along the

tying interface, which are presented on the plots below, are found by the data extrapolated from Gauss points and averaged at mortar nodes.

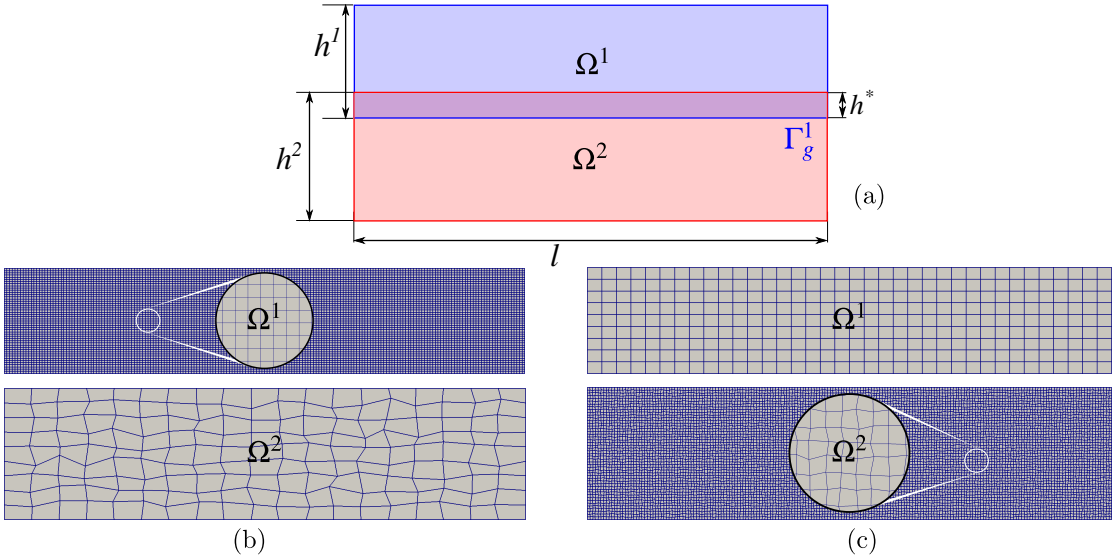


Figure 4.8: Validation tests set-up: (a) problem setting $h^1 = 1.0$ mm, $h^2 = 1.25$ mm, $h^* = 0.25$ mm, $l = 5.0$ mm, the elastic contrast between the patch and the host is given by $E^1/E^2 = 1000$; finite-element discretizations of the patch and host solids are shown in (b) for Case 1 (the patch mesh is finer than that of the host, $m_c \approx 11$, $N_m = 191$), and in (c) for Case 2 (the host mesh is finer than that of the patch, $m_c \approx 0.1$, $N_m = 35$).

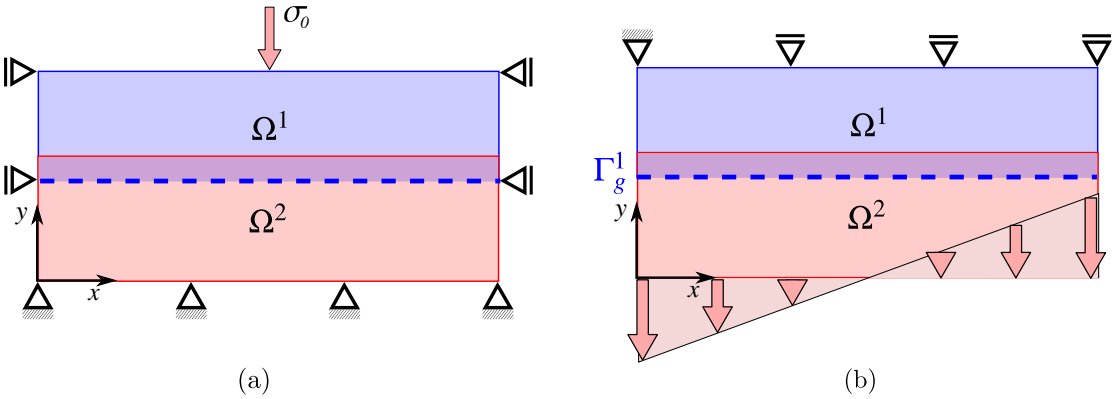


Figure 4.9: Validation tests boundary conditions: (a) bending patch test, (b) compression patch test.

4.4.1 Tension/compression patch test

A uniform pressure σ_0 is applied on the top surface, the bottom surface is fixed in all directions $\underline{\boldsymbol{u}} = 0$ [Fig. 4.9(a)]. This is a classical patch test in contact mechanics, which is used here to test the tying of different materials. This material contrast requires additional lateral conditions (lateral sides are fixed in normal direction $u_x = 0$) to avoid singularities at extremities of the interface. The reference solution for σ_{yy} is a uniform field $\sigma_{yy} = \sigma_0$. As expected, in case of stiffer and finer patch mesh (Case 1), spurious oscillations are observed. They have large amplitude that

reaches 300 % of the reference solution, moreover, they are not confined to the interface but propagate into the bulk [Figs. 4.10(a), 4.11(a)]. In case of stiffer and coarser patch mesh (Case 2), the spurious oscillations are of considerably lower amplitude (under 1 %), they are rather localized in the host mesh in close vicinity of the interface and do not extend in the patch mesh [Figs. 4.10(b), 4.11(b)].

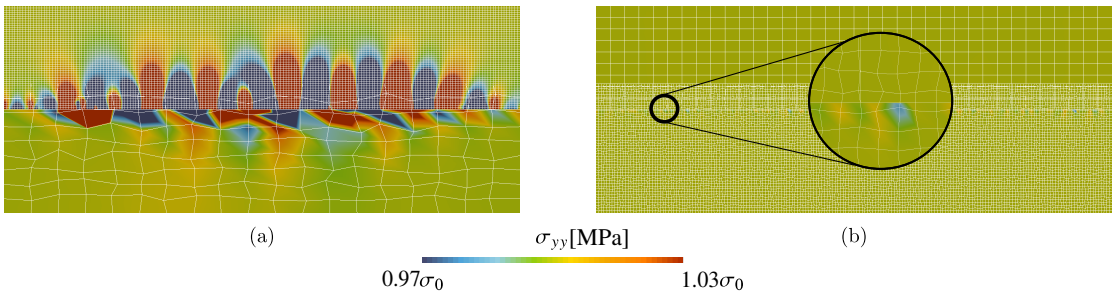


Figure 4.10: Compression patch test: contour plots of stress component σ_{yy} in (a) Case 1; (b) Case 2.

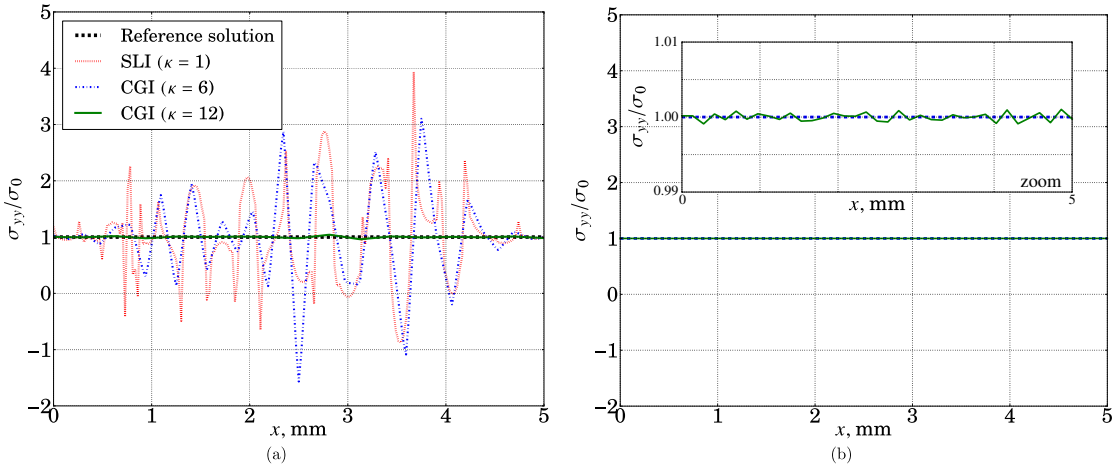


Figure 4.11: Compression patch test: distribution of σ_{yy} along the tying interface in (a) Case 1 for standard (SLI) and coarse-grained (CGI) Lagrange interpolation, and (b) in Case 2 for SLI.

In order to quantify the improvement achieved with the suggested coarse grained interpolation (CGI) and with triangulation technique, we introduce the L^2-norm of the error in the σ_{yy} stress component:

$$E_r(\sigma_{yy}) = \frac{\left\|\sigma_{yy}^{\text{ref}} - \sigma_{yy}\right\|_{L^2(\Gamma_g^1)}}{\left\|\sigma_{yy}^{\text{ref}}\right\|_{L^2(\Gamma_g^1)}}, \tag{4.17}$$

where the norm means $\|f(x) - g(x)\|_{L^2(\Gamma_g^1)} = \sqrt{\sum_i \left[f(x_i) - g(x_i)\right]^2}$, where $x_i \in [0, L]$ are the x-coordinate of mortar nodes, and L is the length of the surface Γ_g^1. In Fig. 4.12 we demonstrate the performance of the CGI technique. As seen from the figure, the error in stress greatly reduces compared to the standard interpolation (SLI), when coarse-graining parameter κ increases. However, the error saturates at $\approx 10^{-3}$ and the convergence to the reference solution is missing. On the contrary,

the triangulation technique [Fig. 4.13] enables to achieve a superior precision as shown in Figs. 4.12(a) and 4.14.

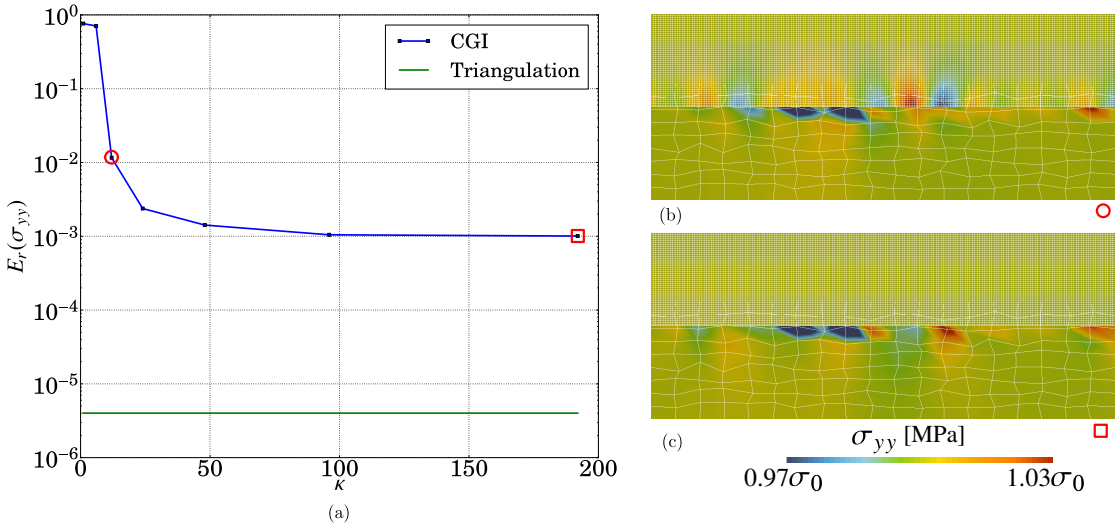


Figure 4.12: Compression patch test: (a) decay and saturation of the relative error $E_r(\sigma_{yy})$ for $\kappa = \{1, 6, 12, 24, 48, 96, 192\}$ for CGI in comparison with the error obtained with triangulation of blending elements; (b,c) contour plots of stress component σ_{yy} for (b) $\kappa = 12$ and (c) $\kappa = N_m = 192$.

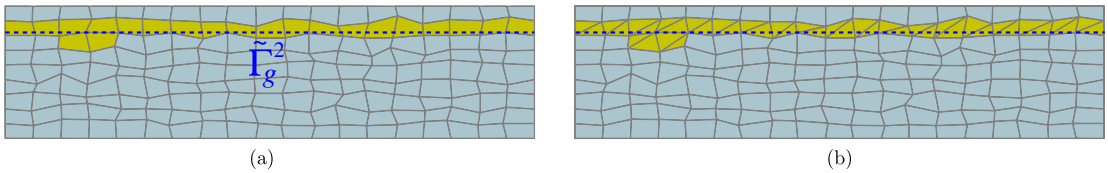


Figure 4.13: (a) Blending elements; (b) Triangulated blending elements.

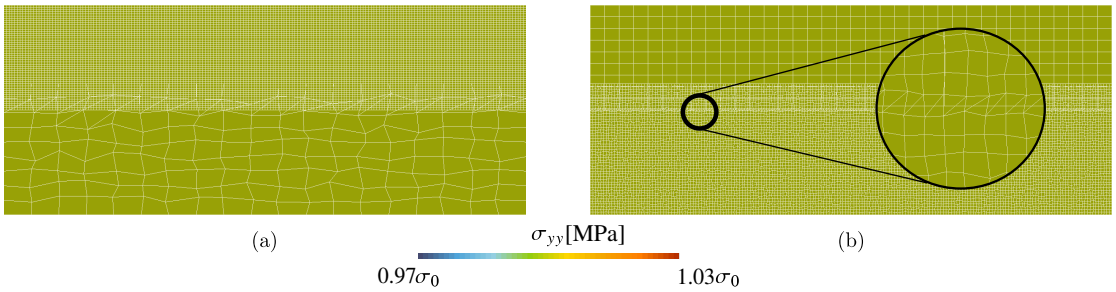


Figure 4.14: Patch test stress σ_{yy} with Triangulation of blending elements: (a) Case 1; (b) Case 2.

4.4.2 Bending patch test

A linear distribution of pressure

$$\sigma_{yy} = 2\sigma_0(x/l - 1/2), \tag{4.18}$$

with $\sigma_0 = 1$ MPa is applied on the bottom surface, while keeping the top surface fixed in vertical direction, only the corner point is fixed in horizontal direction, the lateral sides remain free $\sigma_{xy} = \sigma_{xx} = 0$ [Fig. 4.9 (b)]. The same linear distribution of the vertical stress component (4.18) through the two solids should take place. This case study (Case 1, 2) was inspired from the work [Sanders et al., 2012], where the authors also used the combination of the mortar method and the selective integration. It was shown that Case 1, in particular, results in high-amplitude spurious oscillations in the interface contrary to Case 2 that has a smoother stress profile along the interface. Under the standard Lagrangian interpolation set-up we could reproduce similar results, see Fig. 4.15, 4.16. These oscillations could be removed with Nitsche method provided some adjustment of the stabilization penalty parameters on each side of the interface [Sanders et al., 2012]. We demonstrate below that using the coarse-grained interpolation for Lagrange multipliers, as suggested in Section 4.3, also permits avoiding these oscillations in the MorteX framework. As discussed in Section 4.3, the choice of the optimal coarse-graining parameter is governed by local or global mesh contrast, which can be easily determined either for every segment or for the whole interface. It renders the choice of the coarse-graining parameter κ fully automatic. Contrary to the stabilized Nitsche method, no knowledge about local material contrast is needed.

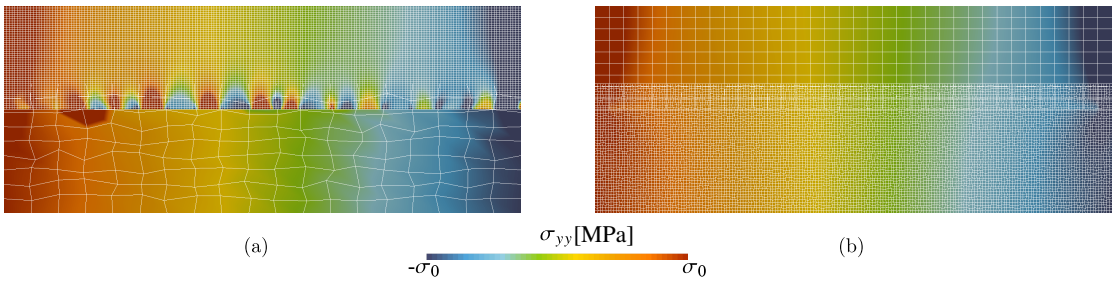


Figure 4.15: Bending stresses σ_{yy}: (a) Case 1; (b) Case 2.

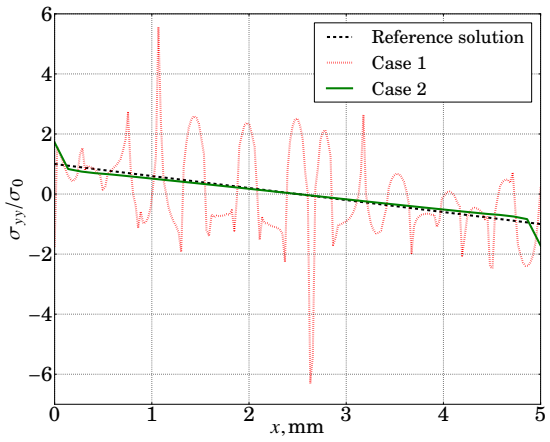


Figure 4.16: Bending stress σ_{yy} along the tying interface, comparison with the reference solution.

In Fig. 4.17, the vertical stress component is shown for the Case 1 when the coarse-graining parameter for Lagrange multipliers is set to (a) $\kappa = 6$ and (b)

$\kappa = 12$. It is clearly seen in the figures that $\kappa = 6$ does not sufficiently relaxes the over-constraining of the Lagrange multipliers space (we recall that $m_r = 11$) to obtain a smooth reference solution, even though the amplitude of oscillations is slightly reduced compared to standard Lagrange multipliers, which can be seen in Fig. 4.18(a) where the standard solution (obtained with the standard interpolation of Lagrange multipliers SLI) is compared with the coarse-grained interpolation (CGI). With the coarse-graining parameter $\kappa = 12$ we obtain a much improved result comparable to the reference solution.

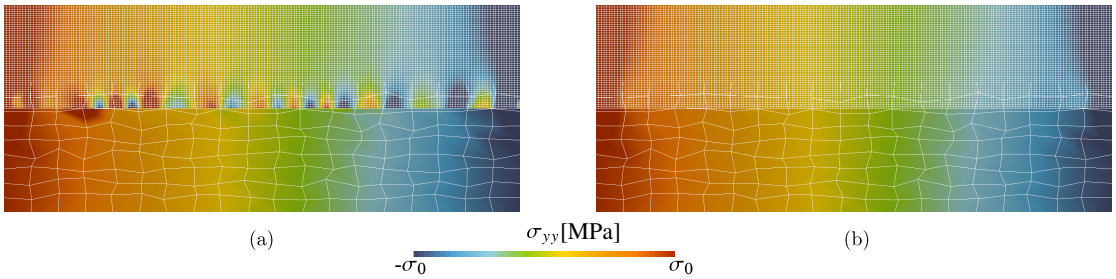


Figure 4.17: Coarse grained Lagrange multiplier space for Case 1: (a) $\kappa = 6$; (b) $\kappa = 12$.

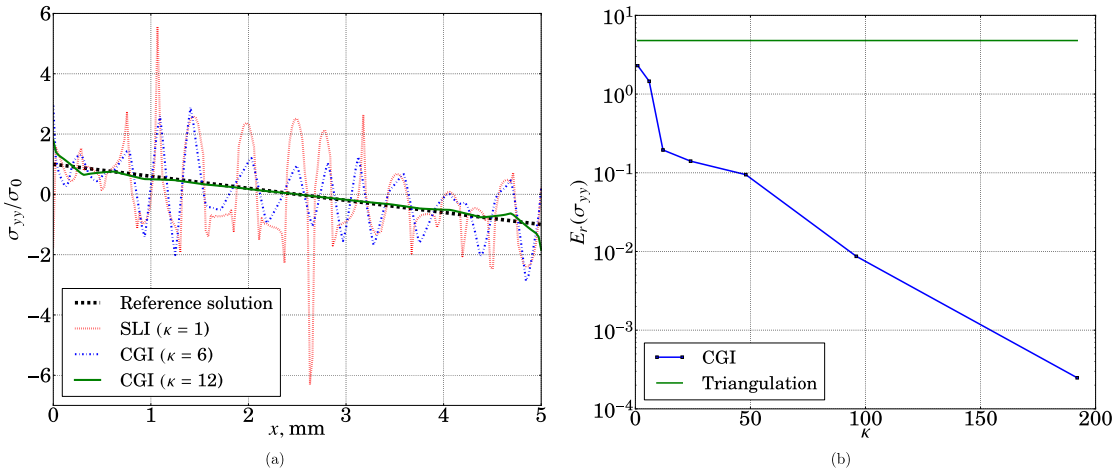


Figure 4.18: (a) Case 1: comparison of bending stresses (σ_{yy}) for SLI and CGI with the reference solution; (b) decay of the relative error $E_r(\sigma_{yy})$ for $\kappa = \{1, 6, 12, 24, 48, 96, 192\}$ for CGI in comparison with the error obtained with triangulation of blending elements.

The relative L^2 error $E_r(\sigma_{yy})$ Eq. (4.17) is shown in Fig. 4.18(b) for different values of κ. The error becomes acceptable only for $\kappa \geq m_c$, however, the accuracy of the solution constantly improves with a further increase of κ. The fast drop of the error for $\kappa \in [1, m_c]$ is associated with the graduate removal of spurious oscillations in the stress distribution, whereas for $\kappa > m_c$ improves further the error by better approximation of stresses at extremities of the interface. Since the reference stress distribution is linear, only two Lagrange multipliers are sufficient to capture it, leading to an error reduction up to $\kappa = N_m$. In general, as will be shown later, too coarse a representation of Lagrange multipliers leads to deterioration of the solution (see Section 4.5). The triangulation of blending element is also tested

in the bending patch test, see Fig. 4.19. In contrast to the compression patch test, triangulation here does not help with the removal of spurious oscillations.

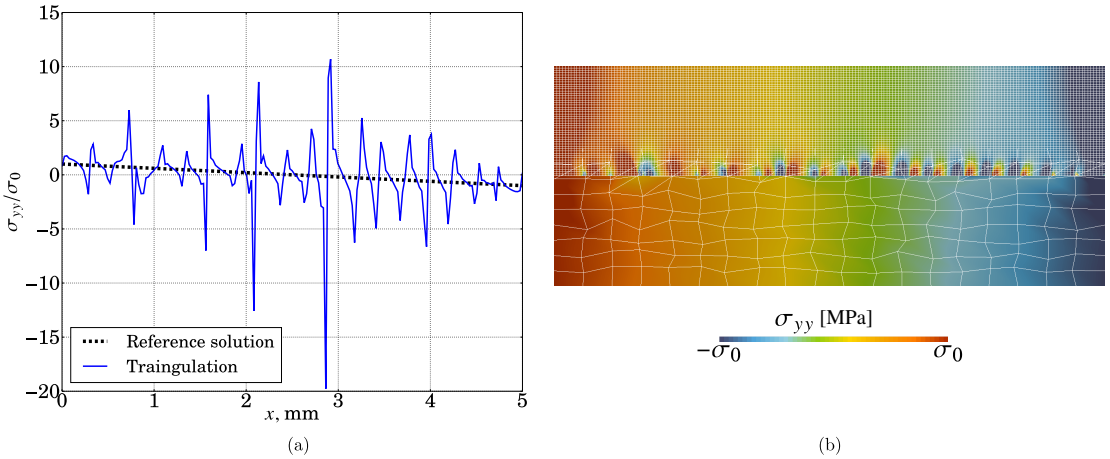


Figure 4.19: Bending stresses σ_{yy} in Case 1 for triangulated blending elements: (a) stress distribution along the tying interface; (b) contour plot of the stress component.

4.4.3 Summary of patch tests

Here, we present the ensemble of patch-test results for various combinations of host/patch meshes and material contrasts. As before, we consider two cases: Case 1 corresponds to a fine patch mesh [mesh-density contrast $m_c = 10$, Fig. 4.20(a)] which is tied with a coarser host mesh made of triangular, aligned or distorted quadrilateral elements Fig. 4.20(b,c,d), respectively. Results of Case 1 are presented in Table 4.1. In Case 2, the patch mesh [mesh-density contrast $m_c = 0.1$, Fig. 4.21(a)] is coarser than the host mesh, which again can be made of triangles, aligned or distorted quadrilateral elements, see Fig. 4.21(b,c,d), respectively. Results of Case 2 are presented in Table 4.2. Softer ($E^1/E^2 = 10^{-3}$) and stiffer ($E^1/E^2 = 1000$) patch materials are compared to the host material were considered. We also tested different interpolation order (p0 and p1) for standard Lagrange interpolation (SLI), and p1-interpolation for coarse-grained interpolation (CGI) in which the coarse-graining parameter takes its maximum value $\kappa = N_m$. The table clearly demonstrates that the tying performance is strongly dependent on the type of patch test. Cases which show a small error in bending test can demonstrate a slightly higher error in compression test as in case $m_c = 10$, $E^1/E^2 = 1000$ for distorted quads with CGI scheme. However, with a high fidelity it could be stated that if the tying method passes the bending patch test then it passes the compression patch test. The inverse is, in general, false. Interestingly, the triangulation of quadrilateral elements can considerably increase the error in case of SLI scheme, it does not happens with the CGI scheme. Clearly, from these tables it can be concluded that the CGI scheme outperforms the standard SLI scheme (both p0 and p1) in all studied combinations of mesh, element types and patch-test type (48 tests in total).

E^1/E^2	Host-mesh type	Triangulation of blending elements	Dual interpolation	L^2 error $E_r(\sigma_{yy})$ (bending patch test)	L^2 error $E_r(\sigma_{yy})$ (compression patch test)
1000	Triangles	No	SLI (p0)	1.668e+01	3.57e-06
∼	Aligned quads	No	∼	3.357e-01	0.00
∼	Distorted quads	No	∼	2.95e+00	1.064e+00
∼	∼	Yes	∼	7.66e+00	4.76e-05
1000	Triangles	No	SLI (p1)	1.266e+01	3.55e-06
∼	Aligned quads	No	∼	2.789e-01	0.00
∼	Distorted quads	No	∼	2.30e+00	7.627e-01
∼	∼	Yes	∼	4.786e+00	4e-06
1000	Triangles	No	CGI (p1)	4.53e-05	0.00
∼	Aligned quads	No	∼	4.51e-05	0.00
∼	Distorted quads	No	∼	2.4e-04	1.e-3
∼	∼	Yes	∼	2.3e-04	3.e-4
1.e-3	Triangles	No	SLI (p0)	2.192e-01	0.00
∼	Aligned quads	No	∼	2.192e-01	0.00
∼	Distorted quads	No	∼	2.186e-01	2.9e-05
∼	∼	Yes	∼	2.192e-01	2.4e-05
1.e-3	Triangles	No	SLI (p1)	2.193e-01	0.00
∼	Aligned quads	No	∼	2.194e-01	0.00
∼	Distorted quads	No	∼	2.191e-01	1.15e-05
∼	∼	Yes	∼	2.194e-01	0.00
1.e-3	Triangles	No	CGI (p1)	4.51e-05	0.00
∼	Aligned quads	No	∼	4.51e-05	0.00
∼	Distorted quads	No	∼	2.4e-04	2.8e-4
∼	∼	Yes	∼	2.3e-04	2.8e-4

Table 4.1: Patch test performance for overlapping domains with a finer patch ($m_c = 10$).

E^1/E^2	Host-mesh type	Triangulation of blending elements	Dual interpolation	L^2 error $E_r(\sigma_{yy})$ (bending patch test)	L^2 error $E_r(\sigma_{yy})$ (compression patch test)
1000	Triangles	No	SLI (p0)	2.122e-01	2.34e-05
∼	Aligned quads	No	∼	2.113e-01	0.00
∼	Distorted quads	No	∼	2.114e-01	3.58e-04
∼	∼	Yes	∼	7.66e+00	4.76e-05
1000	Triangles	No	SLI (p1)	2.493e-01	1.63e-05
∼	Aligned quads	No	∼	2.481e-01	0.00
∼	Distorted quads	No	∼	2.483e-01	5.00e-04
∼	∼	Yes	∼	4.786e+00	4e-06
1000	Triangles	No	CGI (p1)	6.00e-04	7.42e-07
∼	Aligned quads	No	∼	6.00e-04	0.00
∼	Distorted quads	No	∼	6.00e-04	1.75e-06
∼	∼	Yes	∼	6.00e-04	0.00
1.e-3	Triangles	No	SLI (p0)	1.702e-01	0.00
∼	Aligned quads	No	∼	1.702e-01	0.00
∼	Distorted quads	No	∼	1.702e-01	1.72e-06
∼	∼	Yes	∼	2.192e-01	2.4e-05
1.e-3	Triangles	No	SLI (p1)	1.734e-01	0.00
∼	Aligned quads	No	∼	1.734e-01	0.00
∼	Distorted quads	No	∼	1.735e-01	0.00
∼	∼	Yes	∼	2.194e-01	0.00
1.e-3	Triangles	No	CGI (p1)	5.5e-04	0.00
∼	Aligned quads	No	∼	5.5e-04	0.00
∼	Distorted quads	No	∼	5.5e-04	0.00
∼	∼	Yes	∼	5.5e-04	0.00

Table 4.2: Patch test performance for overlapping domains with a coarser patch mesh ($m_c = 0.1$).

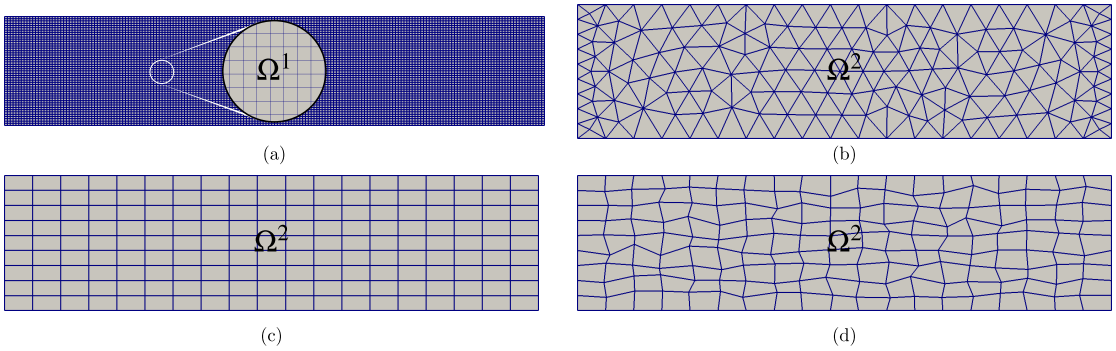


Figure 4.20: Discretized setting of coarse host mesh and finer patch mesh ($m_c \approx 10$): (a) patch mesh (Ω^1); (b) host mesh with linear triangular elements (Ω^2); (c) host mesh with bilinear quadrilateral elements (Ω^2); (d) host mesh with bilinear distorted quadrilateral elements (Ω^2).

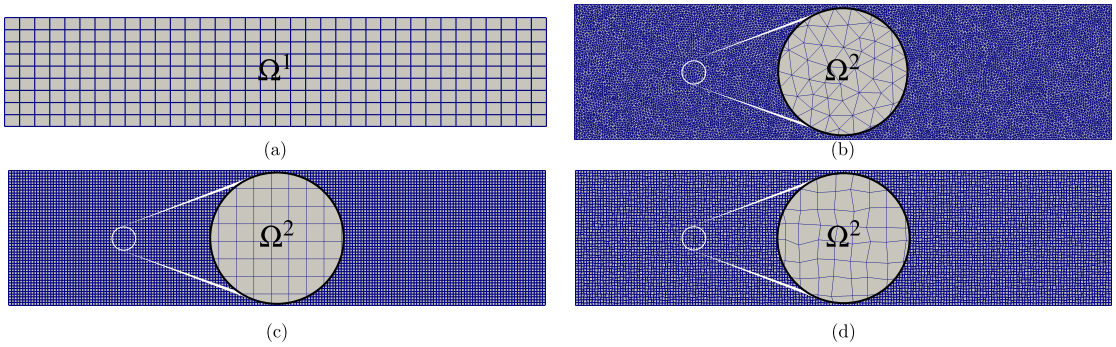


Figure 4.21: Discretized setting of fine host mesh and coarse patch mesh ($m_c \approx 0.1$): (a) patch mesh (Ω^1); (b) host mesh with linear triangular elements (Ω^2); (c) host mesh with bilinear quadrilateral elements (Ω^2); (d) host mesh with bilinear distorted quadrilateral elements (Ω^2).

4.5 Circular inclusion in infinite plane: convergence study

Having demonstrated a general good performance of the coarse-graining interpolation, here we carry out a mesh-convergence study. We focus on the worse case scenario (see Section 4.4) when the patch mesh is finer and stiffer than the host mesh. We consider a circular inclusion embedded in an infinite softer matrix in plane strain formulation, and subject to a uniform traction applied at infinity [Sharma, 1979, Kachanov et al., 2013, Hervé and Zaoui, 1995]. This particular problem represents a sub-case of a general Eshelby problem of an ellipsoidal inclusion in a matrix [Eshelby, 1959, Muskhelishvili,]. Fig. 4.22(a) shows the used computational set-up: a circular inclusion Ω^1 (patch) with radius $R = 0.1$ mm, centered at origin, is superposed on a matrix Ω^2 (host) represented by a square of side $L = 10$ mm ($L \gg R$). Linear elastic material properties are applied to both the inclusion (E^1, ν^1) and the matrix (E^2, ν^2). The inclusion is made more rigid than the matrix by choosing $E^1/E^2 = 1000$, $E^1 = 1$ GPa, the same Poisson's ratio is used for both $\nu^1 = \nu^2 = 0.3$. A uniform pressure $\sigma_0 = 0.1$ MPa is applied on the right side as shown in Fig. 4.22(a), displacements on the left side are fixed in

horizontal direction $u_x = 0$ and the lower left corner is fixed. The inclusion patch is tied to the host matrix along the boundary of the inclusion Γ_g^1.

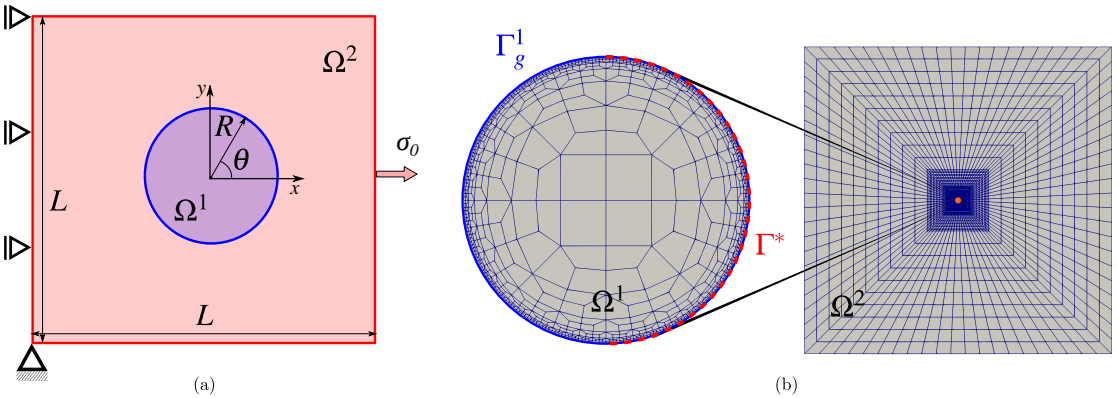


Figure 4.22: Square matrix with circular inclusion: (a) problem setting (not to scale) with inclusion domain Ω^1 (radius $R = 0.1$ mm) superposed over the matrix domain Ω^2 ($L = 10$ mm); (b) inclusion and matrix domain discretizations, where Γ_g^1 is the tying boundary and $\Gamma^* \subset \Gamma_g^1$ ($\theta \in [-\pi/2, \pi/2]$).

The analytical solution for the stress state inside and outside the inclusion is given below in polar coordinates (r, θ) [Kachanov et al., 2013]. Stress components inside the inclusion ($r < R$) are given by:

$$\sigma_{rr}^- = \frac{\sigma_0}{2}\left(\beta^1 + \delta^1 \cos 2\theta\right) \tag{4.19}$$

$$\sigma_{\theta\theta}^- = \frac{\sigma_0}{2}\left(\beta^1 - \delta^1 \cos 2\theta\right) \tag{4.20}$$

$$\sigma_{r\theta}^- = -\frac{\sigma_0}{2}\delta^1 \sin 2\theta \tag{4.21}$$

where

$$\beta^1 = \frac{\mu^1(k^2+1)}{2\mu^1 + \mu^2(k^1-1)}, \quad \delta^1 = \frac{\mu^1(k^2+1)}{\mu^2 + \mu^1 k^2}. \tag{4.22}$$

Outside the inclusion ($r > R$), the stress components are given by:

$$\sigma_{rr}^+ = \frac{\sigma_0}{2}\left[1 - \gamma^2\frac{R^2}{r^2} + \left(1 - 2\beta^2\frac{R^2}{r^2} - 3\delta^2\frac{R^4}{r^4}\right)\cos 2\theta\right] \tag{4.23}$$

$$\sigma_{\theta\theta}^+ = \frac{\sigma_0}{2}\left[1 + \gamma^2\frac{R^2}{r^2} + \left(1 - 3\delta^2\frac{R^4}{r^4}\right)\cos 2\theta\right] \tag{4.24}$$

$$\sigma_{r\theta}^+ = -\frac{\sigma_0}{2}\left(1 + \beta^2\frac{R^2}{r^2} + 3\delta^2\frac{R^4}{r^4}\right)\sin 2\theta \tag{4.25}$$

$$\tag{4.26}$$

where

$$\beta^2 = -\frac{2(\mu^1 - \mu^2)}{\mu^2 + \mu^1 k^2}, \quad \delta^2 = \frac{\mu^1 - \mu^2}{\mu^2 + \mu^1 k^2}, \quad \gamma^2 = \frac{\mu^2(k^1-1) - \mu^1(k^2-1)}{2\mu^1 + \mu^2(k^1-1)}. \tag{4.27}$$

For the considered plane strain formulation the material constants $\mu^{1,2}$ and $k^{1,2}$ are given by

$$\mu^{1,2} = \frac{E^{1,2}}{2(1+\nu^{1,2})}, \quad k^{1,2} = 3 - 4\nu^{1,2}. \tag{4.28}$$

4.5.1 Mesh convergence

We are particularly interested in the stress state along the tying boundary ($r = R$) where possible spurious oscillations take place. The distribution of the radial stress component $\sigma_{rr}(\theta)$ at $r = R$ can be obtained either from[3] Eqs. (4.19) or (4.23). This analytical solution is compared with the numerical one obtained using MorteX method along the interface $\Gamma^* \subset \Gamma_g^1$, for which $\theta \in [-\pi/2, \pi/2]$ [see Fig. 4.22(b)]. For this purpose we use the L^2 error norm as in (4.17) defined along Γ^*:

$$E_r(\sigma_{rr}) = \frac{||\sigma_{rr} - \lambda_{rr}||_{L^2(\Gamma^*)}}{||\sigma_{rr}||_{L^2(\Gamma^*)}}, \tag{4.29}$$

where σ_{rr} is the analytical solution, and λ_{rr} is the radial component of Lagrange multiplier vector obtained by projecting it on the radial basis vector of polar coordinates $\lambda_{rr} = \underline{\boldsymbol{\lambda}} \cdot \underline{\boldsymbol{e}}_r$. The mesh refinement is carried out maintaining a constant mesh contrast between the patch and the host meshes, i.e. the ratio of the mortar segments to the number of blending elements is fixed to be $m_c \approx 3$ [Fig. 4.23]; the number of mortar segments was varied $N_m \in \{256, 512, 1024, 2048\}$ (in Fig. 4.22(b) and 4.23 the coarsest mesh with $N_m = 256$ is shown). Four cases are considered for this convergence study: (i) standard p1 interpolation (SLI) is used for Lagrange multipliers; (ii) blending elements are triangulated; (iii) coarse grained interpolation (CGI) is used for Lagrange multipliers with various coarse graining parameter κ; (iv) both triangulation and coarse graining are used.

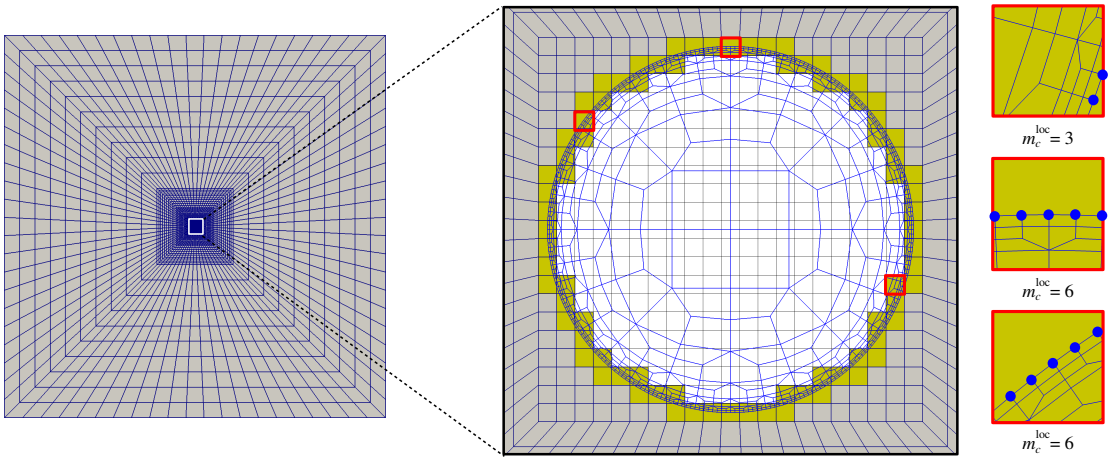


Figure 4.23: Illustration of the local mesh density contrast. The ratio of mortar segment per number of blending elements is $m_c \approx 3$, but locally the number of mortar segments intersecting different blending elements can vary considerably $m_c^{\text{loc}} = 3, 6, 6$ in zooms shown on the right.

We recall that to represent the interfacial tractions in compression and bending patch tests one or two Lagrange multipliers, respectively, were enough. In contrast to these patch tests, here the stress distribution is no longer affine along the tying interface, therefore it is expected to obtain a more practical result for the selection of the coarse graining parameter κ, which ensures optimal convergence. Results obtained for mesh contrast $m_c \approx 6$ and for $N_m = 1024$ and

[3]Note however that $\sigma_{\theta\theta}$ is not continuous across the interface

different coarse-graining are shown in Fig. 4.24. The oscillations are clearly seen near the inclusion/matrix interface, especially inside the inclusion. However, for high enough κ these oscillations are completely removed and a uniform stress field is recovered inside the inclusion. A quantitative convergence study is presented in Fig. 4.25(a). It clearly demonstrates that for standard interpolation (SLI, i.e. $\kappa = 1$) or coarse-grained interpolation (CGI) used with small values of $\kappa = \{2, 4\}$, the presence of spurious oscillations induces very high errors in interfacial tractions [see Fig. 4.26(a)]. When $\kappa = 16, 32$, the error reaches its minimum. This is due to the fact that for the given discretization, this level of coarse graining offers an appropriate balance between on the one hand the relaxation of the over-constraining of the Lagrange multipliers, and on the other hand the ability to accurately describe the complex traction field at the interface. For higher values of $\kappa = \{64, 128\}$ the error increases again because of too coarse representation of interfacial tractions. It is thus expected that, in general case, there exists a range for κ which ensures oscillation free and accurate enough solution. It is also expected that optimal κ is determined by the global mesh density contrast m_c. However, as demonstrated in Fig. 4.23, the local mesh density contrast can be more pronounced than the average one, therefore it is expected that the optimal value of coarse graining parameter κ lies in the range $\kappa > m_c$; for the considered case the error is minimized for $\kappa/m_c = \{2.667, 5.333\}$, probably, the optimal value lies in between. The effect of optimal κ is clearly demonstrated in Fig. 4.26(b) where interfacial tractions λ_{rr} for different κ are plotted.

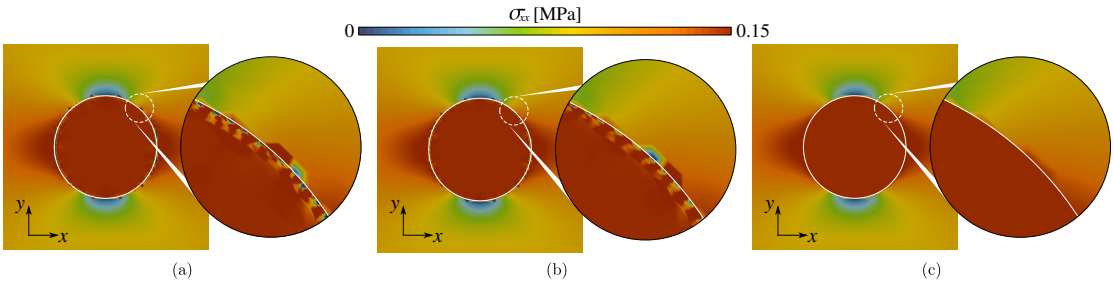


Figure 4.24: Stress component σ_{xx} computed using MorteX method: (a) standard interpolation (SLI); (b) coarse grained interpolation (CGI) for $\kappa = 4$; (c) CGI for $\kappa = 16$.

For the fixed mesh contrast $m_c = 6$ and optimal $\kappa = 16$ and sub-optimal $\kappa = 8$, the mesh convergence was carried out with meshes of different densities $N_m \in \{128, 256, 512, 1024, 2048\}$. In Fig. 4.25(b) we plot the error decay with decreasing mesh size, for which we select the length of mortar edge normalized by the total length of the interface $h/2\pi R = 1/N_m$. For the selected error-measure along the inclusion/matrix interface, the standard interpolation for Lagrange multiplier (SLI) results in optimal convergence ($E_r \sim h$). However, even though the convergence is optimal, the error remains very high due to the spurious oscillations, implying that an excessively fine mesh would be required to achieve an acceptable error. For example, to reach $E_r = 0.1$ % in case of SLI, 51 200 elements on the mortar side would be needed. Moreover, for quadrilateral host mesh in absence of triangulation of blending elements, the convergence is lost for very fine meshes. In contrast, the coarse graining technique (CGI) used with optimal $\kappa = 16$ results in the error below 0.1 %, even with the coarsest mesh used in our study $N_m = 128$.

At the same time, the optimal convergence is preserved. As expected, the triangulation of the blending elements slightly deteriorates the quality of the solution, but preserves the optimality of the convergence.

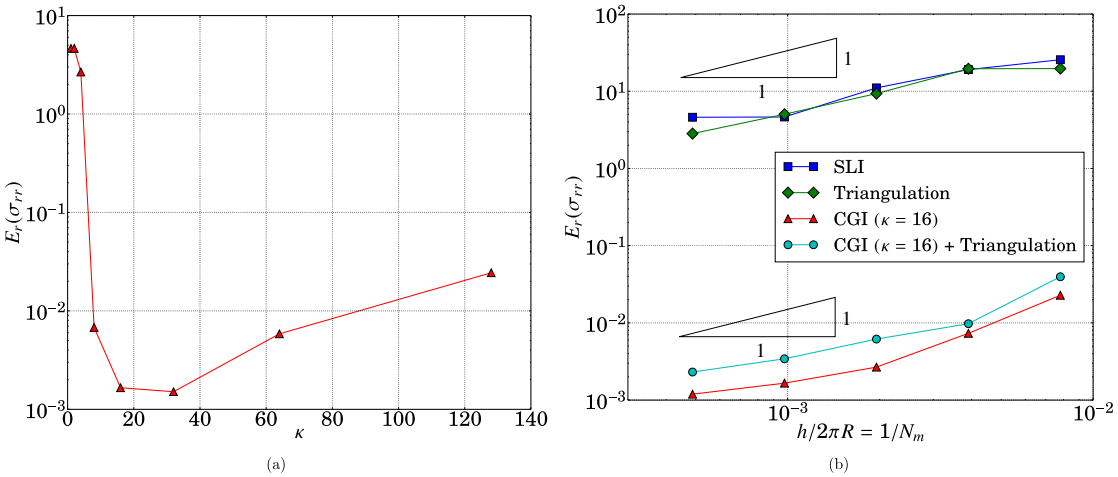


Figure 4.25: Convergence study results for the circular inclusion problem for mesh contrast $m_c \approx 6$: (a) $E_r(\sigma_{rr})$ error change with coarse-graining parameter κ; (b) comparison of convergence of SLI and CGI with and without triangulation of blending elements, the mesh size along the interface is normalized by the circumference $h/2\pi R = 1/N_m$ which is equivalent to the inverse of the number of mortar segments.

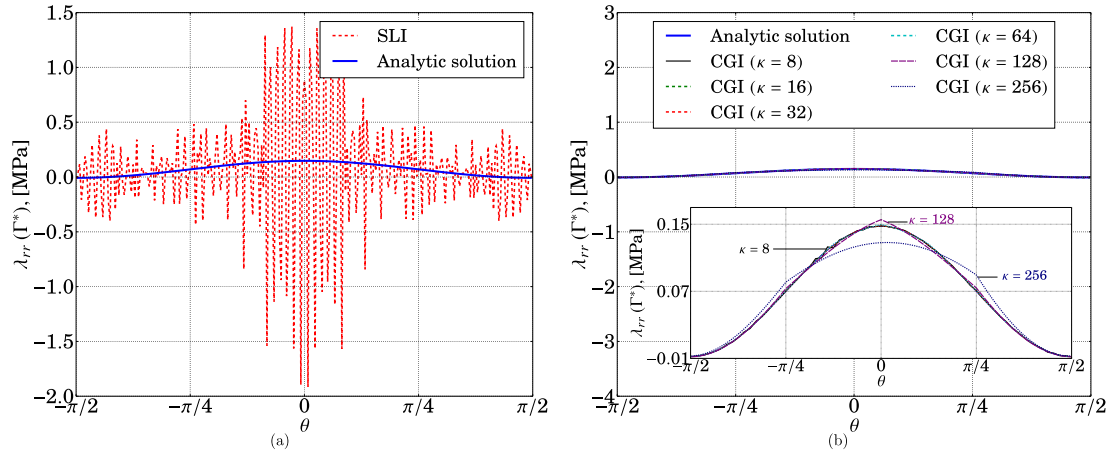


Figure 4.26: Comparison of λ_{rr} with analytical solution for various values of κ along Γ^* (for the mesh with $N_m = 1024$): (a) the standard Lagrange multiplier spaces ($\kappa = 1$); (b) the coarse grained Lagrange multiplier solution for $\kappa \in [8, 32, 64, 128, 256]$.

The effect of the parameter m_c on the amplitude of spurious oscillations is demonstrated in Fig. 4.27. For a fixed host-mesh discretization we increase the number of mortar edges $N_m = [64, 128, 256]$, which correspond to $m_c \approx \{1, 3, 6\}$. Fig. 4.27(a) demonstrates the increase in the amplitude of oscillations with increasing mesh contrast m_c for SLI interpolation. The removal of spurious oscillations with the CGI scheme is shown in Fig. 4.27(b) for reasonable choice of coarse graining parameter $\kappa = \{2, 4, 8\}$ for $m_c \approx \{1, 3, 6\}$, respectively.

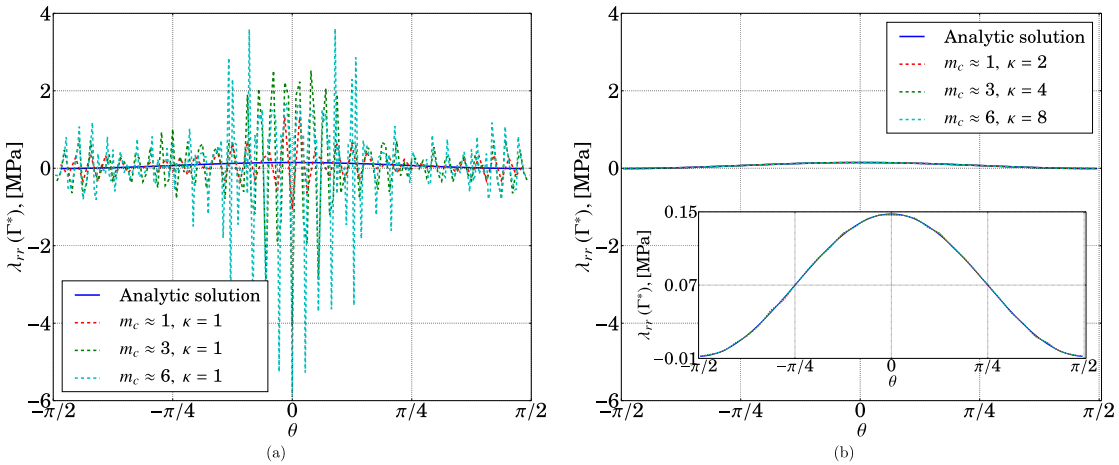


Figure 4.27: The effect of m_c on λ_{rr} along Γ^*: (a) with standard Lagrange multiplier spaces ($\kappa = 1$); (b) with coarse grained Lagrange multiplier spaces ($\kappa \approx m_c$).

4.6 Numerical examples

In this section we illustrate the method of mesh tying along embedded interfaces in light of potential applications. In all the presented examples, we use a linear elastic material model under plane strain assumption. All triangular and quadrilateral elements used in simulations possess three and four Gauss points for integration, respectively. The triangulated blending elements also use three Gauss points. The MorteX interface uses three integration points to evaluate the MorteX integrals.

4.6.1 Plate with a hole

As a first example, we solve the problem of a square plate with an embedded square patch containing a circular hole, which was used to illustrate the method in Section 4.1 [Fig. 4.1]. This example demonstrates the ease with which arbitrary geometrical features can be included into the host mesh. Classically, in the X-FEM method a void can be easily included in the host mesh, however, in the vicinity of the void a stronger stress gradients take place, therefore the mesh around the void should be properly refined. It can be easily achieved by surrounding the void with a finer patch mesh, as done here, and by embedding this refined geometry in a coarse host mesh. The geometric dimensions used in the problem are the following: the plate's side is $L^H = 12$ mm, the hole's radius is $R = 0.75$ mm and the patch's side is $L^P = 4.5$ mm [Fig. 4.28(a)]. The patch and the host are made of the same material with Young's modulus $E = 1000$ MPa and Poisson's ratio $\nu = 0.3$. The left edge of the host domain is fixed in x ($u_x = 0$), the lower left corner is fixed, and a uniform traction $\sigma_0 = 1$ MPa is applied on the right edge, the upper and lower boundaries remain free $\sigma_{xy} = \sigma_{yy} = 0$. For comparison purposes, a reference solution is obtained with a classical monolithic mesh [Fig. 4.28(c)].

Fig. 4.29(a) shows a rather smooth contour of stress component σ_{xx}, which was obtained using MorteX tying with SLI scheme. However, the seemingly smooth stress field near the interface, exhibits oscillations near the interface as can be seen in Fig. 4.29(b), where σ_{xx} was plotted over a part of the interface Γ^*. These oscillations have a smaller amplitude than in cases with high material contrast,

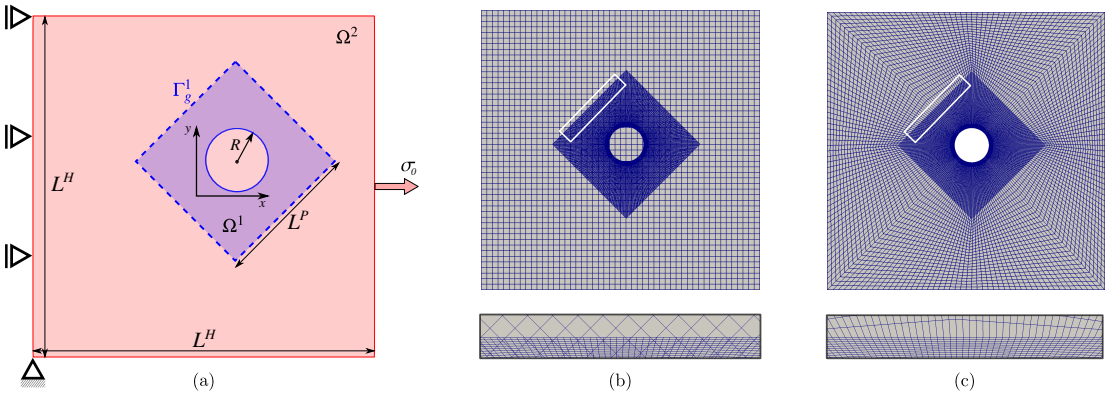


Figure 4.28: Plate with a hole: (a) overlapping domain setting; (b) discretized overlapping domains; (c) monolithic discretization used to obtain the reference solution.

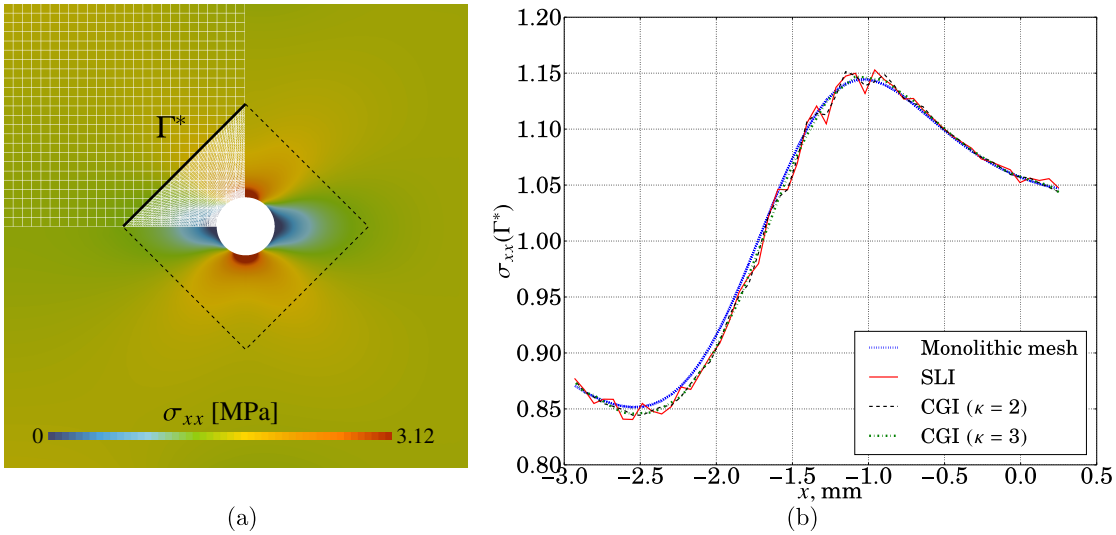


Figure 4.29: Regular host mesh with an embedded patch containing a circular hole ($m_c \approx 3$): (a) contour plot of σ_{xx} stress component obtained using MorteX (SLI scheme), (mesh is shown only in the second quadrant, the interface is marked with a dashed line; (b) comparison of σ_{xx} distributions along Γ^* between the solution obtained with a monolithic mesh [Fig. 4.28(b)] and MorteX solution obtained with overlapping meshes [Fig. 4.28(a)] using SLI and CGI schemes.

and as previously, they can be efficiently removed when coarse-grained interpolation CGI is used, what is shown in the same figure. Coarse-graining parameter $\kappa = 3$ appears to be sufficient to remove them. Note that the slight difference between the MorteX tying and the monolithic mesh comes from inherently different extrapolation/interpolation of stresses to the interface nodes.

4.6.2 Crack inclusion in a complex mesh

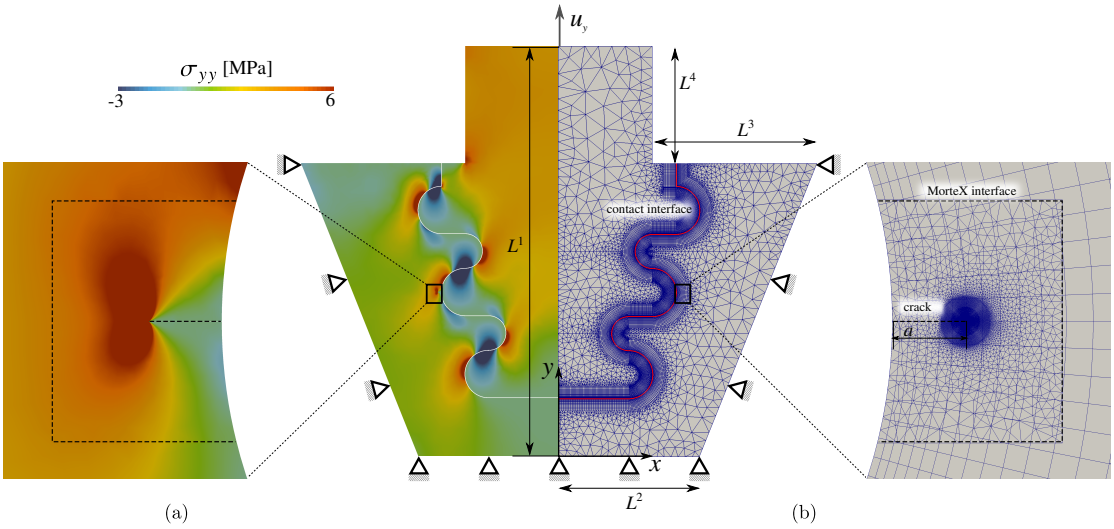


Figure 4.30: Problem setting for the model blade-disk assembly with an embedded patch mesh with a crack: (a) shows stress component σ_{yy} contour plot and a zoom near the tying interface; (b) shows used host and patch mesh.

In many engineering applications, the solids are subjected to cyclic loads and therefore modeling of structures with fatigue cracks appears essential for computational lifespan prediction. The structural finite element analysis can indicate potential locations of the onset of fatigue cracks, however, insertion of cracks is not always trivial [Proudhon et al., 2016, Feld-Payet et al., 2015], especially, in the common case where the original CAD model is not available. Moreover, the position of the onset of the crack is subjected to statistical perturbations, therefore it is often of interest to probe various scenarios in which the crack starts at different locations. Within the proposed framework, studying various fracture scenarios (crack in this case) merely implies placing the patch at a different location on a host mesh, avoiding potential creation of conformal geometries. Here we demonstrate an example of incorporating a crack in a model blade-disk fir-tree connection subject to a vertical tensile load. The frictionless contact is handled using the augmented Lagrangian method in the framework of the standard mortar method. The following dimensions [see Fig. 4.30(a)] are used for the blade disk assembly: $L^1 = 35$ mm, $L^2 = 12$ mm, $L^3 = 14$ mm and $L^4 = 10$ mm. The Young's modulus is $E = 1000$ MPa and $\nu = 0.3$ for the blade, disk, and the patch containing the crack of length $a = 0.3$ mm. A vertical displacement $u_y = 0.2$ mm is applied on the top surface of the blade. In Fig. 4.30(a) we present the resulting stress field for the case of intact structure and for the case of a structure with embedded crack, respectively. As seen in the later case, the stress fields are very smooth across the tying interface (shown as white dashed boundary) ensured by MorteX with SLI only. The coarse graining is not needed here as the mesh densities are comparable and the same material is used for the patch and for the host. Similarly to the presented case of crack insertion, the method can be used in general for introducing various other geometric features into the existing mesh. Using the MorteX method, the location/orientation of these features can be adjusted with ease and without remeshing to perform a sensitivity analysis.

4.6.3 Multi-level submodeling: patch in a patch

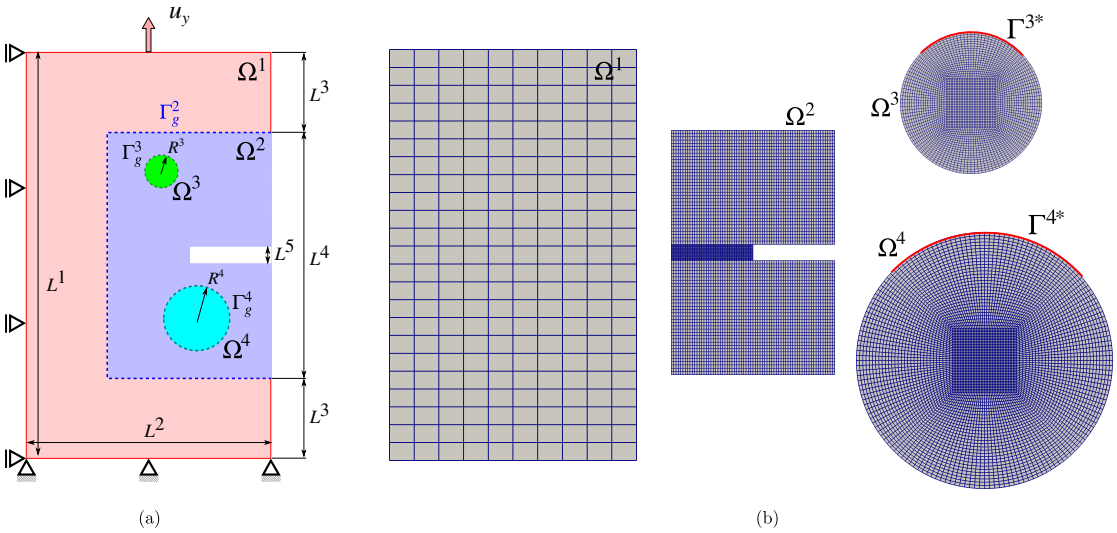


Figure 4.31: (a) Multi-level overlapping domain set-up; (b) finite element meshes to be coupled (shown not in proportion), Γ^{3*}, Γ^{4*} denote interfaces over which CGI and SLI are compared in Fig. 4.33.

In this example we demonstrate the ability of the MorteX method to handle multi-level overlapping domains, i.e. when an embedded patch mesh hosts other domains. In Fig. 4.31(a) we present such a scenario where a patch with a notch (Ω^2) is embedded into a host domain Ω^1, both made of the same material. At the same time, the patch Ω^2 itself hosts 2 circular inclusions ($\Omega^{3,4}$), which are stiffer than the surrounding material. The following dimensions are used: $L^1 = 5$ mm, $L^2 = 3$ mm, $L^3 = 1$ mm, $L^4 = 3$ mm, $R^3 = 0.2$ mm, and $R^4 = 0.4$ mm. The material properties used are: $E^1 = 1.0$ MPa, $E^2 = 1.0$ MPa, $E^3 = 100.0$ MPa, $E^4 = 1000.0$ MPa (the upper indices correspond to the domains Ω^1, Ω^3, Ω^3, and Ω^4 respectively). A Poisson's ratio of $\nu = 0.3$ is used for all the domains. A vertical displacement $u_y = 0.1$ mm is applied on top surface of the Ω^1, while the left surface is fixed in the x direction and bottom is fixed in all directions. The contour plots of σ_{yy} for the cases of SLI and CGI are shown in Fig. 4.32(a) and (b), respectively. The oscillations in the stress are distinctly seen in case of SLI, but they are removed by applying CGI with $\kappa = 4$. Fig. 4.33 compares σ_{yy} along Γ^{3*}, Γ^{4*}, which form $\pi/2$ portions of matrix/inclusion interfaces. This example illustrates the case where a host mesh with embedded domains of different material properties can be dealt within CGI MorteX scheme. Note that in contrast to the Nitsche method where the stabilization parameter needed to avoid mesh locking is dependent on the local material contrasts [Sanders et al., 2012], the CGI stabilization does not require the knowledge of this contrast. In the CGI scheme, knowing a local or a global contrast of mesh densities across the tying interface m_c is enough to automatically select the coarse graining parameter κ, which efficiently stabilizes the mixed formulation and removes spurious oscillations present in the standard mortar scheme.

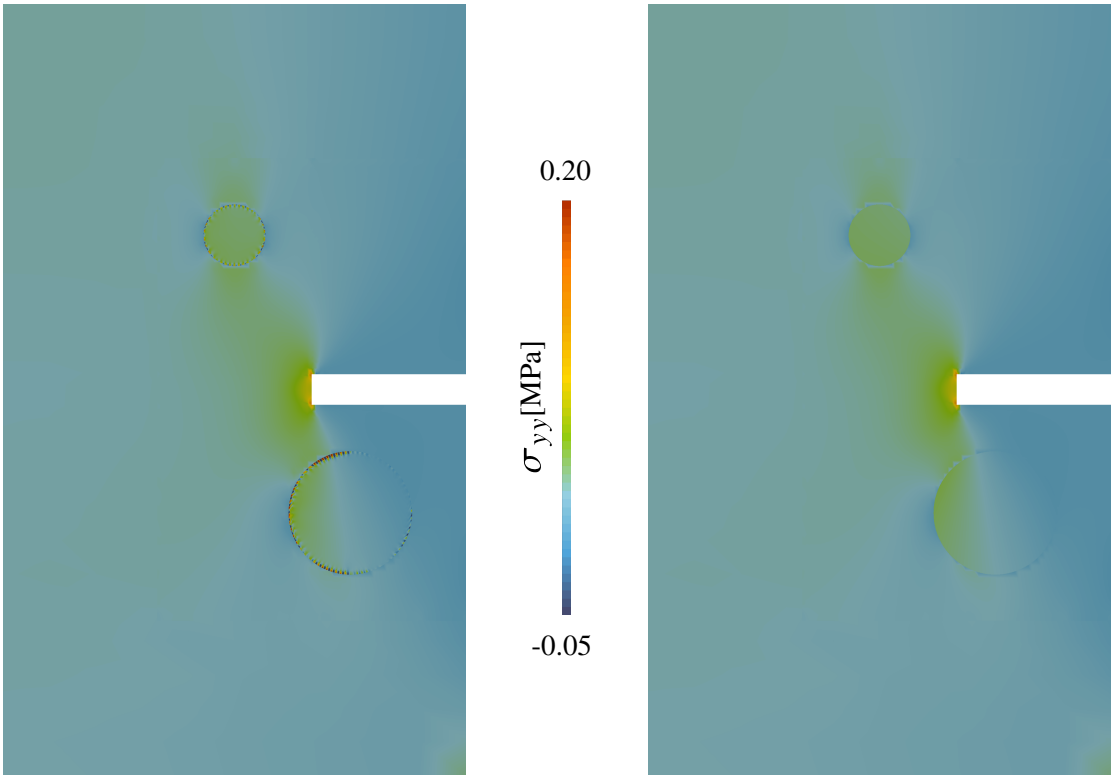


Figure 4.32: Contour plots of σ_{yy} for multi-level overlapping domains: (a) SLI; (b) CGI used with $\kappa = 4$.

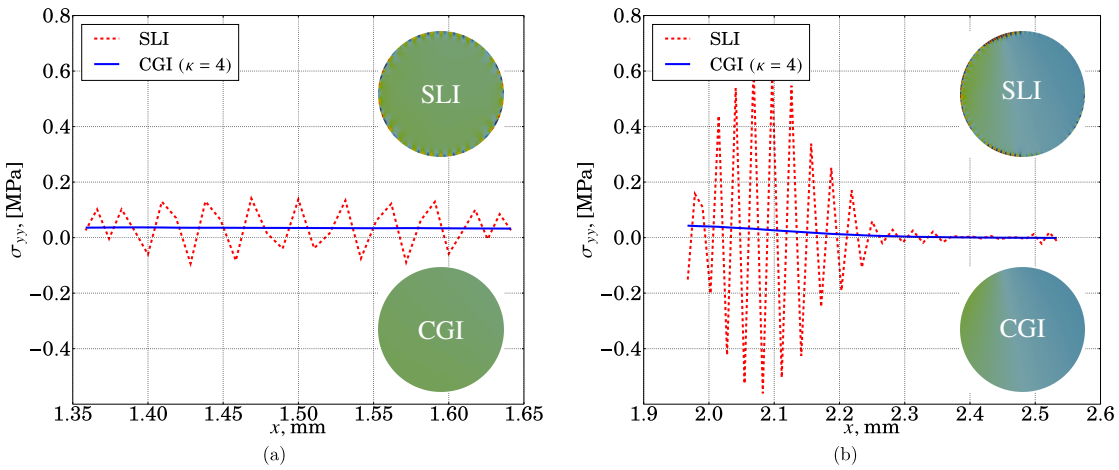


Figure 4.33: Comparison of σ_{yy} stress for standard (SLI) and coarse grained (CGI) interpolations along : (a) Γ^{3*}; (b) Γ^{4*}.

4.7 Summary

We presented a unified framework for mesh tying between overlapping domains. This framework was entitled MorteX as it combines features of the mortar and X-FEM methods. As known, the resulting mixed finite element problem may be prone to mesh locking phenomena especially for high material or mesh-density contrasts between the host and the patch meshes. Manifestation of the emerging spurious oscillations for different element types and various material as well

as mesh contrasts was illustrated on two patch tests (bending and compression) and on selected examples. These oscillations strongly deteriorate solution in the vicinity of interfaces resulting in poor mesh convergence. Even though triangular elements help to avoid oscillations in compression patch tests, they do not perform well in bending patch test, nor in more complicated examples. These oscillations comes from the over-constraining of the interface in case of mesh-density contrast, when few mortar-side nodes located on the patch mesh are tied to displacement field of a single host element. To get rid of the resulting mesh locking, we suggested to coarse-grain interpolation (CGI) of Lagrange multipliers by interpolating the associated field along few mortar edges. It implies that only every $(\kappa + 1)$ node along the mortar side stores a Lagrange multipliers and a linear interpolation is used in between. The value of coarse-graining spacing parameter κ controls the performance of the scheme. If κ is too small compared to mesh-density contrast, the spurious oscillations persist as in standard interpolation of Lagrange multipliers (SLI). If the value of κ is too high, the spatial variation of resulting interface tractions cannot be captured properly. Therefore, in general problem, there exists an optimal choice for the spacing parameter κ which can be automatically determined either by local mesh-density contrast between the patch and the host mesh or by the global mesh-density contrast. The performance of the MorteX method with coarse-grained interpolation was demonstrated on Eshelby problem for a stiff inclusion in a softer matrix (elastic contrast of 1000). Few other examples, demonstrating the ease with which the method can be used for: submodeling, local mesh refinement and inclusion of arbitrary geometrical features in the existing mesh, without remeshing. Among these examples, a multi-level/hierarchical overlapping is shown, where a patch is inserted into a host mesh, which in turn is inserted into another host mesh. The MorteX method equipped with CGI demonstrates a very good performance, removes the mesh locking oscillations, and ensures optimal convergence. The important feature of the method is that its stabilization requires knowledge of local mesh densities only, thus it presents a good alternative to the Nitsche method, which requires stabilization constructed on a priori knowledge of material stiffness in the interface. In analogy with the classical mortar method, the MorteX method can be extended to handle contact problems along virtual interfaces embedded in a mesh; this extension is presented in Chapter 5.

Chapter 5

Extended mortar methods for contact

Résumé: Dans ce chapitre, le cadre MorteX présenté dans le chapitre 4 est élargi afin de traiter les problèmes de contact le long des surfaces immergées. L'application de contraintes d'inégalité le long des surfaces immergées rajoute des difficultés supplémentaires et nécessite d'affiner le cadre de calcul. L'applicabilité des techniques de stabilisation proposées pour le problème de couplage dans le cadre MorteX sera démontrée pour les problèmes de contact.

In this chapter we extend the MorteX framework presented in Chapter 4 to treat contact problems along the embedded surfaces. Enforcing inequality constraints along embedded surfaces brings in additional complexities and requires refinement of the computational framework. The applicability of the stabilization techniques proposed for the tying problem within the MorteX framework will be demonstrated for contact problems.

5.1 MorteX framework

It consists of two distinct procedures as for the tying problem in MorteX framework. The first procedure accounts for the internal virtual work on the effective volume $\tilde{\omega}^2$ of the host mesh: the one hosting the embedded surface $\tilde{\gamma}_c^2$ which is accomplished with the X-FEM method. The second procedure deals with the enforcement of the contact constraints between the "real" surface γ_c^1 (explicitly represented by mesh edges) and the embedded virtual surface $\tilde{\gamma}_c^2$ (geometrically non-adhering internal surface).

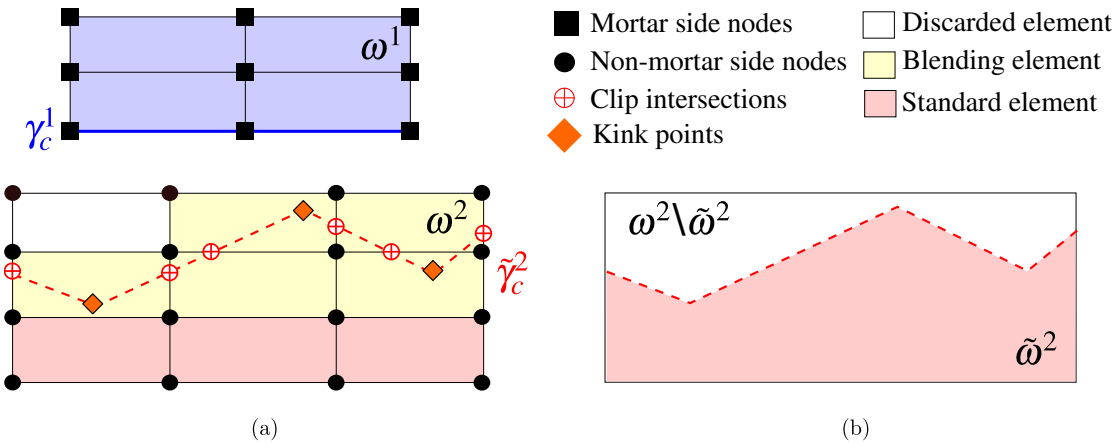


Figure 5.1: (a) Discrete contact interface pair: Real γ_c^1 (mortar) and embedded $\tilde{\gamma}_c^2$ (non-mortar); (b) effective volume of the host domain $\tilde{\omega}^2$.

5.1.1 Extended finite elements

The underlying numerical philosophy of selective integration remains the same for the contact problem as for the tying one. However, unlike in the setting of tying between overlapping domains, the embedded surface $\tilde{\gamma}_c^2$ is not an imprint of the surface of the other body ω_1 (referred to as "patch" for the tying problems) under consideration. Instead, it is independently and explicitly defined based on the surface contour desired [see Fig. 5.1].

5.1.2 MorteX interface discretization

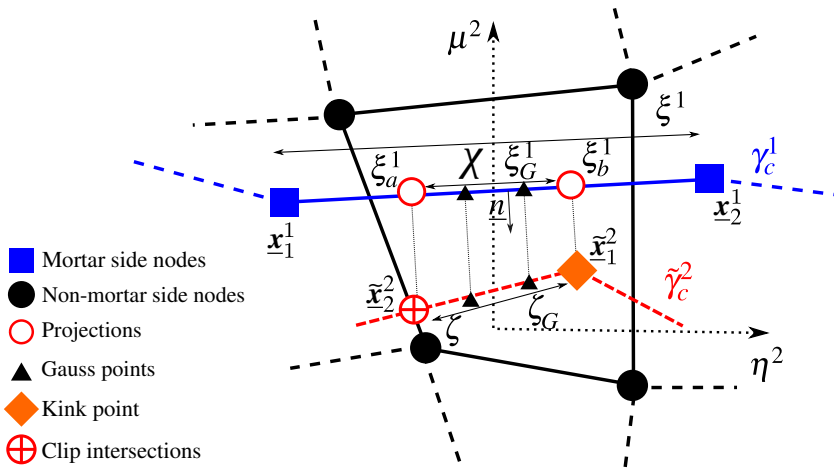


Figure 5.2: Example of mortar domain.

Similar to the tying problem in the MorteX framework, the rationale behind the choice of mortar and non-mortar sides, holds for the contact problems as well [Section 4.1.2]. The interpolations of the nodal positions in the current configuration and the Lagrange multipliers along the mortar side γ_c^1 remain the same as for the classical schemes, and are given by:

$$\underline{\boldsymbol{x}}^1(\xi^1) = N_m(\xi^1)\underline{\boldsymbol{x}}_m^1, \quad m \in [1, \mathrm{M}], \tag{5.1}$$

$$\lambda_{n,\tau}(\xi^1) = \Phi_l(\xi^1)\lambda^l_{n,\tau}, \quad l \in [1, \mathrm{L}]. \tag{5.2}$$

where as previously M and L is the number of nodes per mortar segment and those carrying Lagrange multipliers respectively. Vector $\underline{\boldsymbol{x}}^1_m$ represents the nodal positions of the mortar segment and $\xi^1 \in [-1; 1]$ is the parametric coordinate of the mortar side. The parametrization on the non-mortar side requires more consideration. As shown in Fig. 5.2, the embedded surface $\tilde{\gamma}^2_c$ is divided into straight segments, whose vertices can either be a clip intersection (i.e. the intersection between $\tilde{\gamma}^2_c$ and the edge of a bulk element of the non-mortar side) or a kink point (a vertex of the discrete virtual surface $\tilde{\gamma}^2_c$ that lies inside a blending element). Hereafter, we will refer these vertices as "points of interest", without distinction between kink and clip intersection points. Here, we introduce notations for the points of interest $\underline{\tilde{\boldsymbol{x}}}^2_j, j \in [1, 2]$. As a result, the geometric interpolation in the current configuration along the embedded surface $\tilde{\gamma}^2_c$, parametrized by $\zeta \in [-1; 1]$ can be defined using two-dimensional quantities inherited from on the host element [see Fig. 5.2]:

$$\underline{\boldsymbol{x}}^2(\zeta) = N_i\left(\mu^2(\zeta), \eta^2(\zeta)\right)\underline{\boldsymbol{x}}^2_i. \tag{5.3}$$

Irrespective of the fact that the embedded surface is assumed piece-wise linear in the reference configuration, it can become piece-wise non-linear in the current configuration, therefore the displacement along this line cannot be parameterized by linear shape functions. The kink points here in MorteX contact set-up represent the vertices of the discretized surface $\tilde{\gamma}^2_c$, whereas they represented the mortar side nodes in case of MorteX tying set-up [see Section 4.1.2].

MorteX contact element

Here, a mortar contact element is formed with a single mortar segment and a blending element. Each contact element consists of (M + N) nodes, M from the mortar segment, and N from blending element. In addition the contact element stores L, L × dim Lagrange multipliers for the frictionless and frictional cases, respectively. As was shown in Chapter 4, to avoid mesh-locking (or over-constraining of the interface), which results in spurious oscillations, the efficient choice of the number of Lagrange multipliers should be balanced by the number of blending elements, i.e. often it is reasonable to use one set per blending element.

Discrete integral kinematic quantities

The aforementioned selective integration scheme, used to accommodate for the presence of an embedded virtual interface, leads to changes in the discrete contact integral quantities: the integral gap vector (3.13), the integral normal gap (3.17) and the incremental slip (3.27). For a MorteX contact element, these nodal quantities are now evaluated along the interface formed by a pair of real and embedded segments. Tilde notations are used for denoting these quantities.

$$\underline{\tilde{\tilde{\boldsymbol{g}}}}^l = \left[D_{lm}\underline{\boldsymbol{x}}^1_m - \tilde{M}_{li}\underline{\boldsymbol{x}}^2_i\right], \tag{5.4}$$

(integral gap vector)

$$\tilde{\tilde{g}}^l_n = \underline{\tilde{\tilde{\boldsymbol{g}}}}^l \cdot \underline{\boldsymbol{n}}, \tag{5.5}$$

(integral normal gap)

$$\mathring{\tilde{\tilde{g}}}_{\tau}^{l} = -\underline{\boldsymbol{\tau}} \cdot \left[\left(D_{lm}(t_j) - D_{lm}(t_{j-1}) \right) \underline{\boldsymbol{x}}_m^1(t_{j-1}) - \left(\tilde{M}_{li}(t_j) - \tilde{M}_{li}(t_{j-1}) \right) \underline{\boldsymbol{x}}_i^2(t_{j-1}) \right] \tag{5.6}$$

(nodal incremental slip)

where D_{lm} (4.5) and $\tilde{M}_{li}$ (4.6) are the mortar integrals in the MorteX framework. Note that within the MorteX framework, the definitions of purely mortar side quantities, like the mortar segment normal $\underline{\boldsymbol{n}}$, remain the same.

Discrete contact virtual work

The MorteX residual vector is larger than the standard mortar one as it involves the displacement DoFs from the bulk of the non-mortar side of the interface. The discretized form of the virtual work for a contact element for the frictionless case is:

$$\delta \tilde{W}_{\mathrm{c}}^{\mathrm{el}} = \begin{bmatrix} \delta \underline{\boldsymbol{x}}_m^1 \\ \delta \underline{\boldsymbol{x}}_i^2 \\ \delta \lambda_n^l \end{bmatrix}^{\mathsf{T}} \begin{bmatrix} \left(\hat{\lambda}_n^l \underline{\boldsymbol{n}} \right) D_{lm} \\ - \left(\hat{\lambda}_n^l \underline{\boldsymbol{n}} \right) \tilde{M}_{li} \\ \tilde{\tilde{g}}_n^l \end{bmatrix}, \qquad \hat{\lambda}_n^l \leq 0 \, (\text{in contact}) \tag{5.7}$$

$$\delta \tilde{W}_{\mathrm{nc}}^{\mathrm{el}} = \begin{bmatrix} \delta \underline{\boldsymbol{x}}_m^1 \\ \delta \underline{\boldsymbol{x}}_i^2 \\ \delta \lambda_n^l \end{bmatrix}^{\mathsf{T}} \begin{bmatrix} 0 \\ 0 \\ -\frac{1}{\varepsilon_n} \lambda_n^l \end{bmatrix}, \qquad \hat{\lambda}_n^l > 0 \, (\text{no contact}). \tag{5.8}$$

The residuals for stick, slip and no contact statuses in case of frictional contact are:

$$\delta \tilde{W}_{\mathrm{stick}}^{\mathrm{el}} = \begin{bmatrix} \delta \underline{\boldsymbol{x}}_m^1 \\ \delta \underline{\boldsymbol{x}}_i^2 \\ \delta \lambda_n^l \\ \delta \lambda_\tau^l \end{bmatrix}^{\mathsf{T}} \begin{bmatrix} \left(\hat{\lambda}_n^l \underline{\boldsymbol{n}} + \hat{\lambda}_\tau^l \underline{\boldsymbol{\tau}} \right) D_{lm} \\ - \left(\hat{\lambda}_n^l \underline{\boldsymbol{n}} + \hat{\lambda}_\tau^l \underline{\boldsymbol{\tau}} \right) \tilde{M}_{li} \\ \tilde{\tilde{g}}_n^l \\ \mathring{\tilde{\tilde{g}}}_\tau^l \end{bmatrix}, \qquad \hat{\lambda}_n^l \leq 0, \, |\hat{\lambda}_\tau^l| \leq -\mu \hat{\lambda}_n^l \, (\text{stick}) \tag{5.9}$$

$$\delta \tilde{W}_{\mathrm{slip}}^{e} = \begin{bmatrix} \delta \underline{\boldsymbol{x}}_m^1 \\ \delta \underline{\boldsymbol{x}}_i^2 \\ \delta \lambda_n^l \\ \delta \lambda_\tau^l \end{bmatrix}^{\mathsf{T}} \begin{bmatrix} \left(\hat{\lambda}_n^l \underline{\boldsymbol{n}} - \mu \hat{\lambda}_n^l \mathrm{sign}(\hat{\lambda}_\tau) \underline{\boldsymbol{\tau}} \right) D_{lm} \\ - \left(\hat{\lambda}_n^l \underline{\boldsymbol{n}} - \mu \hat{\lambda}_n^l \mathrm{sign}(\hat{\lambda}_\tau) \underline{\boldsymbol{\tau}} \right) \tilde{M}_{li} \\ \tilde{\tilde{g}}_n^l \\ -\frac{1}{\varepsilon_\tau} \left(\lambda_\tau^l + \mu \hat{\lambda}_n^l \mathrm{sign}(\hat{\lambda}_\tau^l) \right) \end{bmatrix}, \qquad \hat{\lambda}_n^l \leq 0, \, |\hat{\lambda}_\tau^l| > -\mu \hat{\lambda}_n^l \, (\text{slip}) \tag{5.10}$$

$$\delta \tilde{W}_{\mathrm{nc}}^{e} = \begin{bmatrix} \delta \underline{\boldsymbol{x}}_m^1 \\ \delta \underline{\boldsymbol{x}}_i^2 \\ \delta \lambda_n^l \\ \delta \lambda_\tau^l \end{bmatrix}^{\mathsf{T}} \begin{bmatrix} 0 \\ 0 \\ -\frac{1}{\varepsilon_n} \lambda_n^l \\ -\frac{1}{\varepsilon_\tau} \lambda_\tau^l \end{bmatrix}, \qquad \hat{\lambda}_n^l > 0 \, (\text{no contact}). \tag{5.11}$$

It is important to remark that the contact status is again based on the integral quantities (5.5, 5.6):

$$\hat{\lambda}_n^l = \lambda_n^l + \varepsilon_n \tilde{\tilde{g}}_n^l, \quad \hat{\lambda}_\tau^l = \lambda_\tau^l + \varepsilon_\tau \mathring{\tilde{g}}_\tau^l.$$

Evaluation of integrals

The numerical procedures of the classical mortar framework to evaluate the mortar integrals need to be adapted for the MorteX framework similar to what was done for the tying problem. However, the numerical procedure adapted for the tying problem in MorteX framework cannot be used directly for the contact problems. The tying problem does not require any projections between the tying surfaces $\tilde{\Gamma}_g^2$ and Γ_g^1. This simplification for linear tying problem can be attributed to the fact that $\tilde{\Gamma}_g^2$ is an imprint of the embedded surface of the patch domain ($\tilde{\Gamma}_g^2 \equiv \Gamma_g^1$) in the reference configuration and remains so in any configuration. However, for the non-linear contact problem, where the contact interface continuously changes in response to the deformations the bodies undergo, the conformity of the surfaces cannot be ensured in the reference configuration and certainly not in the current configuration ($\tilde{\gamma}_c^2 \not\equiv \gamma_c^1$). This implies a need for a projection step similar to the classical mortar framework to determine the limits of mortar domain S^{el}. The quantities projected are points of interest rather than non-mortar segment nodes like it was in the classical mortar. The mortar projection coordinates are found by solving the following equation:

$$\left[N_m(\xi^1)\underline{\boldsymbol{x}}_m^1 - \underline{\tilde{\boldsymbol{x}}}_i^2\right] \times \underline{\boldsymbol{n}} = 0, \tag{5.12}$$

for $\zeta = -1$ and $\zeta = 1$. The resulting projections $\xi_{a/b}^1$ serve as limits of the mortar domain, parametrized by $\chi \in [-1, 1]$. They are mapped onto the segment space χ according to (3.44). To evaluate the integrals using Gauss quadrature, the mortar-side Gauss points ξ_G^1 are projected along mortar segment normal $\underline{\boldsymbol{n}}$ onto the non-mortar side and the corresponding local coordinates ζ_G are determined by solving for each Gauss point ξ_G^1:

$$\left[N_i^2(\mu^2(\zeta_G), \eta^2(\zeta_G))\underline{\boldsymbol{x}}_i^2 - N_m(\xi_G^1)\underline{\boldsymbol{x}}_m^1\right] \times \underline{\boldsymbol{n}} = 0. \tag{5.13}$$

The Gauss point location obtained in the parametric space ζ is mapped onto the underlying parameterization of the non-mortar element (μ^2, η^2).

$$\mu_G^2(\zeta_G) = \frac{1}{2}(1-\zeta_G)\mu_1^2 + \frac{1}{2}(1+\zeta_G)\mu_2^2, \quad \eta_G^2(\zeta_G) = \frac{1}{2}(1-\zeta_G)\eta_1^2 + \frac{1}{2}(1+\zeta_G)\eta_2^2, \tag{5.14}$$

where (μ_j^2, η_j^2) are the coordinates of the points of interest $\underline{\tilde{\boldsymbol{x}}}_j^2$ $j \in [1, 2]$ in the parent domain. The integrals are evaluated as:

$$D_{lm} = \int_{S^{\text{el}}} \Phi_l(\xi^1) N_m^1(\xi^1)\, d\Gamma = \sum_{G=1}^{N_G} w_G \Phi_l(\xi_G^1) N_m^1(\xi_G^1) J_{\text{seg}}(\xi_G^1) \tag{5.15}$$

$$\tilde{M}_{li} = \int_{S^{\text{el}}} \Phi_l(\xi^1) N_i^2(\mu^2(\zeta), \eta^2(\zeta))\, d\gamma = \sum_{G=1}^{N_G} w_G \Phi_l(\xi_G^1) N_i^2\left(\mu_G^2, \eta_G^2\right) J_{\text{seg}}(\xi_G^1) \tag{5.16}$$

where J_{seg} is the normalized Jacobian (3.48).

5.2 Numerical examples

In this section, we carefully selected numerical tests to demonstrate the accuracy, the robustness and the efficiency of the method. Linear elements are used in all the examples. All the examples are set-up using both boundary-fitted domains and embedded surfaces, solved within classical mortar and MorteX frameworks, respectively. In the figures displaying contour stress plots, transparency is used to visually differentiate between the "discarded" (most transparent), "blending" (semi transparent) and "standard" (opaque) elements.

5.2.1 Frictionless contact of cylinders

As demonstrated in Chapter 4 for mesh tying problems, set-up involving a host that is softer and coarser than the patch tend to exhibit spurious interface oscillations as a result of mesh-locking. In order to illustrate the mesh-locking effect within the context of contact problems, we choose the below set-ups for mortar [Fig. 5.3] and MorteX [Fig. 5.4]. The problem under consideration is the frictionless Hertzian contact between two infinite cylinders. For the MorteX set-up, the bottom cylinder surface is embedded into a host rectangular domain ω^2. The cylinders are of equal radius $R_1 = R_2 = 8$ mm. Linear elastic materials are used for both the domains ω^1 (E_1, ν_1) and ω^2 (E_2, ν_2). A material contrast is introduced by choosing $E_1/E_2 = 100$. The same Poisson ratio of $\nu_1 = \nu_2 = 0.3$ is used for both the domains. The top cylinder has a finer discretization compared to the bottom cylinder, with the mesh contrast parameter $m_c \approx 3$. A vertical displacement $u_y = 0.005$ mm is applied on the surface of the top cylinder. This results in a total reaction force of $P \approx 0.016512$ N. The bottom surface is fully fixed. The middle point of the top surface is fixed in x direction. For the classical mortar framework contact is enforced between the real-real surface pair (γ_c^1/γ_c^2), while for MorteX the contact is enforced between the real-virtual surface pair $(\gamma_c^1/\tilde{\gamma}_c^2)$.

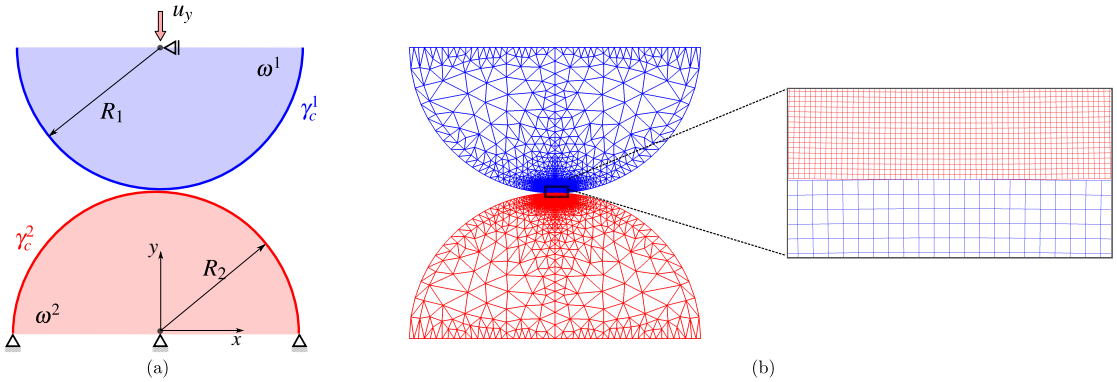


Figure 5.3: Mortar Hertzian contact: (a) problem set-up and boundary conditions; (b) FE discretization, with mesh contrast parameter $m_c \approx 3$ (zoom at the interface mesh).

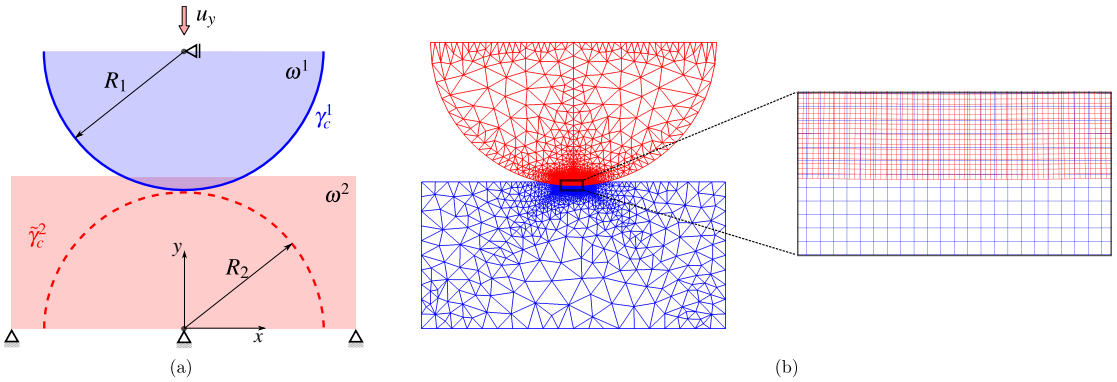


Figure 5.4: MorteX Hertzian contact: (a) problem set-up and boundary conditions; (b) FE discretization, with mesh contrast parameter $m_c \approx 3$ (zoom at the interface mesh).

The analytical solution for this problem is derived from the Hertzian contact formulae for two cylinders, which defines the maximum contact pressure (p_0), the semi contact width a and the contact pressure distribution p along the x coordinate [Johnson, 1985].

$$p_0 = \sqrt{\frac{PE^*}{\pi R^*}}, \quad (5.17) \qquad a = \sqrt{\frac{4PR^*}{\pi E^*}}, \quad (5.18) \qquad p = p_0\sqrt{1-\left(\frac{x}{a}\right)^2}. \quad (5.19)$$

In the above equations, the effective elastic modulus E^* is defined as

$$E^* = \frac{E_1 E_2}{E_1(1-\nu_2^2) + E_2(1-\nu_1^2)} \quad (5.20)$$

and the effective radius R^*, is evaluated as:

$$R^* = \frac{R_1 R_2}{R_1 + R_2}. \quad (5.21)$$

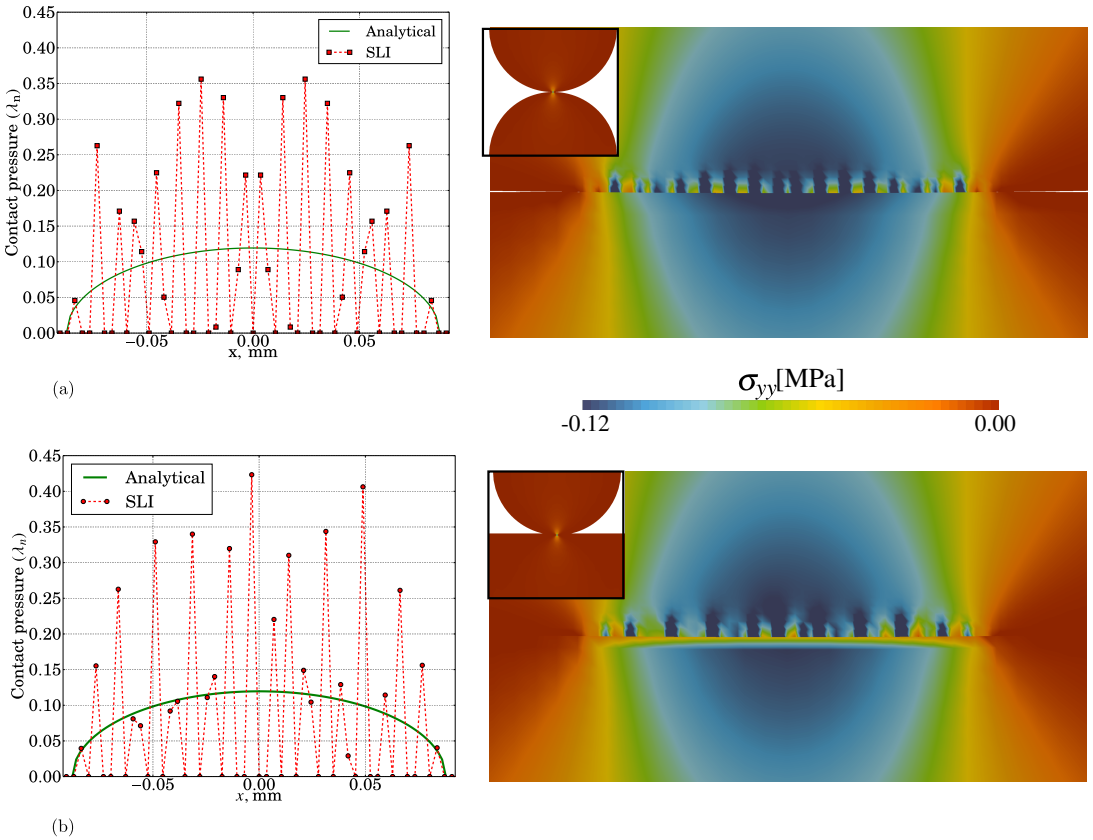


Figure 5.5: Contact stresses (λ_n, σ_{yy}) for the standard Lagrange multiplier interpolations (SLI): (a) Mortar; (b) MorteX methods.

Using the standard Lagrange multiplier interpolation in which every mortar side node holds a Lagrange multiplier results in spurious oscillations in the stresses [see Fig. 5.5]. Similar oscillations have been demonstrated in Chapter 4 for the mesh tying problems. The stabilization technique, coarse graining of Lagrange multiplier interpolations [see Section 4.3], has been proved efficient in removing these oscillations on various problem settings for mesh tying problems. We adapt the same stabilization technique for contact problems, to both the mortar and MorteX frameworks. The plots in Fig. 5.6, shows the results obtained by applying the CGI scheme for two values of spacing parameter $\kappa = 2, 3$ [see Section 4.3]. The amplitude of spurious oscillations are reduced for $\kappa = 2$ $(\kappa < m_c)$ and they are almost eliminated for $\kappa = 3$ $(\kappa = m_c)$, enabling an accurate representation of the analytical contact pressure distribution. The results corroborate the facts established by the patch and Eshelby tests [see Chapter 4], concerning the minimum value that κ needs to take for minimizing the effect of mesh-locking $(\kappa \approx m_c)$. Note that the change in vertical stress in the MorteX framework on the non-mortar side is simply a visualization issue.

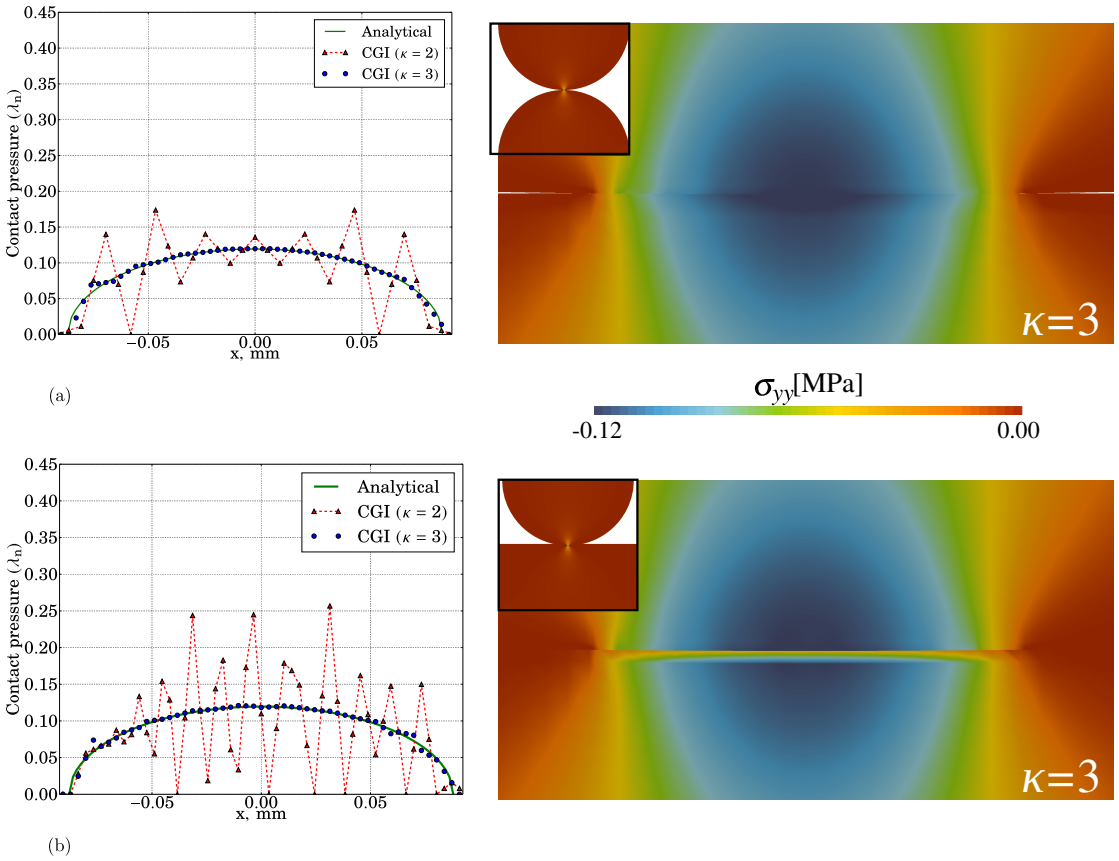


Figure 5.6: Contact stresses (λ_n, σ_{yy}) for the coarse grained Lagrange multiplier interpolations (CGI): (a) Mortar; (b) MorteX methods.

5.2.2 Frictional contact of cylinders

A set-up similar to the frictionless case [Fig. 5.4(a)] is considered for the frictional contact. The Coulomb's friction law is used with the coefficient of friction $\mu = 0.2$. The same linear elastic material is assigned to the two cylinders with $E = 200$ MPa and $\nu = 0.3$. The two cylinders are discretized ensuring equal meshes densities ($m_c \approx 1$). This test was also considered in [Yang et al., 2005, Gitterle et al., 2010]. In the first load sequence, a vertical displacement $u_y = 0.182$ mm is applied on the top surface of the upper cylinder in 100 load steps, which results in a total reaction force of $P \approx 10.0$ N ($p_0 \approx 0.625$ N/mm). This is followed by a second sequence of loading where a horizontal displacement $u_y = 0.03$ mm is applied in 100 load steps. This results in a total reaction force of $Q \approx 0.936$ N ($q_0 \approx 0.05851$ N/mm). The bottom surface of the lower cylinder is fixed throughout the simulation. The contact pressure profile $p(x)$ and the contact semi-width a are still given by Eqs. (5.19)-(5.18).

According to the analytical solution [Johnson, 1985], the contact zone is divided into a stick zone in the central area $|x| \leq c$ and two peripheral slip regions $c < |x| \leq a$, where the semi-width of the stick zone is given by:

$$c = a\sqrt{1 - \frac{q_0}{\mu p_0}}. \tag{5.22}$$

The tangential traction p_τ is given as:

$$p_\tau(x) = \begin{cases} \mu \dfrac{4Rp_0}{\pi a^2}(\sqrt{a^2 - x^2} - \sqrt{c^2 - x^2}) & , \text{if } |x| \leq c \\ \mu \dfrac{4Rp_0}{\pi a^2}(\sqrt{a^2 - x^2}) & , \text{if } c < |x| \leq a \\ 0 & , \text{elsewhere.} \end{cases} \tag{5.23}$$

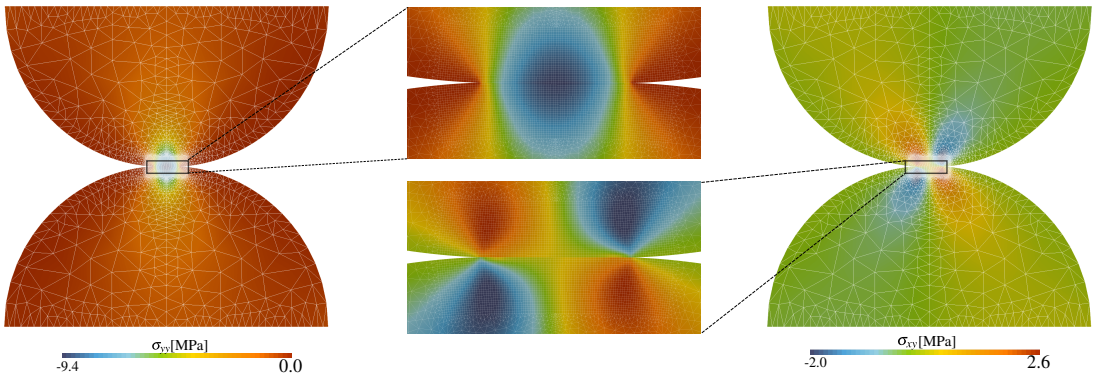


Figure 5.7: Contour stress plots σ_{yy}, σ_{xy} for results obtained by Mortar method.

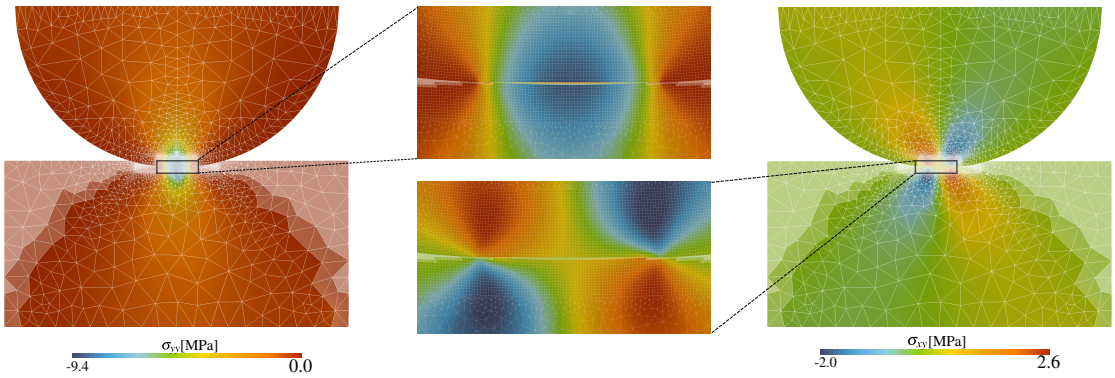


Figure 5.8: Contour stress plots σ_{yy}, σ_{xy} for results obtained by MorteX method.

Fig. 5.7 and 5.8 show the contour stress plots for σ_{yy} and σ_{xy} at the end of the second load sequence, obtained with Mortar and MorteX methods, respectively. The results obtained by the Mortar and MorteX methods are very similar to each other, and provide a descent approximation of the analytical solution in terms of contact tractions [see Fig. 5.9].

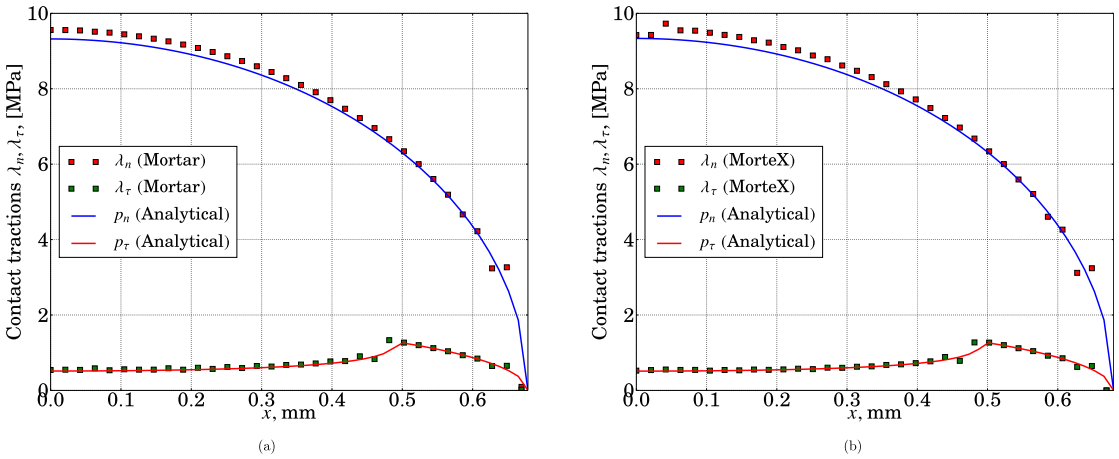


Figure 5.9: Normal and tangential contact tractions: (a) Mortar; (b) MorteX methods.

5.2.3 Ironing a wavy surface

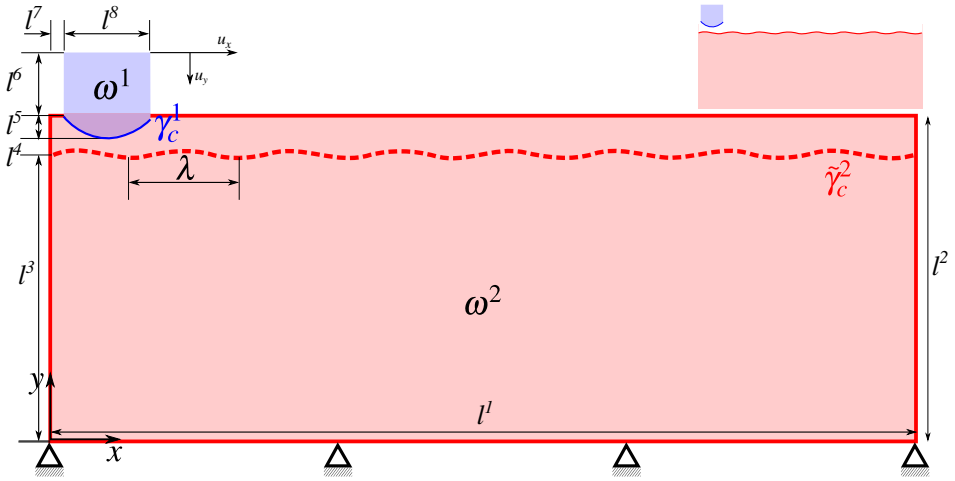


Figure 5.10: MorteX frictionless ironing of wavy surface set-up, equivalent Mortar set-up (inset).

In this example we consider a frictionless sliding contact between an elastic slider and an elastic wavy substrate [see Fig. 5.10]. Within the MorteX framework, a discretized surface γ_c^1 comes in contact and slides along a virtual surface $\tilde{\gamma}_c^2$ embedded inside a rectangular block. The slider and substrate are meshed with a comparable mesh density ($m_c \approx 1$). The geometric dimensions used are: $l^1 = 12$, $l^2 = 4.5$, $l^3 = 4$, $l^4 = 0.2$, $l^5 = 0.3$, $l^6 = 0.9$, $l^7 = 0.2$, $l^8 = 1.2$ (all the length dimension are in mm). The wavy surface is described by $y(x) = \Delta(y/l^3)\sin(2\pi x/\lambda)$, with $\lambda = 1.5$ mm and $\Delta = 0.05$ mm. Both solids are made of the same material: Young's modulus $E = 100$ kPa and Poisson's ratio $\nu = 0.3$. A vertical displacement of $u_y = -0.75$ mm is applied on the top of the slider within first 20 load steps ($t \in [0, 1]$) while the horizontal displacement of the same boundary is kept zero. During the following sequence ($t \in [1, 2]$), the vertical displacement is maintained and the horizontal displacement of $u_x = 10$ mm is applied in 100 load steps. The bottom surface of the rectangular block is fixed throughout the simulation.

For the sake of comparison, the same problem is also solved within the classical mortar contact formulation, in which the contact occurs between two surfaces explicitly represented by body-fitted meshes [see Fig. 5.10, inset]. The contour plots of stresses σ_{yy}, at $t = 1.0, 2.0$ seconds, for both the Mortar and MorteX methods are shown in Fig. 5.11-5.12. These fields are very smooth and indistinguishable by naked eye. The normalized contact tractions (λ_n/E) and the displacements in y-direction (u_y) along the real surface γ_c^1 are compared between the two methods. In order to quantify the difference, we introduce the L^2 norm of the error in the displacements and contact tractions as below:

$$\underline{\boldsymbol{u}}_{y(L^2)} = \frac{||\underline{\boldsymbol{u}}_y^{\text{Mortar}} - \underline{\boldsymbol{u}}_y^{\text{MorteX}}||_{L^2(\gamma_c^1)}}{||\underline{\boldsymbol{u}}_y^{\text{Mortar}}||_{L^2(\gamma_c^1)}}, \qquad \underline{\boldsymbol{\lambda}}_{n(L^2)} = \frac{||\underline{\boldsymbol{\lambda}}_n^{\text{Mortar}} - \underline{\boldsymbol{\lambda}}_n^{\text{MorteX}}||_{L^2(\gamma_c^1)}}{||\underline{\boldsymbol{\lambda}}_n^{\text{Mortar}}||_{L^2(\gamma_c^1)}}. \tag{5.24, 5.25}$$

where the norm means $\|f(x) - g(x)\|_{L^2(\gamma_g^1)} = \sqrt{\sum_i [f(x_i) - g(x_i)]^2}$, where $x_i \in [0, L]$ are the x-coordinate of mortar nodes, and L is the length of the surface γ_g^1. The displacement and traction error norms [see Fig. 5.13], reflects the accuracy and robustness of the MorteX framework.

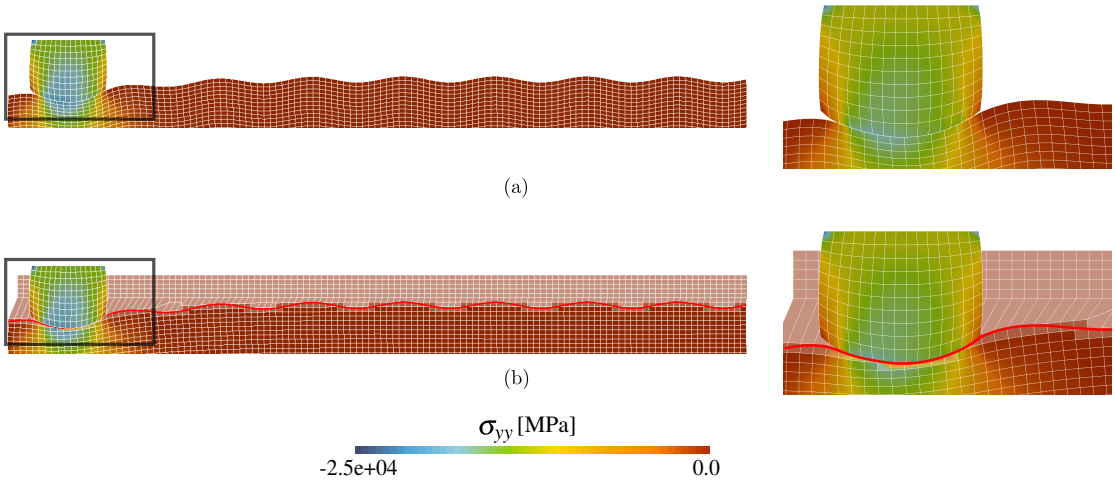


Figure 5.11: σ_{yy} contour plots at $t = 1$ seconds (a) Mortar; (b) MorteX methods.

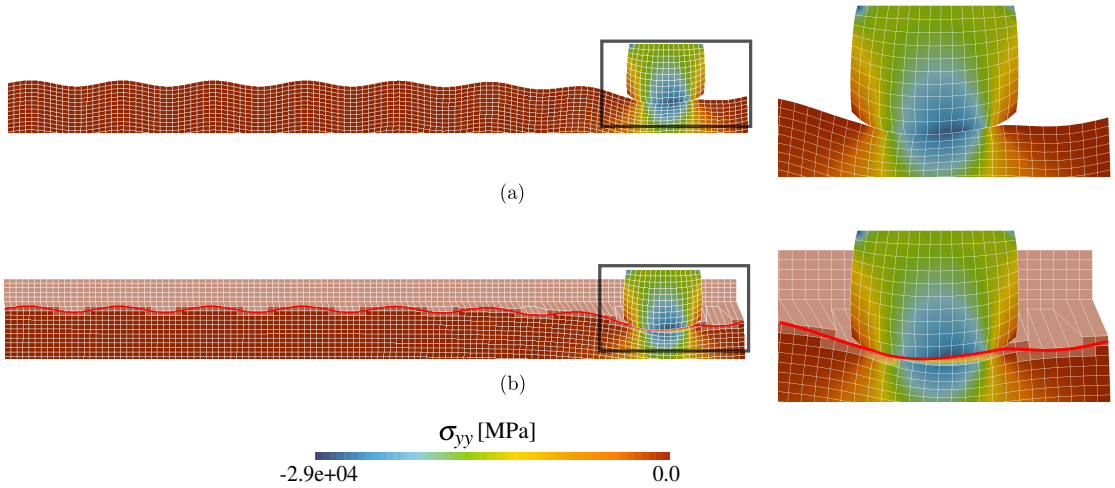


Figure 5.12: σ_{yy} contour plots at $t = 2$ seconds (a) Mortar; (b) MorteX methods.

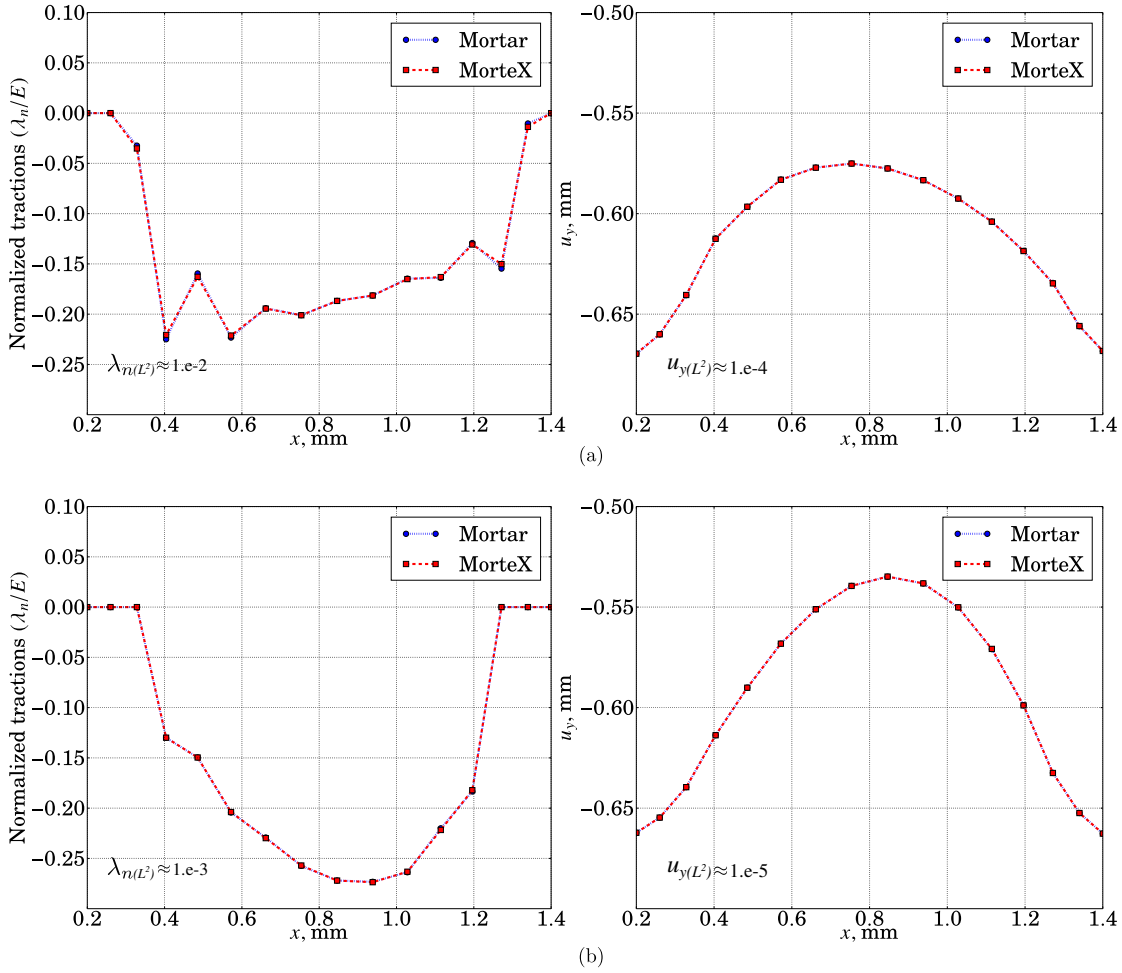


Figure 5.13: Comparison of normalized contact tractions (λ/E) and displacements (u_y) along slider surface γ_c^1: (a) t=1.0; (b) t=2.0 seconds.

5.2.4 Frictional shallow ironing

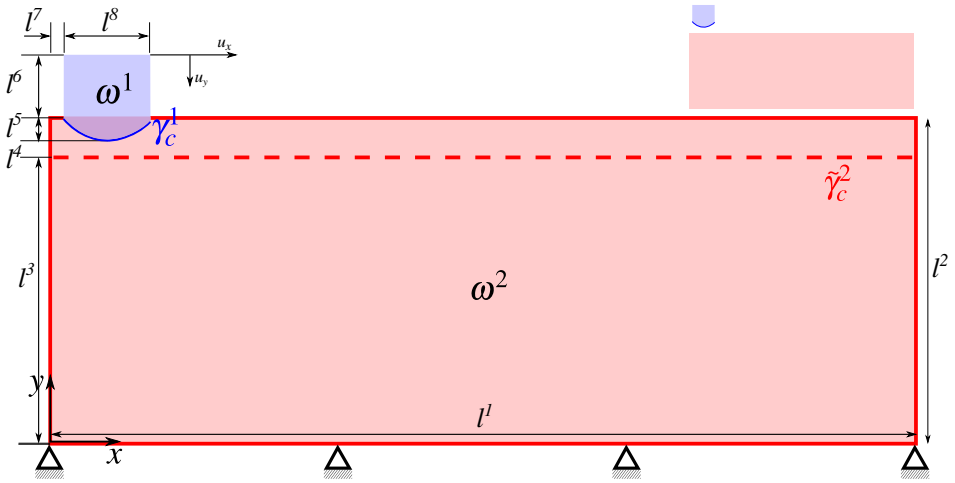


Figure 5.14: MorteX frictional ironing set-up, equivalent Mortar set-up (inset).

Here we consider the same geometrical set-up as in the previous example (Section 5.2.3), but here the virtual surface $\tilde{\gamma}_c^2$ embedded into the host domain ω^2 is flat [see Fig. 5.14]. The slider and substrate properties are respectively $E_1 = 68.96$ MPa, $\nu_1 = 0.32$ and $E_2 = 6.896$ MPa, $\nu_1 = 0.32$ ($E_1/E_2 = 10$). In addition, the slider has a finer mesh than the substrate: so that the mesh contrast is $m_c \approx 3$. A coefficient of friction $\mu = 0.3$ is used. The contrast in material and mesh density is introduced purposefully to better illustrate the manifestation of the mesh locking phenomenon. A vertical displacement of $u_y = -0.75$ mm is applied on the top of the slider within first 50 load steps ($t \in [0, 1]$), while the horizontal displacement of the same boundary is kept zero. During the following sequence ($t \in [1, 2]$), the vertical displacement is maintained and the horizontal displacement of $u_x = 10$ mm is applied in 500 load steps. The bottom surface of the rectangular block is fixed in all directions throughout the simulation.

Fig. 5.15, shows the oscillation in the stress field σ_{yy} for the standard Lagrange multiplier interpolations, both in the context of Mortar and MorteX. These results suffer from spurious oscillations. Similar to the Hertzian contact problem setting in Section 5.2.1, we use the coarse grained Lagrange multiplier interpolations to both the Mortar and MorteX formulations. It results in a reduced amplitude of oscillations, as seen in Fig. 5.16, and 5.17 where the contact tractions are presented for different coarse grain spacing parameter κ. A good agreement between the Lagrange multiplier field λ_n and hence the reaction forces (R_x, R_y) on the slider for the Mortar and MorteX methods can bee seen in Fig. 5.18. In this example, the applicability of CGI for Lagrange multipliers is once again proved efficient for both the MorteX and for the classical Mortar frameworks.

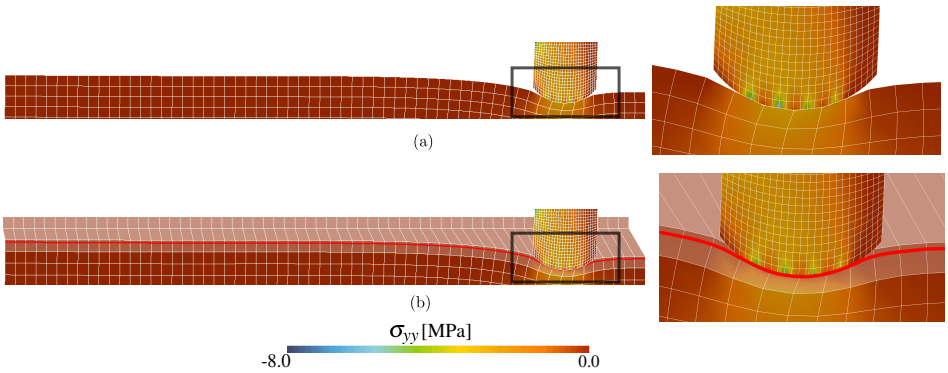


Figure 5.15: Standard Lagrange multiplier space ($\kappa = 1$), σ_{yy} contour plots at $t = 2$ seconds (a) Mortar; (b) MorteX methods.

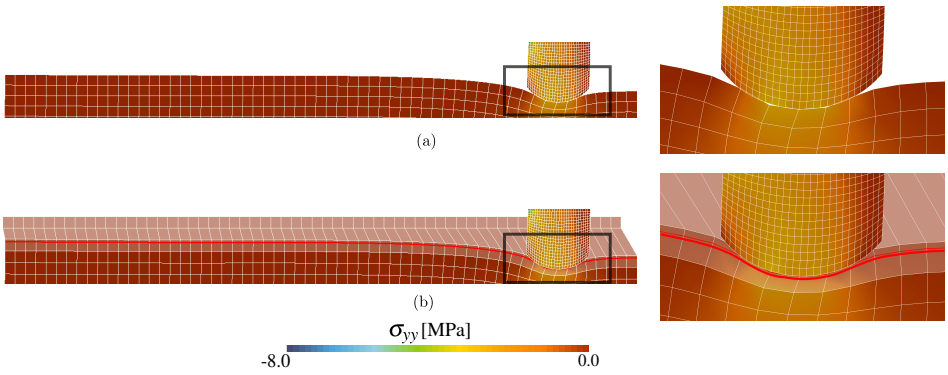


Figure 5.16: Coarse grained Lagrange multiplier space ($\kappa = 3$), σ_{yy} contour plots at $t = 2$ seconds (a) Mortar; (b) MorteX methods.

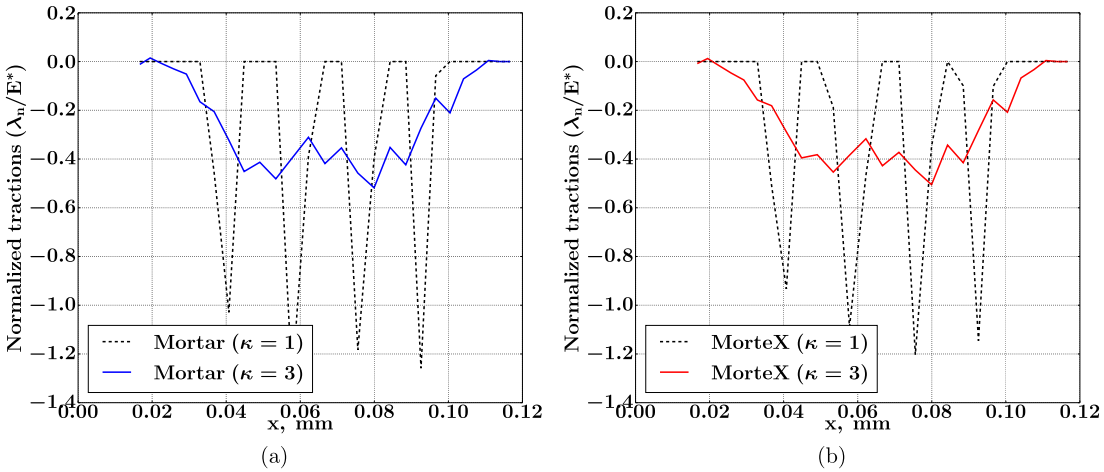


Figure 5.17: Comparison of λ_n at nodes along γ_c^1 for $\kappa = 1$ and $\kappa = 3$: (a) Mortar; (b) MorteX methods.

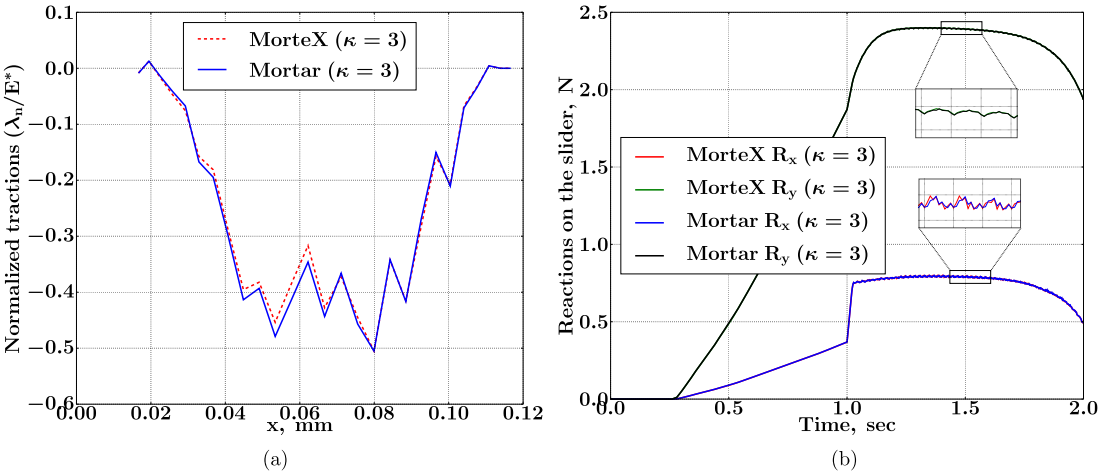


Figure 5.18: Comparison of fields between Mortar and MorteX methods for $\kappa = 3$: (a) contact tractions λ_n along γ_c^1; (b) reaction forces in x and y along the top surface of the slider.

5.3 Simulation of wear using MorteX method

In the numerical examples presented in the previous section the virtual surface was embedded in the host mesh once in the beginning of the simulation. However, this needs not always be the case, i.e. the embedded surface can evolve during the simulations either from one time step to another or even within the iterative procedure at every iteration, like for e.g. in wear problems. Wear is a complex interfacial phenomenon resulting from relative motion between contacting bodies, and manifests itself in material removal. This phenomenon is observed specifically in mechanical assemblies such as the rivets, bearing and gears, turbine blade fixings, etc. Accurate representation of contact tractions while accounting for the continuously evolving geometrical changes due to wear, is challenging. The most commonly used wear criterion is the one proposed by Archard [Archard, 1953]. It is formulated at the scale of entire contact interface (global/macro scale) as:

$$V = K\frac{P}{H}\delta, \tag{5.26}$$

where V is the total worn out volume at the interface, K is a dimensionless wear coefficient, P is the normal load, H is the hardness of the material and δ is the accumulated sliding distance. In [Fouvry et al., 1996], the authors proposed a rather similar wear law which links the worn volume with the dissipated energy. The Archard's law and the energetic laws for wear are equivalent under the assumption of constant and uniform coefficient of friction. Alternatively, these macroscopic laws can be assumed valid at the micro/local scale, which is essential for the numerical treatment of the wear problem. Hence, the worn out volume (5.26) can be represented in terms of the local wear depth increment dh for a local slip increment $d\delta$ and the local contact pressure $p(x)$:

$$dh = k\mu p(x)d\delta = kdW_d, \tag{5.27}$$

where μ is the coefficient of friction, k is local wear coefficient with units (1/Pa), and $dW_d = \mu p(x)d\delta$ is an increment of the surface density of the dissipated energy.

The localized nature of this approach allows local update of the worn geometry. Such localized Archard's wear law was first implemented within the FEM framework in [Johansson, 1994]. Subsequent works were based on an iterative simulation of wear surface evolution using updated geometry [Podra and Andersson, 1999, Öqvist, 2001, Kim et al., 2005, Lengiewicz and Stupkiewicz, 2013, Farah et al., 2016, Farah et al., 2017]. However the numerical complexities involved in these classical approaches are considerable: remeshing procedures to capture the geometrical changes and field remapping of history variables.

Here, the MorteX method is intended to mitigate the numerical complexities accompanied with the remeshing techniques only. This requires imposing the contact constraints between real and virtual surfaces when the latter evolves between the load steps. A local "dissipated energy" wear criterion (5.27) is used. The process of wear profile evolution and its incorporation into the underlying geometry will be detailed. The methodology will be demonstrated on a classical fretting-wear set-up (cylinder on flat plate) for the gross-slip regime.

5.3.1 Problem set-up

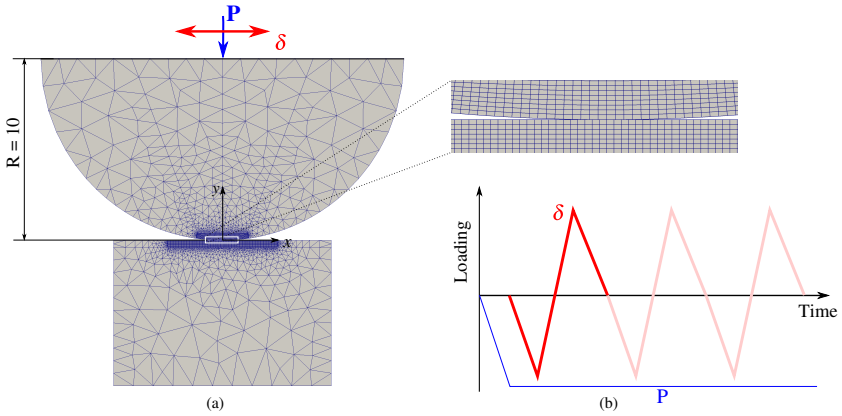


Figure 5.19: Wear problem setup: (a) FE mesh; (b) Boundary conditions, the normal load P is applied and maintained throughout the simulation. The δ sawtooth horizontal displacement is applied for many fretting cycles.

We consider a cylinder-plate fretting wear example. The modeling framework assumes that only the plate wears out as a result of frictional contact. This assumption is made in order to restrict the problem setting to the bounds of the MorteX framework, which is capable of treating contact between a real and virtual surface. A linear elastic material model under plane strain assumption is considered here with Young's modulus $E = 119$ GPa and Poisson's ratio $\nu = 0.29$. This choice confines the modeling to the problematic of material removal or the evolution of virtual interface due to wear and does not involve extra difficulties from variables transfer. A coefficient of friction $\mu = 0.8$ is considered between the contacting surfaces, and the Coulomb's friction law is used. The geometric dimensions are: radius $R = 10$ mm for the half cylinder and 12 mm$\times$8 mm along x and y directions respectively for the rectangular plate. A finer mesh discretization with quadrilateral element is chosen around the contact zone [Figure 5.19(a)].

The boundary conditions are applied in two sequences. In the first sequence a a normal load of $P = 300$ N/mm is imposed on the central point of the top surface of the cylinder. Equality of displacements u_y for the nodes of the top surface is ensured with multi-point constraints. This is followed by a second sequence, when a periodic saw-tooth horizontal displacements of amplitude δ is applied on the top of the cylinder. A fretting load cycle is constituted of one complete horizontal displacement cycle as illustrated in Fig. 5.19(c). The sides and bottom of the plate are kept fixed through out the simulation. The choice of $\delta = 100\,\mu$m is made to reproduce the gross sliding conditions.

5.3.2 Methodology of MorteX wear-modeling

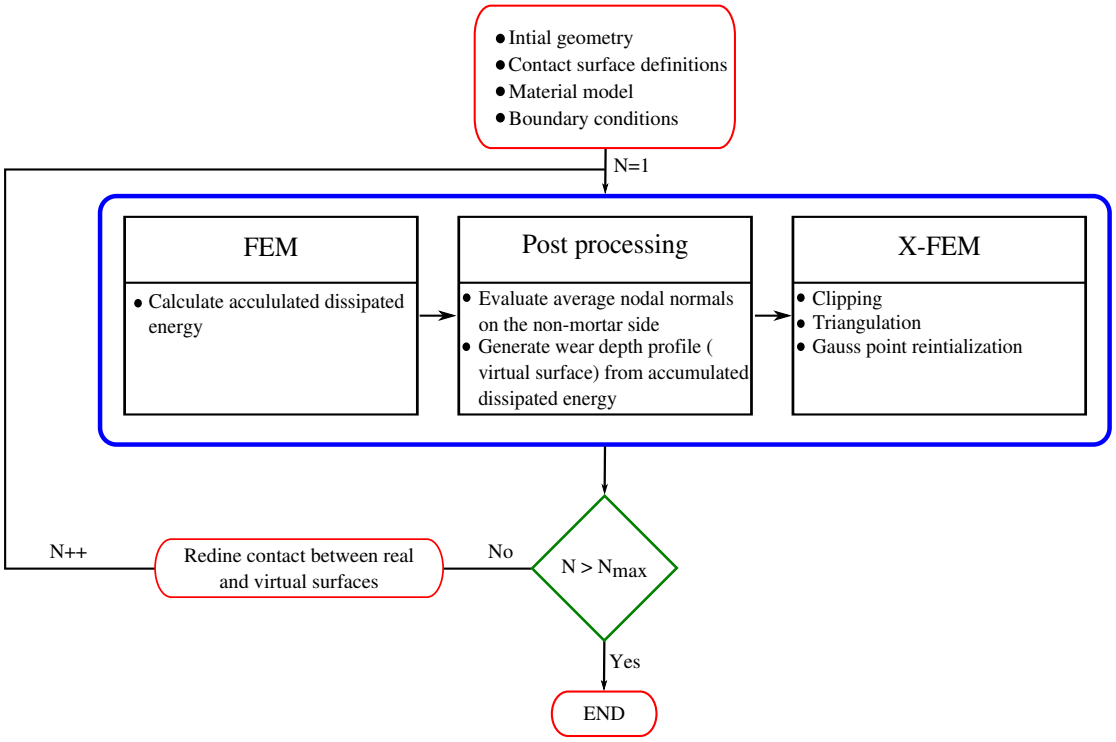


Figure 5.20: The flowchart illustrating the incremental wear modeling within the MorteX framework.

The main steps involved in the numerical wear modeling within the framework of MorteX are shown in Fig. 5.20. The accumulated dissipated energy during a load sequence is very slow, i.e. the evolution of the wear profile is practically negligible. To accelerate the simulations, we use the concept of the effective fretting cycle, where each numerical cycle represents N real cycles. Each effective fretting cycle is divided into N^* load steps. During the first effective fretting cycle, contact is imposed between the boundary-fitted mortar and non-mortar surfaces. Therefore the classical mortar methods are used for resolving the contact problem. This implies the use of standard discrete kinematic quantities (3.18), (3.27). After the end of a load step increment, the linear density of the dissipated energy ΔW_d^{incr} of the mortar element in sliding state is evaluated as an integral of the product between the tangential contact tractions $\lambda_\tau^l\,(= \mu\lambda_n^l)$ and the incremental slip $\overset{\circ}{\bar{g}}{}_\tau^l$

over the mortar domain S^{el} as:

$$\Delta W_d^{\text{incr}} = \int_{S^{\text{el}}} \lambda_\tau^l \overset{\circ}{g}_\tau^l \, d\gamma = \int_{S^{\text{el}}} \Delta w_d \, d\gamma, \tag{5.28}$$

The Gauss quadrature for the above integral follows as:

$$\Delta W_d^{\text{incr}} = \sum_{G=1}^{N_G} w_G N_m(\xi_G^1) \Delta w_d^m J_{\text{seg}}(\xi_G^1), \tag{5.29}$$

where as previously $m \in [1, M]$ and $G \in [1, N_G]$ (N_G is number of Gauss integration points), J_{seg} is the normalized Jacobian. After the completion of an effective fretting cycle, the accumulated dissipated energy of the mortar element is summed over all the load steps that constitute the cycle.

$$\Delta W_d = \sum_{\text{incr}=1}^{N^*} \Delta W_d^{\text{incr}} \tag{5.30}$$

The accumulated dissipated energy is then divided equally among the non-mortar nodes of the contact element. In the case of linear meshes, each non-mortar node i of the contact element is assigned with $\frac{\Delta W_d}{2}$. This manner of assigning is chosen to simplify the implementation, and a split of the energy between the non-mortar nodes can be based on geometrical considerations. Finally, the total dissipated energy for every non-mortar node is the summation of the energies from all the mortar elements sharing it. The nodal dissipated energy ΔW_d^i is converted into an equivalent wear depth $\Delta \underline{\boldsymbol{h}}^i$ of the node as [Fig. 5.21(a)]:

$$\Delta \underline{\boldsymbol{h}}^i = -k\beta(N) \Delta W_d^i \underline{\boldsymbol{n}}^{i*}, \tag{5.31}$$

where $\underline{\boldsymbol{n}}^{i*}$ is the average nodal normal of the non-mortar node i and k is the local wear coefficient calibrated by integrating the model on the whole wear scar and by adjusting the result to experimental data [McColl et al., 2004]. An acceleration factor $\beta(N, \Delta h(t))$ is, in general, a function of the number of real cycles in the effective cycle and also of the wear history. In the simplest case, it can be assumed to be equal to the number of real cycles $\beta = N$. Note, that the average nodal normal $\underline{\boldsymbol{n}}^{i*}$ is computed for the non-mortar side of the interface only for the virtual surface definition. For the contact related integral evaluations, the non-averaged mortar segment normal $\underline{\boldsymbol{n}}$ is used. The positions of the virtual nodes is obtained by adding the nodal wear depth $\Delta \underline{\boldsymbol{h}}^i$ (5.31) to the positions of the corresponding non-mortar node i. A new virtual 1D surface $\tilde{\gamma}_c^2$, is formed by joining the virtual points of the non-mortar side nodes. The limits of the each virtual non-mortar segment are the "points of interest" [see Fig. 5.21(b)]. The volume lying outside the virtual surface is considered as the worn out volume. Accounting for the worn out volume within the MorteX framework involves clipping, triangulation and Gauss point reinitializations as a part of the selective integration scheme. The worn out volume is excluded from the weak form integration.

Starting from the second effective fretting cycle onwards, the frictional contact is considered between the unworn real mortar side surface and the worn out geometry represented by a virtual surface $\tilde{\gamma}_c^2$ [Fig. 5.21(b)]. Unlike the three non-mortar segments S^1, S^2, S^3 in the first fretting cycle, the virtual non-mortar segments

defined by the non-mortar nodes and the points of interest $(S^1, S^2, S^3, S^4, S^5)$ increase in number at every update [see Fig. 5.22(a)]. For subsequent cycles, this increase can become unbounded. To limit this, the points of interest lying close to each other are combined into a single point. The dissipated energy for a MorteX contact element $(\Delta\tilde{W}_d^{\text{incr}})$ is now evaluated using the modified integral incremental slip $\tilde{\tilde{g}}_\tau^l$ (5.6). Then, Eq. (5.29) changes as follows:

$$\Delta\tilde{W}_d^{\text{incr}} = \sum_{G=1}^{N_G} w_G N_m(\xi_G^1)\Delta\tilde{w}_d^m J_{\text{seg}}(\xi_G^1). \tag{5.32}$$

The accumulated dissipated energy over the effective fretting cycle is calculated similar to (5.30), using $\Delta\tilde{W}_d^{\text{incr}}$. The dissipated energy density $\Delta\tilde{W}_d$ is split between the limits of the non-mortar segment namely: non-mortar nodes or points of interest. The non-mortar segment limits are translated along the averaged normal defined at the limits of the virtual non-mortar segment $\underline{\tilde{\boldsymbol{n}}}^{i*}$ (5.33). A new virtual surface is formed by joining the translated positions [see Fig. 5.22(b)].

$$\Delta\underline{\tilde{\boldsymbol{h}}}^i = -k\beta(N)\Delta\tilde{W}_d^i\underline{\boldsymbol{n}}^{i*}. \tag{5.33}$$

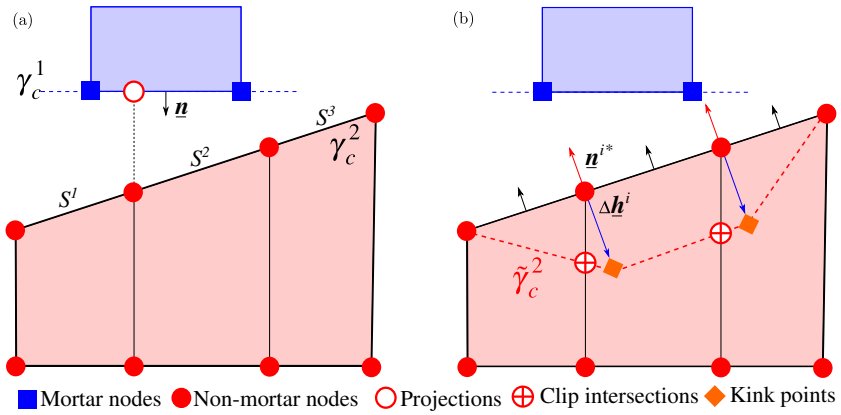


Figure 5.21: First effective fretting cycle: (a) The mortar segmentation process; involving projections of the non-mortar side nodes along mortar segment normal $\underline{\boldsymbol{n}}$; (b) worn out surface formed by joining translated (by $\Delta\underline{\boldsymbol{h}}$) non-mortar segment nodes.

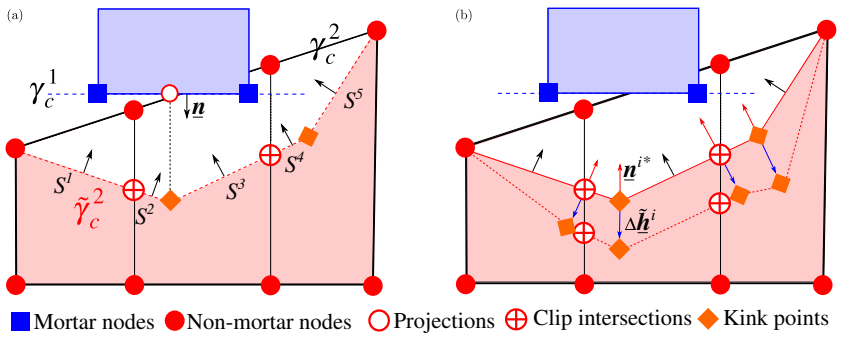


Figure 5.22: Second effective fretting cycle: (a) The mortar segmentation process; involving projections of the non-mortar nodes/point of interest along mortar segment normal $\underline{\boldsymbol{n}}$; (b) new worn out surface formed by joining translated (by $\Delta\underline{\tilde{\boldsymbol{h}}}$) point of interest.

5.3.3 Wear simulation

Here we summarize the steps involved in simulation of material removal process for the introduced wear problem set-up [see Fig. 5.19]. The wear simulation parameters used are: $k = -5 \cdot 10^{-8}$ Pa^{-1}, $\beta = 100$. The three steps are:

1. Evaluation of dissipated energy (ΔW_d) during a effective fretting cycle along the entire contact interface [see Fig.5.23(a)].

2. Conversion of ΔW_d into an equivalent wear depth profile [see Fig.5.23(b)].

3. Superposition of the depth profile onto the current plate mesh, followed by clipping and triangulation [see Fig.5.24(a), (b)].

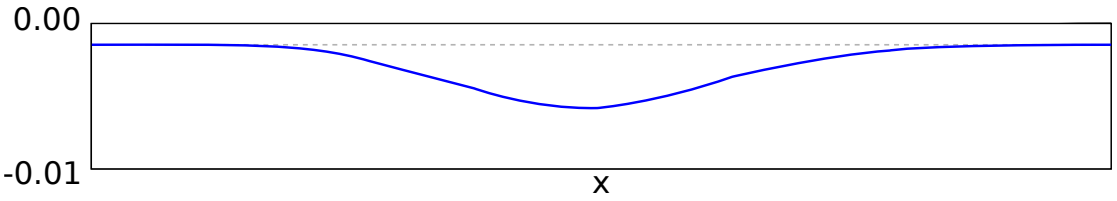


Figure 5.23: Wear simulation: Equivalent wear depth profile.

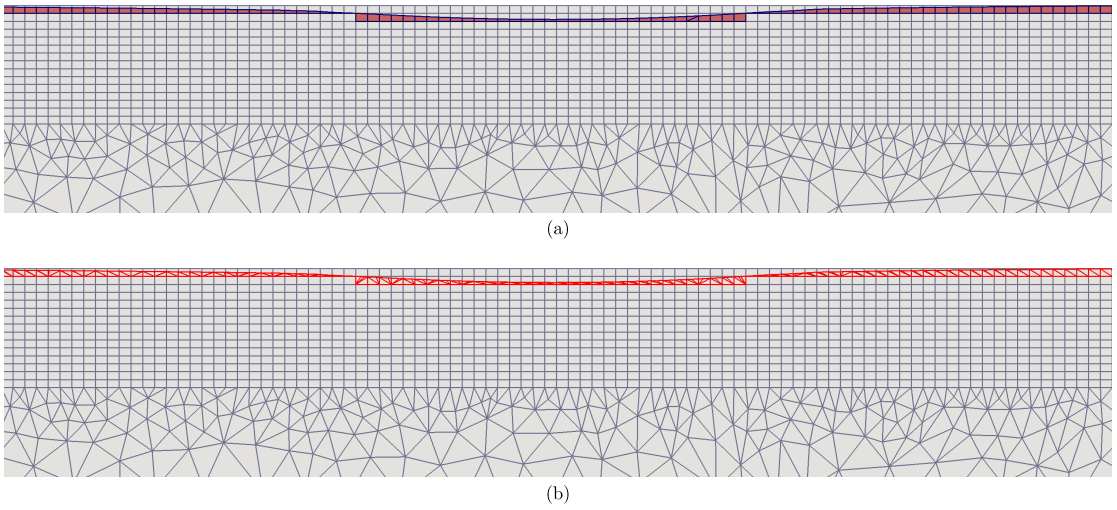


Figure 5.24: Wear simulation: (a) Clipping of the underlying mesh with the virtual surface; (b) Triangulation of the clipped polygons for selective integration.

The mesh configurations at three different stages of the cyclic loading are shown in Fig. 5.25. The material removal as a result of frictional contact (wear) is clearly seen as the top cylinder penetrates into the volume of plate. The material removal is mimicked by the evolving virtual surface, where contact is redefined between the newly formed virtual surface after each effective fretting cycle. The contact stress σ_{yy} redistribution in response to the evolving contact interface can be seen in Fig. 5.26.

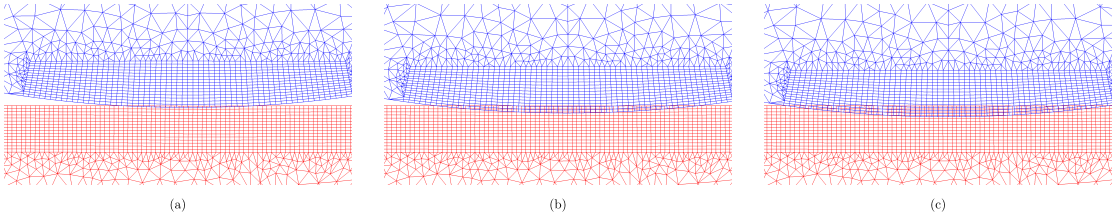


Figure 5.25: Mesh configuration after: (a) 300; (b) 1500; (c) 2500 fretting cycles.

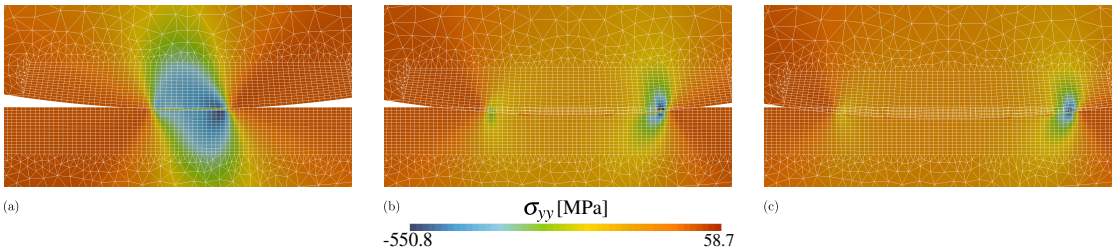


Figure 5.26: Contour stress (σ_{yy}) after: (a) 300; (b) 1500; (c) 2500 fretting cycles.

5.4 Summary

We extended the MorteX framework to treat contact problems along real-embedded interfaces. The algorithmic differences compared to the tying problem in terms of integral evaluation which now involves extra projections are discussed. The applicability of the coarse graining strategy for the contact problems is demonstrated with numerical examples both for the mortar and MorteX methods has been demonstrated with numerical examples. The dependency of the coarse grain Lagrange multiplier strategy on the parameter κ was demonstrated with the frictionless Hertz problem. The minimum value that κ needs to take is approximately m_c, the mesh contrast parameter. This corroborates the findings from the mesh tying problem settings considered in Chapter 4.

A prototype wear simulation, between a half cylinder and a flat plate in gross slip regime was solved within the MorteX framework. Within the MorteX framework the evolution of surface as a result of material removal due to wear is modeled. This circumvents the need for conventional remeshing procedures to account for material removal.

Chapter 6

Conclusion and perspectives

6.1 Conclusions

Résumé: Cette thèse a pour but de développer un ensemble de méthodes permettant de gérer les problèmes de contact et de couplage de maillages dans le cadre de la méthode des éléments finis classiques et étendus. Dans ce chapitre, les contributions originales de la thèse ainsi que les perspectives pour les travaux futurs sont présentés.

In this thesis, the focus is laid on development of a robust three-dimensional numerical framework to treat contact problems. For this purpose, we coupled the accurate mortar discretization scheme with the monolithic augmented Lagrangian resolution strategy. A fully consistent linearization of the contact residuals is derived and implemented for the first time within this setting (to the best of our knowledge). Various aspects of the numerical treatment of contact are discussed: detection, discretization, accurate evaluation of mortar integrals (projections, clipping, triangulation), and the parallelization on distributed memory architectures. The method's performance under various problem settings involving material and geometric non-linearities was demonstrated. Few algorithmic nuances arising in the presented framework, such as an iterative update of contact interface elements and the importance of slip history storage for frictional problems are discussed and illustrated under simple problem settings. Slip storage enables a more accurate initiation of contact statuses in the beginning of the following load step and thus accelerates the convergence of the Newton method. The classical formulation of the mortar method for mesh tying applications is also recalled and demonstrated for two and three-dimensional set-ups and is also used to impose periodic boundary conditions.

The main novelty of this thesis is the development of a new two-dimensional MorteX framework, which combines features of the extended finite element method (X-FEM) and the classical mortar methods. This framework enables handling such interface problems as mesh tying between overlapping domains as well as frictional contact between embedded (virtual) boundaries. The inherent stability issues resulting from the Lagrange multiplier based mixed formulations, and their manifestations in the form of spurious oscillations are illustrated under various problem settings. These settings also include two patch tests (bending and compression), which were studied in detail for different mesh density contrasts, interpolations and mesh types. Two stabilization techniques, namely automatic

triangulation of blending elements and coarse-grained Lagrange multiplier interpolation are proposed to overcome these adverse effects. The novel contribution of coarse graining Lagrange multipliers enables to efficiently avoid overconstraining in the interface and obtain smooth and oscillation-free stress fields for arbitrary mesh and material contrast. The optimal choice of coarse graining parameter, which determines the number of master mortar nodes storing Lagrange multipliers, is fully determined by the local relevant mesh density of overlapping or coming in contact domains. Therefore, this technique can be used with a greater ease compared to the Nitsche method [Sanders et al., 2012], which requires for the stabilization a prior knowledge of material contrast. This coarse-graining technique is successfully validated and tested on numerous problem settings. Mesh convergence tests is also carried our on Eshelby inclusion problem. The MorteX method is further extended to treat contact problems along embedded boundaries. Numerical examples demonstrate its flexibility in integrating complex geometries within a simple mesh and its performance in context of contact problems. Every frictionless and frictional contact problem is solved both using MorteX and mortar method, for which a conformal mesh of comparable density was used. The results demonstrate that the MorteX method ensures the same accuracy as the classical mortar method. In addition, the coarse graining of Lagrange multipliers is demonstrated to be useful for both the MorteX and classical mortar methods in the context of contact problems involving elevated material and mesh contrasts.

With the inherent ability of the MorteX method to handle contact along virtual boundaries, this framework is particularly suitable to handle wear problems with involving in time worn surfaces. Therefore, an algorithmic scheme enabling accurate update of the worn geometry using a local energy based wear law is presented. Related issues for the surface update are discussed and a fretting wear problem for gross slip regime is presented. The use of MorteX method circumvents the need for complex remeshing techniques to account for contact surface evolution.

6.2 Perspectives

The novel methods and regularization techniques developed in this thesis are promising and are worth further exploration in the future work. In particular, the following aspects of the novel and classical methods are of interest. The augmentation parameters ε_n and ε_τ needed in the monolithic augmented Lagrangian scheme, theoretically do not effect the accuracy of the converged solution. However, their choice could potentially influence the convergence of the iterative scheme: a poor choice could in some cases lead to the lack of convergence. In this regard, we used an original automatic way of choosing the augmentation parameters based on the effective local stiffness of mortar/non-mortar elements forming the contact element. However, an extensive testing of this method is yet missing. The situation is similar with a novel technique used to improve the convergence of contact problems involving strong material non-linearities. We used an original way of fixing material non-linearity in the first iterations of the Newton solver to enable an approximate convergence of the contact statuses, after which both non-linearities are “switched on” together. In some problems this approach was shown beneficial

but a consistent testing is also missing. The contact formulation is implemented only for the first interpolation order in 3D; in 2D it is implemented for both linear and quadratic interpolation functions. Therefore, a further generalization following the classical scheme elaborated in [Puso et al., 2008]. Finally, a smoothing technique for the normal field as the one used in [Popp, 2012] would be desired to render the contact treatment even more robust.

The MorteX framework is presented in two dimensions only. The results obtained in terms of having a stable, accurate and flexible formulation are encouraging both for tying and contact. A natural course would be to extend this framework to three dimensions. This however, poses challenges on the technical front. For example, finding an optimal coarse graining parameter κ in case of 3D is not as trivial as in 2D but is still feasible. In addition, the complexity of clipping (non-convex polygons) algorithms for selective integration increases. Another possible extension is handling higher order interpolations of the underlying meshes. Finally, a more flexible definition of virtual surfaces which can be embedded in the non-mortar side mesh is required, for example NURBS/spline curve/surface can be used to define the contact surface. Currently, the definition of such surfaces is limited to a piece-wise linear line.

The wear problem solved within MorteX framework serves a prototype for further investigations. Currently, only a linear elastic material was considered for the wear problem, which simplifies to a great extend the numerical treatment. This confines the problematic to only handling surface evolution as a result of material removal. However, non-linear materials, common in most industrial applications, would require field transfer procedures, which have to be integrated in the update of the surface geometry within the MorteX framework. Compared to the conventional field transfer procedures where the transfer happens between two completely different meshes (both in terms of nodal positions and element size), the remapping in MorteX is confined to the blending elements only: for each blending element, history variables need only be transferred from the older integration points to the newer ones resulting from the triangulation of the effective volume. Currently, only one of the bodies is subject to wear as a result of frictional contact. Considering wear on both sides requires the extension of the concept of embedded surfaces to both the bodies, thereby imposing contact constraints between a virtual-virtual pair compared to the current state where only real-virtual pair can be treated. A virtual-virtual contact pair would imply that the Lagrange multipliers can no longer be directly associated with reactions on nodes, which places the problem in a more stringent setting in terms of available spaces for Lagrange multipliers that satisfy the inf-sup conditions.

Bibliography

[Abel et al., 1991] Abel, J., Aubert, B., and Hsieh, S. (1991). 'Towards parallel solutions of nonlinear structural dynamics in a network workstation environment. In Proceedings of International Conference on Computational Engineering Science, pages 1–6.

[Akula et al., 2019a] Akula, B. R., Vignollet, J., and Yastrebov, V. A. (2019a). MorteX method for contact along real and embedded surfaces: coupling X-FEM with the mortar method.

[Akula et al., 2019b] Akula, B. R., Vignollet, J., and Yastrebov, V. A. (2019b). Stabilized MorteX method for mesh tying along embedded interfaces.

[Alart, 1988] Alart, P. (1988). Multiplicateurs "augmentés" et méthode de Newton généralisée pour contact avec frottement. Report, Document LMA-DME-EPFL, Lausanne.

[Alart, 1997] Alart, P. (1997). Méthode de Newton généralisée en mécanique du contact. Journal de Mathématiques Pures et Appliquées, 76(1):83–108.

[Alart and Curnier, 1991] Alart, P. and Curnier, A. (1991). A mixed formulation for frictional contact problems prone to Newton like solution methods. Computer Methods in Applied Mechanics and Engineering, 92(3):353 – 375.

[Amestoy et al., 2000] Amestoy, P. R., Duff, I. S., and L'Excellent, J.-Y. (2000). Multifrontal parallel distributed symmetric and unsymmetric solvers. Computer Methods in Applied Mechanics and Engineering, 184(2):501 – 520.

[Archard, 1953] Archard, J. F. (1953). Contact and Rubbing of Flat Surfaces. Journal of Applied Physics, 24(8):981–988.

[Baaijens, 2001] Baaijens, F. P. T. (2001). A fictitious domain/mortar element method for fluid–structure interaction. International Journal for Numerical Methods in Fluids, 35(7):743–761.

[Babuška, 1973] Babuška, I. (1973). The finite element method with Lagrangian multipliers. Numerische Mathematik, 20(3):179–192.

[Babuška and Melenk, 1997] Babuška, I. and Melenk, J. M. (1997). The partition of unity method. International journal for numerical methods in engineering, 40(4):727–758.

[Barbosa and Hughes, 1991] Barbosa, H. J. and Hughes, T. J. (1991). The finite element method with Lagrange multipliers on the boundary: circumventing the Babuška-Brezzi condition. Computer Methods in Applied Mechanics and Engineering, 85(1):109–128.

[Bathe, 1995] Bathe, K.-J. (1995). Finite Element Procedures. Prentice Hall, 1st edition. Published: Hardcover.

[Bathe, 2001] Bathe, K.-J. (2001). The inf–sup condition and its evaluation for mixed finite element methods. Computers & structures, 79(2):243–252.

[Bathe and Chaudhary, 1985] Bathe, K.-J. and Chaudhary, A. (1985). A solution method for planar and axisymmetric contact problems. International Journal for Numerical Methods in Engineering, 21(1):65–88.

[Béchet et al., 2009] Béchet, É., Moës, N., and Wohlmuth, B. (2009). A stable Lagrange multiplier space for stiff interface conditions within the extended finite element method. International Journal for Numerical Methods in Engineering, 78(8):931–954.

[Belgacem, 1999] Belgacem, F. B. (1999). The mortar finite element method with Lagrange multipliers. Numerische Mathematik, 84(2):173–197.

[Belgacem et al., 1998] Belgacem, F. B., Hild, P., and Laborde, P. (1998). The mortar finite element method for contact problems. Mathematical and Computer Modelling, 28(4):263–272.

[Belgacem and Maday, 1994] Belgacem, F. B. and Maday, Y. (1994). A spectral element methodology tuned to parallel implementations. Computer Methods in Applied Mechanics and Engineering, 116(1-4):59–67.

[Belytschko et al., 2009] Belytschko, T., Gracie, R., and Ventura, G. (2009). A review of extended/generalized finite element methods for material modeling. Modelling and Simulation in Materials Science and Engineering, 17(4):043001.

[Belytschko et al., 2013] Belytschko, T., Liu, W. K., Moran, B., and Elkhodary, K. (2013). Nonlinear finite elements for continua and structures. John wiley & sons.

[Belytschko et al., 2003] Belytschko, T., Parimi, C., Moës, N., Sukumar, N., and Usui, S. (2003). Structured extended finite element methods for solids defined by implicit surfaces. International journal for numerical methods in engineering, 56(4):609–635.

[Bernardi, 1994] Bernardi, C. (1994). A new nonconforming approach to domain decomposition: the mortar element method. Nonliner Partial Differential Equations and Their Applications.

[Bernardi et al., 1990] Bernardi, C., Debit, N., and Maday, Y. (1990). Coupling finite element and spectral methods: First results. Mathematics of Computation, 54(189):21–39.

[Bernardi et al., 2005] Bernardi, C., Maday, Y., and Rapetti, F. (2005). Basics and some applications of the mortar element method. GAMM-Mitteilungen, 28(2):97–123.

[Besson and Foerch, 1997] Besson, J. and Foerch, R. (1997). Large scale object-oriented finite element code design. Computer methods in applied mechanics and engineering, 142(1-2):165–187.

[Björkman et al., 1995] Björkman, G., Klarbring, A., Sjödin, B., Larsson, T., and Rönnqvist, M. (1995). Sequential quadratic programming for non-linear elastic contact problems. International journal for numerical methods in engineering, 38(1):137–165.

[Boer et al., 2007] Boer, A. d., Zuijlen, A. H. v., and Bijl, H. (2007). Review of coupling methods for non-matching meshes. Computer Methods in Applied Mechanics and Engineering, 196(8):1515 – 1525.

[Bonet and Wood, 1997] Bonet, J. and Wood, R. D. (1997). Nonlinear continuum mechanics for finite element analysis. Cambridge university press.

[Brezzi and Fortin, 2012] Brezzi, F. and Fortin, M. (2012). Mixed and hybrid finite element methods, volume 15. Springer Science & Business Media.

[Burman et al., 2015] Burman, E., Claus, S., Hansbo, P., Larson, M. G., and Massing, A. (2015). CutFEM: Discretizing geometry and partial differential equations. International Journal for Numerical Methods in Engineering, 104(7):472–501.

[Burman and Hansbo, 2010] Burman, E. and Hansbo, P. (2010). Fictitious domain finite element methods using cut elements: I. A stabilized Lagrange multiplier method. Computer Methods in Applied Mechanics and Engineering, 199(41-44):2680–2686.

[Bykat, 1976] Bykat, A. (1976). Automatic generation of triangular grid: I—Subdivision of a general polygon into convex subregions. II—Triangulation of convex polygons. International Journal for Numerical Methods in Engineering, 10(6):1329–1342.

[Chabrand et al., 1998] Chabrand, P., Dubois, F., and Raous, M. (1998). Various numerical methods for solving unilateral contact problems with friction. Mathematical and Computer Modelling, 28(4-8):97–108.

[Chahine, 2008] Chahine, E. (2008). Etude mathématique et numérique de méthodes d'éléments finis étendues pour le calcul en domaines fissurés. PhD Thesis, Institut National des Sciences Appliquées de Toulouse.

[Chahine et al., 2011] Chahine, E., Laborde, P., and Renard, Y. (2011). A non-conformal eXtended Finite Element approach: Integral matching Xfem. Applied Numerical Mathematics, 61(3):322–343.

[Claus and Kerfriden, 2018] Claus, S. and Kerfriden, P. (2018). A stable and optimally convergent LaTIn-CutFEM algorithm for multiple unilateral contact problems. International Journal for Numerical Methods in Engineering, 113(6):938–966.

[Daux et al., 2000] Daux, C., Moës, N., Dolbow, J., Sukumar, N., and Belytschko, T. (2000). Arbitrary branched and intersecting cracks with the extended finite element method. International Journal for Numerical Methods in Engineering, 48(12):1741–1760.

[Dhia and Rateau, 2005] Dhia, H. B. and Rateau, G. (2005). The Arlequin method as a flexible engineering design tool. International journal for numerical methods in engineering, 62(11):1442–1462.

[Diez et al., 2013] Diez, P., Cottereau, R., and Zlotnik, S. (2013). A stable extended FEM formulation for multi-phase problems enforcing the accuracy of the fluxes through Lagrange multipliers. International Journal for Numerical Methods in Engineering, 96(5):303–322.

[Doca et al., 2014] Doca, T., Pires, F. A., and de Sa, J. C. (2014). A frictional mortar contact approach for the analysis of large inelastic deformation problems. International Journal of Solids and Structures, 51(9):1697–1715.

[Dolbow et al., 2001] Dolbow, J., Moës, N., and Belytschko, T. (2001). An extended finite element method for modeling crack growth with frictional contact. Computer methods in applied Mechanics and engineering, 190(51-52):6825–6846.

[Dolbow, 1999] Dolbow, J. E. (1999). An extended finite element method with discontinuous enrichment for applied mechanics. Northwestern university.

[Duboeuf and Béchet, 2017a] Duboeuf, F. and Béchet, E. (2017a). Embedded solids of any dimension in the X-FEM. Part I – Building a dedicated P1 function space. Finite Elements in Analysis and Design, 130:80 – 101.

[Duboeuf and Béchet, 2017b] Duboeuf, F. and Béchet, E. (2017b). Embedded solids of any dimension in the X-FEM. Part II –Imposing Dirichlet boundary conditions. Finite Elements in Analysis and Design, 128:32 – 50.

[Dumont, 1995] Dumont, G. (1995). The active set algorithm for solving frictionless unilateral contact problems. In Contact mechanics, pages 263–266. Springer.

[El-Abbasi and Bathe, 2001] El-Abbasi, N. and Bathe, K.-J. (2001). Stability and patch test performance of contact discretizations and a new solution algorithm. Computers & Structures, 79(16):1473–1486.

[Eshelby, 1959] Eshelby, J. D. (1959). The elastic field outside an ellipsoidal inclusion. Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences, 252(1271):561–569.

[Farah et al., 2016] Farah, P., Gitterle, M., Wall, W. A., and Popp, A. (2016). Computational wear and contact modeling for fretting analysis with isogeometric dual mortar methods. In Key Engineering Materials, volume 681, pages 1–18. Trans Tech Publ.

[Farah et al., 2017] Farah, P., Wall, W. A., and Popp, A. (2017). An implicit finite wear contact formulation based on dual mortar methods. International Journal for Numerical Methods in Engineering, 111(4):325–353.

[Farhat, 1988] Farhat, C. (1988). A simple and efficient automatic FEM domain decomposer. Computers & Structures, 28(5):579–602.

[Farhat and Crivelli, 1989] Farhat, C. and Crivelli, L. (1989). A general approach to nonlinear FE computations on shared-memory multiprocessors. Computer Methods in Applied Mechanics and Engineering, 72(2):153–171.

[Farhat and Wilson, 1987] Farhat, C. and Wilson, E. (1987). A new finite element concurrent computer program architecture. International Journal for Numerical Methods in Engineering, 24(9):1771–1792.

[Farhat and Wilson, 1988] Farhat, C. and Wilson, E. (1988). A parallel active column equation solver. Computers & Structures, 28(2):289–304.

[Feld-Payet et al., 2015] Feld-Payet, S., Chiaruttini, V., Besson, J., and Feyel, F. (2015). A new marching ridges algorithm for crack path tracking in regularized media. International Journal of Solids and Structures, 71:57–69.

[Fernández-Méndez and Huerta, 2004] Fernández-Méndez, S. and Huerta, A. (2004). Imposing essential boundary conditions in mesh-free methods. Computer methods in applied mechanics and engineering, 193(12-14):1257–1275.

[Ferté et al., 2014] Ferté, G., Massin, P., and Moës, N. (2014). Interface problems with quadratic X-FEM: design of a stable multiplier space and error analysis. International Journal for Numerical Methods in Engineering, 100(11):834–870.

[Fichera, 1973] Fichera, G. (1973). Boundary value problems of elasticity with unilateral constraints. In Linear Theories of Elasticity and Thermoelasticity, pages 391–424. Springer.

[Fischer and Wriggers, 2005] Fischer, K. and Wriggers, P. (2005). Frictionless 2d contact formulations for finite deformations based on the mortar method. Computational Mechanics, 36(3):226–244.

[Fischer and Wriggers, 2006] Fischer, K. A. and Wriggers, P. (2006). Mortar based frictional contact formulation for higher order interpolations using the moving friction cone. Computer methods in applied mechanics and engineering, 195(37):5020–5036.

[Forest and Amestoy, 2018] Forest, S. and Amestoy, M. (2018). Lecture notes Mćanique des milieux continus. Ecole des Mines de Paris.

[Fournié et al., 2014] Fournié, M., Lozinski, A., and others (2014). A fictitious domain approach for Fluid-Structure Interactions based on the eXtended Finite Element Method. ESAIM: Proceedings and Surveys, 45:308–317.

[Fouvry et al., 1996] Fouvry, S., Kapsa, P., and Vincent, L. (1996). Quantification of fretting damage. Wear, 200(1-2):186–205.

[Francavilla and Zienkiewicz, 1975] Francavilla, A. and Zienkiewicz, O. (1975). A note on numerical computation of elastic contact problems. International Journal for Numerical Methods in Engineering, 9(4):913–924.

[Fujun et al., 2000] Fujun, W., Jiangang, C., and Zhenhan, Y. (2000). A contact searching algorithm for contact-impact problems. Acta Mechanica Sinica, 16(4):374–382.

[Gitterle et al., 2010] Gitterle, M., Popp, A., Gee, M. W., and Wall, W. A. (2010). Finite deformation frictional mortar contact using a semi-smooth Newton method with consistent linearization. International Journal for Numerical Methods in Engineering, 84(5):543–571.

[Glowinski and Le Tallec, 1989] Glowinski, R. and Le Tallec, P. (1989). Augmented Lagrangian and operator-splitting methods in nonlinear mechanics, volume 9. SIAM.

[Glowinski et al., 1994] Glowinski, R., Pan, T.-W., and Periaux, J. (1994). A fictitious domain method for Dirichlet problem and applications. Computer Methods in Applied Mechanics and Engineering, 111(3-4):283–303.

[Gravouil et al., 2011] Gravouil, A., Pierres, E., and Baietto, M.-C. (2011). Stabilized global–local X-FEM for 3d non-planar frictional crack using relevant meshes. International Journal for Numerical Methods in Engineering, 88(13):1449–1475.

[Gross and Reusken, 2007] Gross, S. and Reusken, A. (2007). An extended pressure finite element space for two-phase incompressible flows with surface tension. Journal of Computational Physics, 224(1):40 – 58.

[Gupta and Ramirez, 1995] Gupta, S. and Ramirez, M. R. (1995). A mapping algorithm for domain decomposition in massively parallel finite element analysis. Computing Systems in Engineering, 6(2):111–150.

[Haslinger and Renard, 2009] Haslinger, J. and Renard, Y. (2009). A new fictitious domain approach inspired by the extended finite element method. SIAM Journal on Numerical Analysis, 47(2):1474–1499.

[Hautefeuille et al., 2012] Hautefeuille, M., Annavarapu, C., and Dolbow, J. E. (2012). Robust imposition of Dirichlet boundary conditions on embedded surfaces. International Journal for Numerical Methods in Engineering, 90(1):40–64.

[Hertz, 1882] Hertz, H. (1882). Ueber die Berührung fester elastischer Korper. Journal für die reine und angewandte Mathematik, 92:156–171.

[Hervé and Zaoui, 1995] Hervé, E. and Zaoui, A. (1995). Elastic behaviour of multiply coated fibre-reinforced composites. International Journal of Engineering Science, 33(10):1419–1433.

[Hestenes, 1969] Hestenes, M. R. (1969). Multiplier and gradient methods. Journal of optimization theory and applications, 4(5):303–320.

[Hild, 2000] Hild, P. (2000). Numerical implementation of two nonconforming finite element methods for unilateral contact. Computer Methods in Applied Mechanics and Engineering, 184(1):99–123.

[Holzapfel, 2002] Holzapfel, G. A. (2002). Nonlinear solid mechanics: a continuum approach for engineering science. Meccanica, 37(4):489–490.

[Ji et al., 2002] Ji, H., Chopp, D., and Dolbow, J. (2002). A hybrid extended finite element/level set method for modeling phase transformations. International Journal for Numerical Methods in Engineering, 54(8):1209–1233.

[Johansson, 1994] Johansson, L. (1994). Numerical simulation of contact pressure evolution in fretting. Journal of tribology, 116(2):247–254.

[Johnson, 1985] Johnson, K. L. (1985). Contact Mechanics. Cambridge University Press.

[Kachanov et al., 2013] Kachanov, M. L., Shafiro, B., and Tsukrov, I. (2013). Handbook of elasticity solutions. Springer Science & Business Media.

[Khoei and Nikbakht, 2006] Khoei, A. and Nikbakht, M. (2006). Contact friction modeling with the extended finite element method (X-FEM). Journal of materials processing technology, 177(1-3):58–62.

[Kikuchi and Oden, 1988] Kikuchi, N. and Oden, J. (1988). Contact Problems in Elasticity. Society for Industrial and Applied Mathematics.

[Kim et al., 2005] Kim, N. H., Won, D., Burris, D., Holtkamp, B., Gessel, G. R., Swanson, P., and Sawyer, W. G. (2005). Finite element analysis and experiments of metal/metal wear in oscillatory contacts. Wear, 258(11-12):1787–1793.

[Knuth, 1998] Knuth, D. E. (1998). The Art of Computer Programming, Volume 3: (2Nd Ed.) Sorting and Searching. Addison Wesley Longman Publishing Co., Inc., Redwood City, CA, USA.

[Konyukhov and Schweizerhof, 2008] Konyukhov, A. and Schweizerhof, K. (2008). On the solvability of closest point projection procedures in contact analysis: analysis and solution strategy for surfaces of arbitrary geometry. Computer Methods in Applied Mechanics and Engineering, 197(33-40):3045–3056.

[Kopačka et al., 2016] Kopačka, J., Gabriel, D., Plešek, J., and Ulbin, M. (2016). Assessment of methods for computing the closest point projection, penetration, and gap functions in contact searching problems. International Journal for Numerical Methods in Engineering, 105(11):803–833.

[Kuss and Lebon, 2009] Kuss, F. and Lebon, F. (2009). Stress based finite element methods for solving contact problems: Comparisons between various solution methods. Advances in Engineering Software, 40(8):697 – 706.

[Laursen and Simo, 1993] Laursen, T. and Simo, J. (1993). A continuum-based finite element formulation for the implicit solution of multibody, large deformation-frictional contact problems. International Journal for numerical methods in engineering, 36(20):3451–3485.

[Laursen, 1992] Laursen, T. A. (1992). Formulation and treatment of frictional contact problems using finite elements. PhD Thesis.

[Laursen, 2013] Laursen, T. A. (2013). Computational contact and impact mechanics: fundamentals of modeling interfacial phenomena in nonlinear finite element analysis. Springer Science & Business Media.

[Lee and Schachter, 1980] Lee, D.-T. and Schachter, B. J. (1980). Two algorithms for constructing a Delaunay triangulation. International Journal of Computer & Information Sciences, 9(3):219–242.

[Lengiewicz and Stupkiewicz, 2013] Lengiewicz, J. and Stupkiewicz, S. (2013). Efficient model of evolution of wear in quasi-steady-state sliding contacts. Wear, 303(1-2):611–621.

[Liang and Barsky, 1983] Liang, Y.-D. and Barsky, B. A. (1983). An analysis and algorithm for polygon clipping. Communications of the ACM, 26(11):868–877.

[Liu and Borja, 2008] Liu, F. and Borja, R. I. (2008). A contact algorithm for frictional crack propagation with the extended finite element method. International Journal for Numerical methods in engineering, 76(10):1489–1512.

[Marsden and Hughes, 1994] Marsden, J. E. and Hughes, T. J. (1994). Mathematical foundations of elasticity. Courier Corporation.

[Mayer et al., 2010] Mayer, U. M., Popp, A., Gerstenberger, A., and Wall, W. A. (2010). 3d fluid–structure-contact interaction based on a combined XFEM FSI and dual mortar contact approach. Computational Mechanics, 46(1):53–67.

[McColl et al., 2004] McColl, I., Ding, J., and Leen, S. (2004). Finite element simulation and experimental validation of fretting wear. Wear, 256(11-12):1114–1127.

[McDevitt and Laursen, 2000] McDevitt, T. and Laursen, T. (2000). A mortar-finite element formulation for frictional contact problems. International Journal for Numerical Methods in Engineering, 48(10):1525–1547.

[McGee and Seshaiyer, 2005] McGee, W. and Seshaiyer, P. (2005). Non-conforming finite element methods for nonmatching grids in three dimensions. In Domain decomposition methods in science and engineering, pages 327–334. Springer.

[Mei et al., 2013] Mei, G., Tipper, J. C., and Xu, N. (2013). Ear-clipping based algorithms of generating high-quality polygon triangulation. In Proceedings of the 2012 International Conference on Information Technology and Software Engineering, pages 979–988. Springer.

[Melenk and Babuška, 1996] Melenk, J. M. and Babuška, I. (1996). The partition of unity finite element method: basic theory and applications. Computer methods in applied mechanics and engineering, 139(1-4):289–314.

[Moës et al., 2006] Moës, N., Béchet, E., and Tourbier, M. (2006). Imposing Dirichlet boundary conditions in the extended finite element method. International Journal for Numerical Methods in Engineering, 67(12):1641–1669.

[Mueller-Hoeppe et al., 2012] Mueller-Hoeppe, D., Wriggers, P., and Loehnert, S. (2012). Crack face contact for a hexahedral-based XFEM formulation. Computational Mechanics, 49(6):725–734.

[Muskhelishvili,] Muskhelishvili, N. Some basic problems of the mathematical theory of elasticity.

[Nikishkov et al., 1999] Nikishkov, G., Makinouchi, A., Yagawa, G., and Yoshimura, S. (1999). An algorithm for domain partitioning with load balancing. Engineering Computations, 16(1):120–135.

[Ogden, 1997] Ogden, R. W. (1997). Non-linear elastic deformations. Dover Publications 1997 and Ellis Horwood 1984.

[Öqvist, 2001] Öqvist, M. (2001). Numerical simulations of mild wear using updated geometry with different step size approaches. Wear, 249(1-2):6–11.

[Ortiz and Nour-Omid, 1986] Ortiz, M. and Nour-Omid, B. (1986). Unconditionally stable concurrent procedures for transient finite element analysis. Computer Methods in Applied Mechanics and Engineering, 58(2):151–174.

[Pantano and Averill, 2007] Pantano, A. and Averill, R. C. (2007). A penalty-based interface technology for coupling independently modeled 3d finite element meshes. Finite Elements in Analysis and Design, 43(4):271 – 286.

[Papadopoulos and Taylor, 1992] Papadopoulos, P. and Taylor, R. L. (1992). A mixed formulation for the finite element solution of contact problems. Computer Methods in Applied Mechanics and Engineering, 94(3):373–389.

[Parks et al., 2007] Parks, M. L., Romero, L., and Bochev, P. (2007). A novel Lagrange-multiplier based method for consistent mesh tying. Computer Methods in Applied Mechanics and Engineering, 196(35):3335 – 3347.

[Pietrzak and Curnier, 1997] Pietrzak, G. and Curnier, A. (1997). Continuum mechanics modelling and augmented Lagrangian formulation of multibody, large deformation frictional contact problems. Computational Plasticity, pages 878–883.

[Pietrzak and Curnier, 1999] Pietrzak, G. and Curnier, A. (1999). Large deformation frictional contact mechanics: continuum formulation and augmented Lagrangian treatment. Computer Methods in Applied Mechanics and Engineering, 177(3-4):351–381.

[Plimpton et al., 1998] Plimpton, S., Attaway, S., Hendrickson, B., Swegle, J., Vaughan, C., and Gardner, D. (1998). Parallel Transient Dynamics Simulations: Algorithms for Contact Detection and Smoothed Particle Hydrodynamics. Journal of Parallel and Distributed Computing, 50(1):104 – 122.

[Podra and Andersson, 1999] Podra, P. and Andersson, S. (1999). Simulating sliding wear with finite element method. Tribology international, 32(2):71–81.

[Popp, 2012] Popp, A. (2012). Mortar methods for computational contact mechanics and general interface problems. PhD Thesis, Technische Universität München.

[Poulios and Renard, 2015] Poulios, K. and Renard, Y. (2015). An unconstrained integral approximation of large sliding frictional contact between deformable solids. Computers & Structures, 153:75–90.

[Powell, 1978] Powell, M. (1978). Algorithms for nonlinear constraints that use lagrangian functions. Mathematical Programming, 14(1):224–248.

[Powell, 1969] Powell, M. J. (1969). A method for nonlinear constraints in minimization problems. Optimization, pages 283–298.

[Prokopyshyn, 2016] Prokopyshyn, . . (2016). Domain Decomposition Schemes Based on the Penalty Method for the Problems of Perfect Contact of Elastic Bodies. Journal of Mathematical Sciences, 212(1):46–66.

[Proudhon et al., 2016] Proudhon, H., Li, J., Wang, F., Roos, A., Chiaruttini, V., and Forest, S. (2016). 3d simulation of short fatigue crack propagation by finite element crystal plasticity and remeshing. International Journal of Fatigue, 82:238–246.

[Przemieniecki, 1963] Przemieniecki, J. S. (1963). Matrix structural analysis of substructures. AIAA Journal, 1(1):138–147.

[Puso, 2004] Puso, M. A. (2004). A 3d mortar method for solid mechanics. International Journal for Numerical Methods in Engineering, 59(3):315–336.

[Puso et al., 2008] Puso, M. A., Laursen, T., and Solberg, J. (2008). A segment-to-segment mortar contact method for quadratic elements and large deformations. Computer Methods in Applied Mechanics and Engineering, 197(6-8):555–566.

[Puso and Laursen, 2004] Puso, M. A. and Laursen, T. A. (2004). A mortar segment-to-segment contact method for large deformation solid mechanics. Computer methods in applied mechanics and engineering, 193(6):601–629.

[Rabinowicz and Tanner, 1966] Rabinowicz, E. and Tanner, R. (1966). Friction and wear of materials. Journal of Applied Mechanics, 33:479.

[Ramos et al., 2015] Ramos, A. C., Aragón, A. M., Soghrati, S., Geubelle, P. H., and Molinari, J.-F. (2015). A new formulation for imposing Dirichlet boundary conditions on non-matching meshes. International Journal for Numerical Methods in Engineering, 103(6):430–444.

[Raous et al., 1999] Raous, M., Cangémi, L., and Cocu, M. (1999). A consistent model coupling adhesion, friction, and unilateral contact. Computer methods in applied mechanics and engineering, 177(3-4):383–399.

[Reddy, 2014] Reddy, J. N. (2014). An Introduction to Nonlinear Finite Element Analysis: with applications to heat transfer, fluid mechanics, and solid mechanics. OUP Oxford.

[Ribeaucourt et al., 2007] Ribeaucourt, R., Baietto-Dubourg, M.-C., and Gravouil, A. (2007). A new fatigue frictional contact crack propagation model with the coupled X-FEM/LATIN method. Computer Methods in Applied Mechanics and Engineering, 196(33-34):3230–3247.

[Rockafellar, 1973] Rockafellar, R. T. (1973). A dual approach to solving nonlinear programming problems by unconstrained optimization. Mathematical Programming, 5(1):354–373.

[Rousselier et al., 2009] Rousselier, G., Barlat, F., and Yoon, J. W. (2009). A novel approach for anisotropic hardening modeling. Part I: Theory and its application to finite element analysis of deep drawing. International Journal of Plasticity, 25(12):2383–2409.

[Sacco and Lebon, 2012] Sacco, E. and Lebon, F. (2012). A damage–friction interface model derived from micromechanical approach. International Journal of Solids and Structures, 49(26):3666 – 3680.

[Sanders et al., 2009] Sanders, J. D., Dolbow, J. E., and Laursen, T. A. (2009). On methods for stabilizing constraints over enriched interfaces in elasticity. International Journal for Numerical Methods in Engineering, 78(9):1009–1036.

[Sanders et al., 2012] Sanders, J. D., Laursen, T. A., and Puso, M. A. (2012). A Nitsche embedded mesh method. Computational Mechanics, 49(2):243–257.

[Schloegel et al., 2000] Schloegel, K., Karypis, G., and Kumar, V. (2000). Graph partitioning for high performance scientific simulations. Army High Performance Computing Research Center.

[Sethian, 1999] Sethian, J. A. (1999). Level set methods and fast marching methods: evolving interfaces in computational geometry, fluid mechanics, computer vision, and materials science, volume 3. Cambridge university press.

[Sharma, 1979] Sharma, C. (1979). Circular inclusion in an infinite elastic strip. Zeitschrift für angewandte Mathematik und Physik ZAMP, 30(6):983–990.

[Signorini, 1933] Signorini, A. (1933). Sopra alcune questioni di elastostatica. Atti della Societa Italiana per il Progresso delle Scienze, 21(II):143–148.

[Simo and Laursen, 1992] Simo, J. C. and Laursen, T. (1992). An augmented Lagrangian treatment of contact problems involving friction. Computers & Structures, 42(1):97–116.

[Simo et al., 1985] Simo, J. C., Wriggers, P., and Taylor, R. L. (1985). A perturbed Lagrangian formulation for the finite element solution of contact problems. Computer methods in applied mechanics and engineering, 50(2):163–180.

[Simon, 1991] Simon, H. D. (1991). Partitioning of unstructured problems for parallel processing. Computing systems in engineering, 2(2):135–148.

[Sukumar et al., 2001] Sukumar, N., Chopp, D. L., Moës, N., and Belytschko, T. (2001). Modeling holes and inclusions by level sets in the extended finite-element method. Computer methods in applied mechanics and engineering, 190(46):6183–6200.

[Sun and Mao, 1988] Sun, C. and Mao, K. (1988). A global-local finite element method suitable for parallel computations. Computers & structures, 29(2):309–315.

[Sutherland and Hodgman, 1974] Sutherland, I. E. and Hodgman, G. W. (1974). Reentrant polygon clipping. Communications of the ACM, 17(1):32–42.

[Toselli and Widlund, 2006] Toselli, A. and Widlund, O. (2006). Domain decomposition methods-algorithms and theory, volume 34. Springer Science & Business Media.

[Truesdell and Noll, 2004] Truesdell, C. and Noll, W. (2004). The non-linear field theories of mechanics. In The non-linear field theories of mechanics, pages 1–579. Springer.

[Tur et al., 2009] Tur, M., Fuenmayor, F., and Wriggers, P. (2009). A mortar-based frictional contact formulation for large deformations using Lagrange multipliers. Computer Methods in Applied Mechanics and Engineering, 198(37-40):2860–2873.

[Voitsekhovskii, 1992] Voitsekhovskii, S. A. (1992). The fictitious domain method. Journal of Soviet Mathematics, 60(3):1437–1441.

[Weiler and Atherton, 1977] Weiler, K. and Atherton, P. (1977). Hidden surface removal using polygon area sorting. In ACM SIGGRAPH computer graphics, volume 11, pages 214–222. ACM.

[Williams and O'Connor, 1999] Williams, J. R. and O'Connor, R. (1999). Discrete element simulation and the contact problem. Archives of Computational Methods in Engineering, 6(4):279–304.

[Wohlmuth, 2000] Wohlmuth, B. I. (2000). A Mortar Finite Element Method Using Dual Spaces for the Lagrange Multiplier. SIAM J. Numer. Anal., 38(3):989–1012.

[Wohlmuth, 2001] Wohlmuth, B. I. (2001). Discretization Methods and Iterative Solvers Based on Domain Decomposition, volume 17 of Lecture Notes in Computational Science and Engineering. Springer Berlin Heidelberg, Berlin, Heidelberg.

[Wriggers, 2012] Wriggers, P. (2012). Computational contact mechanics. Springer.

[Wriggers et al., 1985] Wriggers, P., Simo, J., and Taylor, R. (1985). Penalty and augmented Lagrangian formulations for contact problems. In Proceedings of the NUMETA, volume 85, pages 97–105. Swansea, British Technique Press.

[Wriggers et al., 1990] Wriggers, P., Van, T. V., and Stein, E. (1990). Finite element formulation of large deformation impact-contact problems with friction. Computers & Structures, 37(3):319–331.

[Yagawa et al., 1991] Yagawa, G., Soneda, N., and Yoshimura, S. (1991). A large scale finite element analysis using domain decomposition method on a parallel computer. Computers & structures, 38(5-6):615–625.

[Yang and Laursen, 2008] Yang, B. and Laursen, T. A. (2008). A contact searching algorithm including bounding volume trees applied to finite sliding mortar formulations. Computational Mechanics, 41(2):189–205.

[Yang et al., 2005] Yang, B., Laursen, T. A., and Meng, X. (2005). Two dimensional mortar contact methods for large deformation frictional sliding. International Journal for Numerical Methods in Engineering, 62(9):1183–1225.

[Yastrebov, 2013] Yastrebov, V. (2013). Numerical methods in contact mechanics. ISTE/Wiley.

[Yastrebov et al., 2011a] Yastrebov, V., Cailletaud, G., and Feyel, F. (2011a). A local contact detection technique for very large contact and self-contact problems: sequential and parallel implementations. In Trends in computational contact mechanics, pages 227–251. Springer.

[Yastrebov et al., 2011b] Yastrebov, V. A., Durand, J., Proudhon, H., and Cailletaud, G. (2011b). Rough surface contact analysis by means of the finite element method and of a new reduced model. Comptes Rendus Mécanique, 339(7-8):473–490.

[Zamani and Eslami, 2011] Zamani, A. and Eslami, M. R. (2011). Embedded interfaces by polytope FEM. International Journal for Numerical Methods in Engineering, 88(8):715–748.

[Zavarise and Wriggers, 1998] Zavarise, G. and Wriggers, P. (1998). A segment-to-segment contact strategy. Mathematical and Computer Modelling, 28(4-8):497–515.

[Zienkiewicz et al., 1977] Zienkiewicz, O. C., Taylor, R. L., Zienkiewicz, O. C., and Taylor, R. L. (1977). The finite element method, volume 3. McGraw-hill London.

RÉSUMÉ

Cette thèse a pour but de développer un ensemble de méthodes permettant de gérer les problèmes de contact et de couplage de maillages dans le cadre de la méthode des éléments finis classiques et étendus. Ces problèmes d'interfaces sont traités le long de surfaces réelles et virtuelles, dites "surfaces immergées". Le premier objectif est d'élaborer une formulation de Mortar tridimensionnelle, efficace et parfaitement cohérente en utilisant la méthode du Lagrangien augmenté monolithique (ALM) pour traiter les problèmes de contact et de frottement. Cet objectif est réalisé dans le cadre de la méthode des éléments finis classique. Divers aspects du traitement numérique du contact sont discutés : la détection, la discrétisation, l'évaluation précise des intégrales de Mortar (projections, découpage, triangulation), la parallélisation du traitement sur des architectures parallèles à mémoire distribuée et l'optimisation de la convergence pour les problèmes impliquant à la fois le contact/frottement et les non-linéarités de comportement des matériaux. Grâce aux formulations de Mortar tirées des méthodes de décomposition de domaines, les problèmes de couplage de maillage pour la classe des interfaces non-compatibles sont également présentés.
En outre, une nouvelle méthode numérique a été élaborée en 2D : nous la dénommons "MorteX", car elle rassemble à la fois des fonctionnalités de la méthode Mortar et de la méthode X-FEM (méthode des éléments finis étendus). Dans ce cas, le couplage des maillages entre des domaines qui se chevauchent ainsi que le contact frottant entre des surfaces réelles d'un solide et certaines surfaces immergées au sein du maillage d'un autre corps peuvent être traités efficacement. Cependant, la gestion du couplage/contact entre des géométries non conformes à l'aide de surfaces immergées pose des problèmes de stabilité numérique. Nous avons donc proposé une technique de stabilisation qui consiste à introduire une interpolation des multiplicateurs de Lagrange à grains grossiers. Cette technique a été testée avec succès sur des "patch-tests" classiques et elle s'est également avérée utile pour les méthodes Mortar classiques, ce qui est illustré par plusieurs exemples pratiques.
La méthode MorteX est aussi utilisée pour traiter des problèmes d'usure en fretting. Dans ce cas, l'évolution des surfaces de contact qui résulte de l'enlèvement de matière dû à l'usure est modélisée comme une évolution de surface virtuelle qui se propage au sein du maillage existant. L'utilisation de la méthode MorteX élimine donc le besoin de recourir aux techniques complexes de remaillage. Les méthodes proposées sont développées et implémentées dans le logiciel éléments finis Z-set. De nombreux exemples numériques ont été considérés pour valider la mise en œuvre et démontrer la robustesse, la performance et la précision des méthodes Mortar et MorteX.

MOTS CLÉS

Méthode mortar, contact, frottement, couplage des maillages, méthode MorteX.

ABSTRACT

In this work we develop a set of methods to handle tying and contact problems along real and virtual (embedded) surfaces in the framework of the finite element method. The first objective is to elaborate an efficient and fully consistent three-dimensional mortar formulation using the monolithic augmented Lagrangian method (ALM) to treat frictional contact problems. Various aspects of the numerical treatment of contact are discussed: detection, discretization, accurate evaluation of mortar integrals (projections, clipping, triangulation), the parallelization on distributed memory architectures and optimization of convergence for problems involving both contact and material non-linearities. With mortar methods being drawn from the domain decomposition methods, the mesh tying problems for the class on non-matching interfaces is also presented.
A new two-dimensional MorteX framework, which combines features of the extended finite element method (X-FEM) and the classical mortar methods is elaborated. Within this framework, mesh tying between overlapping domains and contact between embedded (virtual) boundaries can be treated. However, in this setting, severe manifestation of mesh locking phenomenon can take place under specific problem settings both for tying and contact. Stabilization techniques such as automatic triangulation of blending elements and coarse-grained Lagrange multiplier spaces are proposed to overcome these adverse effects. In addition, the coarse graining of Lagrange multipliers was proven to be useful for classical mortar methods, which is illustrated with relevant numerical examples.
The MorteX framework is used to treat frictional wear problems. Within this framework the contact surface evolution as a result of material removal due to wear is modeled as an evolving virtual surface. Use of MorteX method circumvents the need for complex remeshing techniques to account for contact surface evolution. The proposed methods are developed and implemented in the in-house finite element suite Z-set. Numerous numerical examples are considered to validate the implementation and demonstrate the robustness, performance and accuracy of the proposed methods.

KEYWORDS

Mortar method, contact, friction, mesh tying, MorteX method.