

# An Efficient Finite Element Method for Interface Problems

by

Chandrasekhar Annavarapu

Department of Civil and Environmental Engineering  
Duke University

Date: \_\_\_\_\_

Approved:

---

John E. Dolbow, Supervisor

---

Tod A. Laursen

---

Thomas P. Witelski

---

Guglielmo Scovazzi

---

Martin Hautefeuille

Dissertation submitted in partial fulfillment of the requirements for the degree of  
Doctor of Philosophy in the Department of Civil and Environmental Engineering  
in the Graduate School of Duke University

2013

ABSTRACT

An Efficient Finite Element Method for Interface Problems

by

Chandrasekhar Annavarapu

Department of Civil and Environmental Engineering  
Duke University

Date: \_\_\_\_\_

Approved:

\_\_\_\_\_  
John E. Dolbow, Supervisor

\_\_\_\_\_  
Tod A. Laursen

\_\_\_\_\_  
Thomas P. Witeliski

\_\_\_\_\_  
Guglielmo Scovazzi

\_\_\_\_\_  
Martin Hautefeuille

An abstract of a dissertation submitted in partial fulfillment of the requirements for  
the degree of Doctor of Philosophy in the Department of Civil and Environmental  
Engineering  
in the Graduate School of Duke University  
2013

Copyright © 2013 by Chandrasekhar Annavarapu  
All rights reserved

# Abstract

We focus on developing a computationally efficient finite element method for interface problems. Finite element methods are severely constrained in their ability to resolve interfaces. Many of these limitations stem from their inability in independently representing interface geometry from the underlying discretization. We propose an approach that facilitates such an independent representation by embedding interfaces in the underlying finite element mesh. This embedding, however, raises stability concerns for existing algorithms used to enforce interfacial kinematic constraints. To address these stability concerns, we develop robust methods to enforce interfacial kinematics over embedded interfaces.

We begin by examining embedded Dirichlet problems – a simpler class of embedded constraints. We develop both stable methods, based on Lagrange multipliers, and stabilized methods, based on Nitsche’s approach, for enforcing Dirichlet constraints over three-dimensional embedded surfaces and compare and contrast their performance. We then extend these methods to enforce perfectly-tied kinematics for elastodynamics with explicit time integration. In particular, we examine the coupled aspects of spatial and temporal stability for Nitsche’s approach. We address the incompatibility of Nitsche’s method for explicit time integration by (a) proposing a modified weighted stress variational form, and (b) proposing a novel mass-lumping procedure.

We revisit Nitsche’s method and inspect the effect of this modified variational

form on the interfacial quantities of interest. We establish that the performance of this method, with respect to recovery of interfacial quantities, is governed significantly by the choice for the various method parameters *viz.* stabilization and weighting. We establish a relationship between these parameters and propose an optimal choice for the weighting. We further extend this approach to handle non-linear, frictional sliding constraints at the interface. The naturally non-symmetric nature of these problems motivates us to omit the symmetry term arising in Nitsche's method. We contrast the performance of the proposed approach with the more commonly used penalty method. Through several numerical examples, we show that with the proposed choice of weighting and stabilization parameters, Nitsche's method achieves the right balance between accurate constraint enforcement and flux recovery - a balance hard to achieve with existing methods. Finally, we extend the proposed approach to intersecting interfaces and conduct numerical studies on problems with junctions and complex topologies.

*To Amma (mom) and Nannagaru (dad),*

# Contents

<b>Abstract</b>	<b>iv</b>
<b>List of Tables</b>	<b>xi</b>
<b>List of Figures</b>	<b>xii</b>
<b>List of Abbreviations and Symbols</b>	<b>xxi</b>
<b>Acknowledgements</b>	<b>xxii</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Motivation . . . . .	1
1.1.1 Computational micro-mechanics . . . . .	2
1.1.2 Computational geosciences . . . . .	3
1.2 Computational challenges . . . . .	4
1.3 Thesis objectives and overview . . . . .	7
<b>2 Imposing Dirichlet Constraints Over Embedded Surfaces</b>	<b>9</b>
2.1 Introduction . . . . .	9
2.2 Model Problem and Variational Formulation . . . . .	13
2.2.1 Variational Methods for Constraint Enforcement . . . . .	15
2.2.2 Augmented Lagrangian and Nitsche’s method . . . . .	15
2.3 Discretization and Implementation . . . . .	17
2.3.1 Discretization of the bulk field . . . . .	17
2.3.2 Implementation of Nitsche’s method . . . . .	19

2.3.3	Discretization of the Lagrange multiplier . . . . .	24
2.3.4	Implementation of the stable Lagrange multiplier method . . . . .	31
2.4	Numerical Examples . . . . .	32
2.4.1	Triple cosine field . . . . .	34
2.4.2	Logarithmic field . . . . .	38
2.4.3	Comparative study on constraint enforcement . . . . .	42
2.5	Discussion . . . . .	43
2.6	Conclusion . . . . .	46
<b>3</b>	<b>Imposing Stiff Interfacial Constraints Over Embedded Surfaces in Explicit Dynamics</b>	<b>48</b>
3.1	Introduction . . . . .	48
3.2	Model Problem and Variational Formulations . . . . .	50
3.2.1	Variational forms . . . . .	52
3.3	Spatial Discretization . . . . .	56
3.3.1	Dual approach . . . . .	58
3.3.2	Primal approaches . . . . .	59
3.3.3	Estimate for the stabilization parameter . . . . .	61
3.4	Temporal Discretization . . . . .	64
3.4.1	Stability analysis . . . . .	64
3.4.2	Mass lumping approach based on kinetic energy conservation . . . . .	68
3.5	Numerical Examples . . . . .	71
3.5.1	Impact loading . . . . .	72
3.6	Conclusion . . . . .	81
<b>4</b>	<b>A Robust Nitsche’s Method for Interface Problems</b>	<b>84</b>
4.1	Introduction . . . . .	84
4.2	Model problem and variational formulation . . . . .	87



4.2.1	Weak form . . . . .	88
4.3	Spatial discretization . . . . .	91
4.3.1	Discrete equations . . . . .	93
4.3.2	Estimate for the stabilization parameter . . . . .	94
4.4	Numerical examples . . . . .	99
4.4.1	Sensitivity study . . . . .	99
4.4.2	Pure bending . . . . .	104
4.4.3	Popcorn interface . . . . .	111
4.5	Conclusion . . . . .	112
<b>5</b>	<b>A Nitsche Stabilized Embedded Finite Element Method for Frictional Sliding Problems</b>	<b>114</b>
5.1	Introduction . . . . .	114
5.2	Model problem and variational formulation . . . . .	116
5.2.1	Governing equations for linear elastostatics and sliding contact	116
5.2.2	Variational form . . . . .	119
5.3	Algorithmic treatment and discretization . . . . .	120
5.3.1	Weighted Nitsche's method for frictional sliding . . . . .	120
5.3.2	Spatial discretization . . . . .	122
5.3.3	Discrete equations . . . . .	123
5.3.4	Estimate for the stabilization parameter and interfacial weights	127
5.4	Numerical examples . . . . .	128
5.4.1	Perfectly plastic interface . . . . .	129
5.4.2	Coulombian frictional law . . . . .	134
5.5	Conclusion . . . . .	148
<b>6</b>	<b>A Nitsche Stabilized Embedded Finite Element Method for Junctions</b>	<b>150</b>

6.1	Introduction . . . . .	150
6.2	Model problem and variational formulation . . . . .	153
6.2.1	Weak form . . . . .	154
6.3	Discretization . . . . .	155
6.3.1	Stability . . . . .	158
6.4	Numerical examples . . . . .	163
6.4.1	Sensitivity study . . . . .	163
6.4.2	Poisson's equation . . . . .	166
6.4.3	Composite beam study . . . . .	168
6.4.4	Unit cell geometry . . . . .	172
6.4.5	Polycrystalline geometry . . . . .	176
6.5	Conclusion . . . . .	184
<b>7</b>	<b>Conclusion</b>	<b>186</b>
<b>A</b>	<b>A Nitsche Stabilized Embedded Finite Element Method for Junctions</b>	<b>189</b>
A.0.1	Frictional Sliding . . . . .	189
A.0.2	Weak form . . . . .	190
A.0.3	Discretization . . . . .	190
	<b>Bibliography</b>	<b>195</b>
	<b>Biography</b>	<b>206</b>

# List of Tables

5.1	Newton-Raphson convergence behavior of weighted Nitsche's method and Nitsche's method with respect to position of interface within an element. . . . .	133
5.2	Newton-Raphson convergence behavior of weighted Nitsche's method for the problem where a plate with an inclined interface loaded in compression. . . . .	136
5.3	Newton-Raphson convergence behavior of embedded methods for the sliding block problem. . . . .	139

# List of Figures

1.1	Illustration of representative length scales for multi-scale analysis. Figure reproduced from Zohdi and Wriggers (2005) . . . . .	2
1.2	Illustration of interfaces in geosciences and materials science. Figure on the top left (a) shows a scanning electron micrograph of a polycrystalline specimen while that on the top right (b) shows natural fracture networks that exist in fractured rock in the sub-surface. The bottom image shows the various fault networks that exist in southern California. Figures (a), (b) and (c) are reproduced from Meyers et al. (2006), Gudmundsson et al. (2001) and Plesch et al. (2007) respectively.	5
1.3	Schematic of the proposed methodology. Interface topology given by (a) is superimposed over an independently constructed finite element mesh (b) to result in the embedded interface method proposed here. .	8
2.1	The domain $\Omega$ is divided in two sub-domains, $\Omega^1$ and $\Omega^2$ , by an embedded surface $\Gamma^*$ . It is later discretized with a mesh which does not conform with the surface resulting in the two discretized sub-domains $\Omega^{1h}$ and $\Omega^{2h}$ . The embedded discretized surface $\Gamma^{*h}$ is defined by the zero-level set of an approximate signed distance function to $\Gamma^*$ . . . .	14
2.2	Representation of partial elements using X-FEM. The black squares are the nodes of the bulk mesh and the blue circles are the vertices of the mesh of the surface. . . . .	18
2.3	Simple example of two superposed cubes, each of them being meshed by 6 tetrahedra, cut by an embedded surface. Black squares are end-points and white squares are non-end-points. The circles are the vertices of the interfacial mesh, intersection of the bulk mesh $\Gamma^*$ . Blue circles are vital vertices and white ones are non-vital vertices. The vertex graph is generated with Graphviz (Gansner and North, 2000).	25
2.4	Lagrange multiplier shape functions of the example described in Figure 2.3(a) are defined on the shaded areas which represent their respective support. The blue circles are the corresponding vital vertices.	29

2.5	Comparison between $C_0$ -continuous, as proposed in Béchet et al. (2009), and piecewise continuous shape functions of the Lagrange multipliers defined at vital vertices for the example problem depicted in Figure 2.3(a). . . . .	30
2.6	Finite element approximation of the bulk and the flux for the triple cosine problem (2.42) with a planar (left) and a spherical (right) embedded surface. . . . .	35
2.7	Convergence study for the triple cosine problem (2.42) solved on a unit cubic domain with an embedded planar surface. . . . .	36
2.8	Sensitivity analysis for the triple cosine problem. Variation in the flux error as the surface is moved with respect to the background mesh. . . . .	37
2.9	Convergence study for the triple cosine problem solved on a unit cubic domain with an embedded spherical surface. . . . .	38
2.10	Convergence study for the logarithmic field problem (2.43) solved on a unit cubic domain with an embedded spherical surface. . . . .	39
2.11	Geometry of the computational domain (in green) for the logarithmic field problem with an embedded popcorn shaped surface (in blue). . . . .	40
2.12	Convergence study for the logarithmic field problem (2.43) solved on a unit cubic domain with an embedded popcorn shaped surface. . . . .	41
2.13	Convergence in the bulk field on the surface for Poisson problem solved on a unit cubic domain with an embedded spherical surface (a) and an embedded popcorn shaped surface (b). . . . .	43
2.14	Finite element approximation of the bulk field and the flux for the logarithmic field problem (2.43) with an embedded spherical surface (left) and a popcorn shaped surface (right). The errors in flux are relative and computed with the domain integral technique for Nitsche's method. . . . .	44
3.1	Domain $\Omega$ divided by the interface $\Gamma^*$ into sub-domains $\Omega^1$ and $\Omega^2$ . The Dirichlet boundaries $\Gamma_d^1$ , $\Gamma_d^2$ and the Neumann boundaries $\Gamma_n^1$ and $\Gamma_n^2$ are shown. The complementary part of the boundary is traction free. The normal to the boundary of each sub-domain, $\mathbf{n}^m$ , points outwards as shown. . . . .	51

3.2	Overlapping element formulation for a tetrahedral element cut by an embedded interface. The black circles are the physical nodes corresponding to the background mesh and the hollow circles are the ghost nodes. The blue circles represent the discretization of the embedded interface within an element. The discrete interface is constructed to be piecewise-planar throughout the mesh, and is based on a zero level-set. The discrete interfacial normal $\mathbf{n}$ in each element is then given by that of the discrete interfacial plane. . . . .	57
3.3	Overlapping element formulation for a one-dimensional bar element. The black circles are the physical nodes corresponding to the background mesh and the hollow circles are the ghost nodes. The blue circle represents the embedded interface. . . . .	66
3.4	Variation of normalized critical time step with the volume fraction of the cut element in $\Omega^1$ for the two lumping schemes. . . . .	70
3.5	Comparison of time evolution of displacement for conforming and embedded methods. . . . .	73
3.6	Variation of $L_2$ error in the constraint with time. . . . .	74
3.7	Plot of normal traction at the embedded surface using all the three embedded approaches at a time after the wave first crosses the embedded surface. . . . .	75
3.8	Illustration of the sensitivity of the critical time step with the interface location exhibited by various primal embedded approaches. . . . .	76
3.9	Comparison of time evolution of displacement and stress for a sinusoidal impact load applied at the bottom surface obtained using classical and embedded approaches for the case with an embedded popcorn shaped surface. . . . .	77
3.10	Time snapshot of the displacement contour for classical and embedded approaches with an embedded popcorn shaped surface. . . . .	78
3.11	Illustration of the microstructural geometry and computational domain with 32 randomly distributed spherical inclusions. . . . .	79
3.12	Comparison of time evolution of displacement and stress for a sinusoidal impact load applied at the bottom surface obtained using classical and embedded approaches for a cubic domain filled with microstructural inclusions. . . . .	80

3.13	Time snapshot of the displacement contour for classical and embedded approaches with an embedded popcorn shaped surface. . . . .	80
4.1	Interface $\Gamma^*$ partitions the bulk domains $\Omega^1$ and $\Omega^2$ . The Dirichlet boundaries $\Gamma_d^1$ , $\Gamma_d^2$ and the Neumann boundaries $\Gamma_n^1$ and $\Gamma_n^2$ are shown. The complimentary part of the boundary is traction free. The normal to the boundary of each domain, $\mathbf{n}^m$ , points outwards as shown. . . .	87
4.2	Overlapping element formulation for a tetrahedral element cut by an embedded interface. The black circles are the physical nodes corresponding to the background mesh and the hollow circles are the ghost nodes. The blue circles represent the discretization of the embedded interface within an element. The discrete interface is constructed to be piecewise-planar throughout the mesh, and is based on a zero level-set. The discrete interfacial normal $\mathbf{n}$ in each element is then given by that of the discrete interfacial plane. . . . .	92
4.3	Contour plot of the percentage error in $L_\infty$ norm for equivalent flux at the interface. The variation in the interfacial location $d_z/h$ is along the horizontal axis while the variation in contrast in material properties $\log(\kappa^2/\kappa^1)$ is along the vertical axis. Percentage error is indicated by the color contour. . . . .	101
4.4	Sensitivity in interfacial flux error with interfacial location. Percentage error in the interfacial flux in $L_\infty$ norm is plotted along the vertical axis and the normalized interfacial location is plotted along the horizontal axis for (a) high degree of heterogeneity (b) homogeneous material. .	102
4.5	Sensitivity in interfacial flux error with degree of heterogeneity. Percentage error in the interfacial flux in $L_\infty$ norm is plotted along the vertical axis and the log ratio of material parameters is plotted along the horizontal axis for (a) an interface approaching a lower internal boundary (b) an interface approaching an upper internal boundary. .	103
4.6	Geometry and loading for the beam bending problem. . . . .	104
4.7	Representative mesh geometry zoomed near the interface for the stress locking problem. . . . .	105
4.8	Bending stresses top: for classical Nitsche's method and bottom: for $\gamma$ Nitsche's method. . . . .	105
4.9	Convergence study for the beam bending problem solved on a rectangular beam with a straight vertical interface. . . . .	106

4.10	Plot of normal and tangential traction at the interface for beam bending problem. Contrast in material properties as well as distance between the closest internal boundary and the interface is varied from identical material properties ( $E^1 = E^2 = 10^3$ ) on both sides and $d_x = h/2$ for top-left and right to $E^1 = 10^9$ , $E^2 = 10^3$ and $d_x = .01$ for bottom-left and right. The inset in (a) shows the definition of parameters $d_x$ and $h$ . . . . .	108
4.11	Comparison of analytical surface tractions obtained using the proposed approach with those obtained using other choices of weights. The material parameters are $E^1 = 10^3$ , $E^2 = 10^9$ and $d_x = .05$ . . . . .	110
4.12	Comparison of analytical surface tractions with those obtained using $\gamma$ Nitsche's method for the pathological case: $E^1 = 10^9$ , $E^2 = 10^3$ and $d_x = .01$ . . . . .	110
4.13	Computational domain with $\Omega^2$ shown in green and $\Omega^1$ shown in blue for the logarithmic field problem with a popcorn shaped embedded surface. . . . .	111
4.14	Convergence of flux in $L_\infty$ norm for the logarithmic field problem with a popcorn shaped embedded surface. In the figure legend, we denote the error in flux obtained from the domain integral formulation by DI. . . . .	112
5.1	Domains $\Omega^1$ and $\Omega^2$ separated by a shared boundary $\Gamma_*$ . The Dirichlet boundaries $\Gamma_d^1$ , $\Gamma_d^2$ and the Neumann boundaries $\Gamma_n^1$ and $\Gamma_n^2$ are as shown. The complementary part of the boundary is traction free. The normal to the boundary of each sub-domain, $\mathbf{n}^m$ , points outwards from the domain $\Omega^m$ as shown. . . . .	117
5.2	Spatial discretization for the embedded interface formulation. The black circles are the physical nodes corresponding to the background mesh and the hollow circles are the phantom nodes. The red hollow circles represent the discretization of the embedded interface within an element. The discrete interface is constructed to be piecewise-planar throughout the mesh, and is based on a zero iso-surface of the level-set function. . . . .	123
5.3	A beam with a material interface loaded in pure compression: (a) illustration of geometry and loading conditions (b) representative finite element mesh and definition of the sensitivity parameters. . . . .	130



5.4	Residual strain in the fully unloaded state with yield stress at the interface presented in ascending order. Case (a) represents a frictionless sliding problem while case (f) represents a perfectly tied interface. All the other subcases allow for frictional sliding. . . . .	132
5.5	A unit square domain with an inclined material interface that exhibits Coulombian frictional behavior (a) illustration of geometry and loading conditions - the interface is inclined such that $\tan \theta = 0.2$ (b) finite element mesh used in computations. . . . .	135
5.6	Horizontal displacement contours and deformed geometry for compressive loading of a plate with an inclined material interface. The inclination of the interface is such that $\tan \theta = 0.2$ . Slip is predicted (left) when the frictional coefficient $\mu < \tan \theta$ while stick is predicted (right) when $\mu > \tan \theta$ . The deformation is scaled by a factor of 2. . .	137
5.7	Elastic block sliding on a rigid surface: (a) illustration of geometry and loading conditions: the applied traction boundary conditions are such that $p_y = 200$ units and $p_x = 60$ units. (b) Discretized problem domain used in computations. . . . .	137
5.8	Deformed geometry and contact tractions for the elastic block sliding on a rigid surface. . . . .	140
5.9	Contact tractions for the elastic block sliding on a rigid surface using penalty method with penalty parameters $\alpha_N = 1 \times 10^9$ and $\alpha_\tau = 8 \times 10^5$ . . . . .	142
5.10	A unit square domain loaded in compression and shear at the top surface and constrained at the bottom surface. Frictional sliding is allowed at the material interface $\Gamma_*$ . . . . .	143
5.11	Convergence of normal tractions with mesh refinement for weighted Nitsche's method (left) and penalty method (right). For the penalty method, the parameter used is $\alpha_N = \alpha_\tau = 5 \times 10^{-12}$ and was obtained from a parameter study. . . . .	144
5.12	Convergence of tangential slip with mesh refinement for weighted Nitsche's method (left) and penalty method (right). For the penalty method, the parameter used is $\alpha_N = \alpha_\tau = 5 \times 10^{-12}$ and was obtained from a parameter study. . . . .	144

5.13	Comparison of the relative performance of the weighted Nitsche and penalty methods in enforcing the non-interpenetration constraint (left) and recovery of normal tractions (right). The results for the penalty method are plotted for two parameters $\alpha_N = \alpha_\tau = 5 \times 10^{12}$ (in violet) and $\alpha_N = \alpha_\tau = 1 \times 10^{16}$ (in blue). Traction profile for weighted Nitsche (in red) and penalty method with $\alpha_N = \alpha_\tau = 5 \times 10^{12}$ (in violet) are nearly indistinguishable. . . . .	146
5.14	Finite element mesh for the problem simulating crack face frictional contact. The domain is a unit square $(0, 1) \times (0, 1)$ with an inclined crack extending from the coordinates $(0.3, 0.33)$ to $(0.7, 0.68)$ . . . . .	146
5.15	x and y displacement contours obtained using weighted Nitsche's method (top) and Penalty method (bottom) for the crack face frictional contact problem. . . . .	147
6.1	Schematic of the problem domain. Domains $\Omega^1$ , $\Omega^2$ and $\Omega^3$ and the Dirichlet and Neumann boundaries $\Gamma_d$ , $\Gamma_n$ are as shown. Each pair of domains $\Omega^a$ and $\Omega^b$ are separated by a shared internal boundary $\Gamma^{ab}$ . The normal to the boundary segment $\Gamma^{ab}$ is given by $\mathbf{n}^{ab}$ and points outwards as shown for the interface $\Gamma^{21}$ . . . . .	153
6.2	Spatial discretization for the embedded interface formulation. The black circles are the physical nodes corresponding to the background mesh and the hollow circles are the phantom nodes. The red hollow circles represent the discretization of the embedded interface within an element. . . . .	156
6.3	Decomposition of a parent element with three local domains into corresponding children. The shaded region in the element represents the physical volume of the element. The black circles represent the physical nodes while the black hollow circles represent the phantom nodes. The red circles represent the discretization of the embedded surface. . . . .	157
6.4	Representative elements for the case when the element boundary contains part of the Dirichlet boundary and the element contains: a single embedded interface (left); multiple intersecting interfaces (right). . . . .	161
6.5	Illustration of the parameters for the sensitivity study. On the left, for a particular position of the junction, the orientation of interfaces is shown. On the right, the various positions of the junction used in the study are shown as blue dots. The zoom on the right shows that the junction does not actually lie on the element edge. . . . .	164

6.6	Variation of stabilization parameter for the weighted (left) and the classical (right) Nitsche’s method as a function of the barycentric coordinates for the junction. The plotted values for the stabilization parameter are normalized with the element volume. The ranges for the contour plots on the top and bottom are different as seen from the figure legends. On top, the plots are over the entire range of variation of the stabilization parameter obtained from classical Nitsche’s method while on the bottom the range of variation for the stabilization parameter is that obtained from the weighted Nitsche’s method. . . .	165
6.7	Schematic of computational domain, finite element mesh and distribution of material properties. . . . .	167
6.8	Top: Variation of $\kappa^m \nabla u^m$ across the domain. Bottom left: Convergence in $L_2$ norm for the bulk variable. Bottom right: Convergence in energy norm for $\kappa^m \nabla u^m$ . . . . .	169
6.9	Illustration of the geometry and loading conditions for a beam with several material interfaces loaded in pure compression. . . . .	170
6.10	Illustration of the finite element mesh used in the composite beam study.	171
6.11	Displacement contours at the maximum prescribed load in the composite beam study. . . . .	172
6.12	Residual strain in the fully unloaded state with yield stress at the interface presented in ascending order. Case (a) represents a frictionless sliding problem while case (f) represents a perfectly tied interface. All the other sub-cases allow for frictional sliding. . . . .	173
6.13	Illustration of the geometry and loading conditions for the unit-cell problem. . . . .	174
6.14	Displacement field for the unit-cell problem for the case when the yield traction between the boundaries is $t_Y = 0.10$ units. . . . .	175
6.15	von Mises stress for the unit-cell problem presented in ascending order of yield tractions at the interface. . . . .	176
6.16	Description of the geometry, loading conditions and finite element mesh for studying shear response of a polycrystalline specimen. . . .	177
6.17	Distribution of Young’s moduli (top) and yield tractions (bottom) for randomly generated polycrystalline specimens. . . . .	178

6.18	Magnitude of displacement and von Mises stress for the polycrystalline geometry subjected to shear loading. Number of grains is varied from 14 to 200. Discontinuities in displacements are clearly evident in the contour plots at the grain boundaries. Stress concentrations at the junctions are also clearly visible. Grain boundary regions shown in black are under slip while those shown in white are sticking. . . . .	179
6.19	Spatially averaged stress-strain plot. . . . .	181
6.20	Schematic of the geometry and loading conditions for studying the effect of confinement on a polycrystalline sample. . . . .	182
6.21	Finite element mesh used for studying the effect of confinement on a polycrystalline specimen with 120 grains. . . . .	182
6.22	Spatially averaged stress-strain plot to study the effect of confining pressure. . . . .	183
A.1	Representative element with three intersecting interfaces separating the local domains $\Omega_e^p$ , $\Omega_e^q$ and $\Omega_e^r$ . . . . .	192

# List of Abbreviations and Symbols

FEM	Finite Element Method
X-FEM	eXtended Finite Element Method
G-FEM	Generalized Finite Element Method
CFL	Courant-Friedrichs-Levy
LMVV	Lagrange Multipliers on Vital Vertices
LATIN	LARge Time INcrement
DI	Domain Integral

# Acknowledgements

I have been fortunate to meet some extraordinary individuals during my time at Duke. The work done in this thesis would not have been possible without their support and this is a humble attempt to articulate my gratitude towards them.

Firstly, I wish to express my sincere thanks to Professor John Dolbow. I started Duke with little background in finite element methods. Without his initial guidance, the ideas in this thesis would never have come to form. At the same time, he also allowed me intellectual freedom as a senior graduate student to drive my own research and mature as a researcher. I have learned many things from John, not least his strong work ethic and approachability.

I am privileged to have worked closely with Dr. Martin Hautefeuille. It is no coincidence that some of my most productive time in graduate school coincided with my collaboration with him. His enthusiasm for science, in general, and computational mechanics, in particular, is infectious. I have had many fruitful discussions with him, over lunch and coffee, where he has encouraged even the silliest of my ideas with tremendous enthusiasm. I am sure, in years to come, he will continue to positively influence many generations of students as a director of his own research group.

Thanks are also due to the members of my committee. Professor Tod Laursen, has been a strong co-mentor, an excellent teacher and always approachable for advice on both academic and non-academic matters. Professor Guglielmo Scovazzi has always been willing to help. His advice has been most useful while I have been trying to

find my feet. Professor Thomas Witelski has taught me two of the most enjoyable classes I took at Duke. His ability in translating complex ideas in simple terms is amazing to me. In this regard, thanks are also due to Professor Wilkins Aquino who has made the abstract subject of functional analysis accessible through his wonderful class on advanced finite element methods.

I have also been fortunate to work with some great colleagues. I have been a great admirer of Anand Embar's systematic approach to problem-solving and his keen sense for detail. Temesgen Kindo's intellectual curiosity never ceases to amaze me. He is ever-willing to go to the board to the benefit of others and has taught me many concepts outside of the classroom. Wen Jiang has been a great office-mate and always quick to share a joke.

On a personal level, I have met many great people at Duke whose friendship I will cherish for the rest of my life. Anand Embar and Vindhya Kunduru have very much been family to me at Duke. Knowing that I could count on their help and advice whenever I needed assistance - work or otherwise, has always been very reassuring. Harishanker Gajendran, Frederick Owusu-Nimo, Ming-Yeng Lin, Manjul Kumar, Deepthi Muraleedharan, Kumar Reddi and Wen Jiang have all been great company and we have shared some very memorable times together. I would also like to take a moment to remember Abhijit Mahato, who would continue to be a great friend if not for the tragic turn of events.

My brother, Ravikanth Annavarapu, has been an incredible influence on me and I would never even have considered engineering, let alone graduate school if not for his guidance. Finally, I cannot overstate my parents' help, and whole-hearted support in everything I have done. As a small token of appreciation, this thesis is dedicated to them.

# Introduction

## 1.1 Motivation

Interfaces and defects play a dominant role in governing the response of many physical problems. From length-scales spanning micro and nano dimensions – where heterogeneities give rise to material interfaces, to length-scales spanning several kilometers – where interfaces arise as fault networks in geological systems, the resolution of interfacial mechanics is critical to understanding the physical behavior of these systems. With the advent of modern computing power, computer simulations potentially offer a means for investigating increasingly complex and realistic physical behavior in the presence of these interfaces. However, continuum modeling approaches in many of these areas are still in their infancy due, in part, to the lack of solvers that could efficiently resolve interfacial features.

In this work, we develop numerical techniques that facilitate these continuum approaches in a cost-efficient and accurate manner. In the rest of this chapter, we place this work in a broader context by discussing the various application areas where interfaces significantly influence the physical response of the material systems. We then discuss the associated modeling challenges with these interface problems in



computational mechanics and the strategies explored in this work for addressing the said challenges.

### 1.1.1 Computational micro-mechanics

In many engineering applications, there is an increasing need for advanced materials. For instance, the microelectronic device industry is characterized by its constant need for miniaturization and requires materials with properties far beyond the scope of traditional materials (Rooney et al., 2000). Similarly, in the aviation industry, advanced composite materials, with much larger impact and fire resistance are being considered to design the fans of aircraft engines (see Coroneos, 2012). Naturally then, in recent years, material science has placed much emphasis on developing new materials or a combination of existing materials with desired characteristic properties. Many important mechanical properties such as ultimate strength and ductility of materials depend on their microstructure. The key challenge then lies in determining an optimal microstructure for these materials so that we can maximize the performance of the material for a given set of properties. Also, from an engineering design perspective, it is imperative that we understand these non-traditional materials well and are able to predict their response with some confidence.

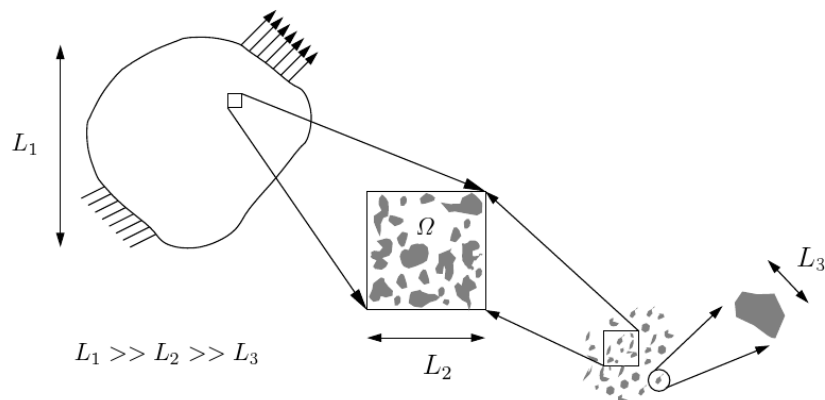


FIGURE 1.1 – Illustration of representative length scales for multi-scale analysis. Figure reproduced from Zohdi and Wriggers (2005)

Numerical simulations exhibit great potential in helping us answer some of these questions which are at the intersection of applied mechanics, materials science and solid-state physics. These simulations could not only help us in designing future experimental programs but also provide a virtual environment to conduct experiments hard to replicate in a physical setting. To a large extent, however, numerical modeling in materials science continues to be driven by molecular dynamics simulations. Even with the vast increase in the computing capabilities, the computationally intensive nature of atomistic simulations means that we are still unable to reproduce laboratory conditions in these simulations.

To overcome these limitations, there is an increasing trend of using molecular dynamics simulations in conjunction with continuum modeling approaches in a multi-scale framework. The basic premise of such a multi-scale approach is to first obtain the parameters governing the constitutive behavior from molecular-dynamics simulations and then hierarchically investigate the physical behavior at an intermediate length scale, called the meso-scale, where the continuum theories are still valid (see Figure 1.1 for a schematic). Finally information is transferred across to a higher length scale of relevance to the engineering design community. However, to put these multi-scale approaches in practice, we need a numerical method that can efficiently resolve the large heterogeneities and the grain boundary effects that arise at the meso-scale. Unfortunately, many existing methods are limited in their capabilities for handling such problems. Some of the limitations of traditional numerical approaches for interface problems are discussed in Section 1.2.

### *1.1.2 Computational geosciences*

Unconventional energy sources such as geothermal energy are increasingly being explored with an eye towards decreasing our dependence on fossil fuels (Tester et al., 2006). Geothermal energy refers to the thermal energy stored in earth's crust that

results primarily from radioactive decay of minerals at the core of the earth. This energy is stored both in the fluid that permeates the natural fractures in the earth's sub-surface as well as the constituent rock. Given the economic constraints, the feasibility of these geothermal reservoirs can only be determined by evaluating the recoverable fraction of this energy. Many well-known feasibility studies (see Sanyal and Butler, 2005) conclude that fractured rock volume, fracture spacing and fracture surface area are key parameters influencing the energy recovery process. We could then significantly improve the recovery process by engineering an optimal network of fractures that maximizes energy extraction. In addition, even conventional sources of energy rely heavily on artificially created fracture networks for efficient capturing of shale gas (Curtis, 2002).

Faults and fractures also arise in many other areas of geo-mechanics. The geological restoration process is increasingly using geomechanics based modeling techniques to predict the spatial and temporal evolution of reservoir deformation (Durand-Riard et al., 2010). Much emphasis is also being placed on incorporating fault topology in great detail for seismic hazard characterization (Bhat et al., 2007). A computational framework that allows these investigations in quick time will go a long way in helping us realize these objectives and meeting our energy requirements.

## 1.2 Computational challenges

Though seemingly very different, the above problems are unified by the computational challenges they face. A common thread that ties together all the areas listed above is the presence of interfaces. Interfaces pose many challenges to the traditional finite element methods. From a modeling perspective, interfaces often represent surfaces of discontinuities. Traditional finite element methods necessitate the finite element mesh to align with these surfaces of discontinuities in order to recover optimal rates of convergence. This is not necessarily surprising considering that shape

functions used to interpolate the solution are continuous within an element.

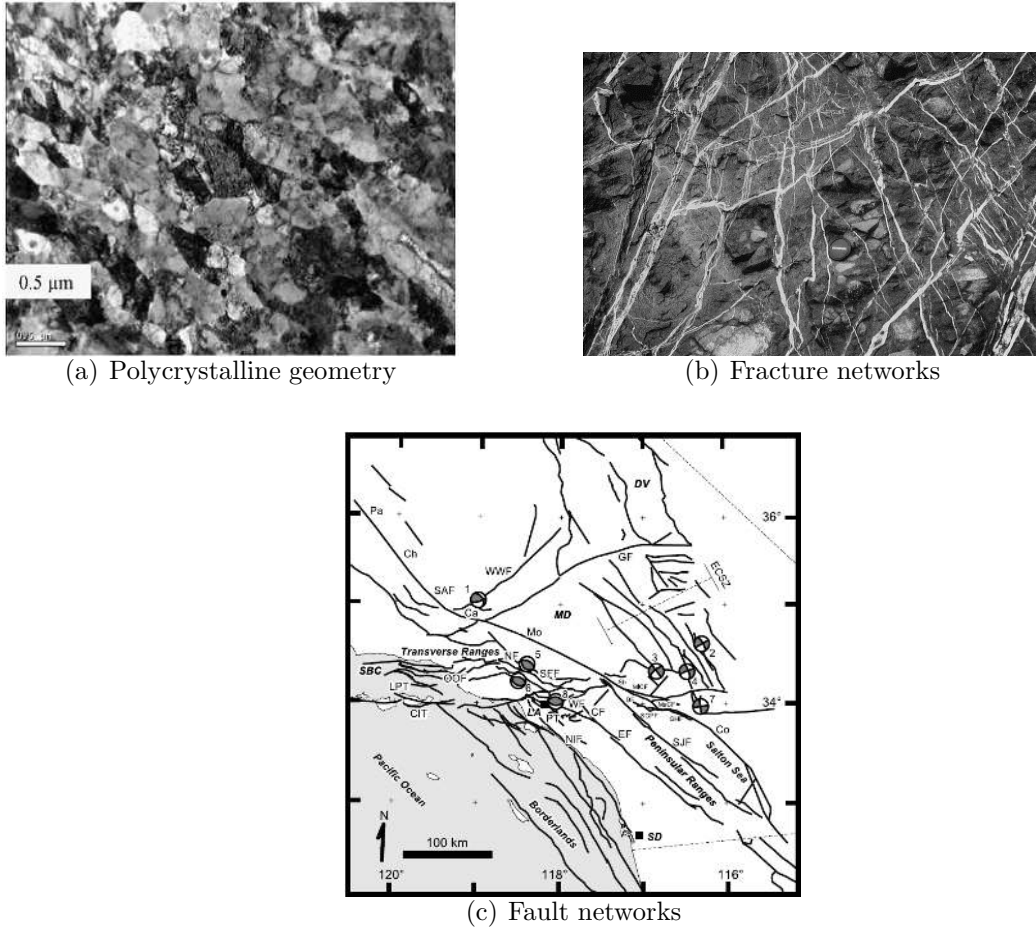


FIGURE 1.2 – Illustration of interfaces in geosciences and materials science. Figure on the top left (a) shows a scanning electron micrograph of a polycrystalline specimen while that on the top right (b) shows natural fracture networks that exist in fractured rock in the sub-surface. The bottom image shows the various fault networks that exist in southern California. Figures (a), (b) and (c) are reproduced from Meyers et al. (2006), Gudmundsson et al. (2001) and Plesch et al. (2007) respectively.

However, generating these finite element meshes that conform to all the interfacial features that arise in the physical problem being investigated is often not straightforward. To appreciate this further, consider the examples shown in Figure 1.2. In Figure 1.2(a), we show a scanning electron micrograph of a polycrystalline metal. Clearly, the polycrystalline microstructure shown here is very challenging to grid explicitly. In addition, to characterize the randomness inherent to a mi-

crostructure, computations need to be performed on several different microstructural realizations. Generating a conforming mesh for every realization makes such computations prohibitively expensive. Moreover, these conforming grids also result in several poor quality elements which need further treatment before being used in a finite element computation. For explicit dynamics simulations, required for investigating high strain-rate behavior such as blast/impact loading, these small elements worsen the temporal stability to an extent that these computations are no longer tractable. On the other hand, using a non-conforming grid results in poor accuracy and convergence rates. These issues are detailed further in Li and Zabaras (2009)

Virtually identical numerical issues also surface in computational geo-mechanics. In Figure 1.2(c), the various fault networks that arise in the southern California region are presented. Clearly, for reasons discussed above, finite element simulations studying fault rupture and motion of tectonic plates would require the generation of finite element meshes which match the fault topology. Similarly, the representation of fracture networks, both naturally existing and artificially generated, in applications for harnessing geothermal energy and conventional shale gas capture would also require complicated meshing procedures. Additionally, as these fracture networks evolve, there would be a need to continually remesh the domain. To avoid these challenges, many of these simulations either make simplifying assumptions on the topology or smear the fracture surfaces. However, as discussed in Bhat et al. (2007) and Marshall et al. (2008), these simplifying assumptions are likely to yield inaccurate results.

Moreover, even in the presence of explicitly gridded geometries, enforcing stiff kinematic constraints that arise when investigating frictional sliding along interfaces is a non-trivial challenge. Warner and Molinari (2006) describe the numerical challenges associated with exploring contact between grains and propose the use of an explicit multi-body contact algorithm. Further, Espinosa and Zavattieri (2003) illus-

trated the numerical instabilities that result from a poorly chosen initial slope for a phenomenological cohesive law in an interface-element based cohesive-zone formulation.

### 1.3 Thesis objectives and overview

The broader goal of this work is to develop a finite element method that alleviates meshing constraints and allows investigation of interfacial effects by embedding interfaces within a background mesh. We facilitate this embedding by enriching the standard finite element basis functions in the vicinity of the interface as described in the partition of unity framework (Babuška and Melenk, 1997; Moës et al., 1999). In other words, interface topology is represented independently of the finite element mesh. A schematic of the proposed approach is shown in Figure 1.3.

The particular contribution of this work is in addressing the stability issues that arise for these methods while enforcing stiff kinematic constraints over these embedded interfaces. We develop stable and stabilized variational methods that efficiently enforce these constraints while preserving the integrity of interfacial fluxes. The stable Lagrange multiplier approach builds on the vital vertex method of Béchet et al. (2009) and the stabilized approach is an extension of Nitsche’s method (Nitsche, 1971). We develop these methods to investigate both quasi-static and transient problems in elasticity. We also extend the stabilized methods to investigate frictional sliding over these embedded interfaces.

The outline of the thesis is as follows. We begin by developing methods to enforce embedded Dirichlet constraints in Chapter 2. These embedded Dirichlet problems are a convenient pre-cursor to the more difficult task of enforcing interfacial constraints. They preserve the stability challenges that arise for these mixed methods but significantly simplify notational complexity. We then extend the developed methods to enforce perfect continuity constraints across interfaces in a transient regime in

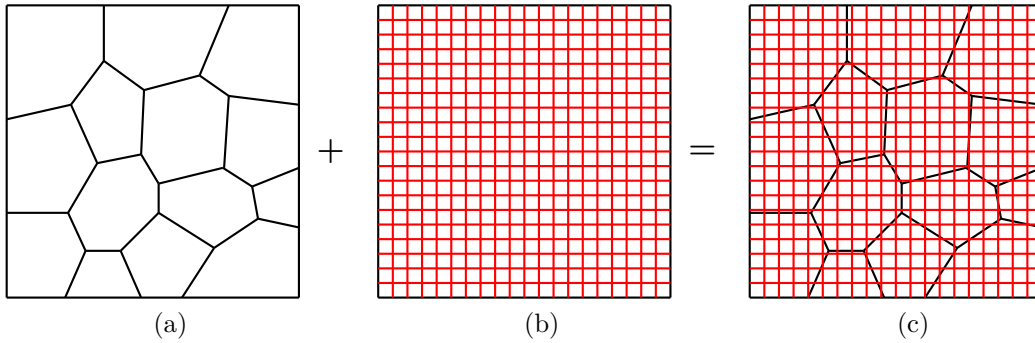


FIGURE 1.3 – Schematic of the proposed methodology. Interface topology given by (a) is superimposed over an independently constructed finite element mesh (b) to result in the embedded interface method proposed here.

Chapter 3. In particular, we establish the relationship between spatial and temporal stability for explicit dynamics problems. We propose a modification to Nitsche’s method as well as a mass-lumping procedure that allows explicit time-stepping for constrained interface problems. We then return to quasi-static problems and look back on the effect of the proposed modification to Nitsche’s method on the interfacial quantities of interest in Chapter 4. We further adapt the Nitsche stabilized method to enforce non-linear frictional sliding constraints in Chapter 5. Finally, in Chapter 6, we extend this method to domains with multiple intersecting interfaces within an element. In Chapter 7, we provide concluding remarks and comment on the future directions for this work.

# Imposing Dirichlet Constraints Over Embedded Surfaces

## 2.1 Introduction

This Chapter focuses on weak methods for enforcing Dirichlet constraints on embedded surfaces. These embedded Dirichlet problems arise frequently while modeling processes such as solidification which involve a moving front (see Chessa et al. (2002) and references therein). Often, Dirichlet constraints need to be enforced on this evolving front. In addition, the front velocity is governed by the flux across this evolving interface. So a robust method to model these problems would not only require enforcement of Dirichlet constraints with good accuracy but also an accurate recovery of the interfacial flux. Also, as discussed previously, these problems are a convenient pre-cursor to the more challenging case of enforcing stiff interfacial kinematic constraints such as those arising in perfect and sliding contact where stability in the interfacial traction field is again essential. With these issues in mind, in this Chapter, we investigate variational strategies for constraint enforcement on embedded Dirichlet boundaries that ensure good accuracy in both the primary variable and



the gradient.

While enforcing such Dirichlet constraints can be accomplished in a straightforward manner through collocation approaches for aligned meshes, these approaches are no longer available for embedded interfaces. This is primarily because the approximation for elements intersected by interfaces is constructed with shape functions that only reside in the physical part of these elements. Consequently, the shape functions no longer satisfy the Kronecker-delta property and render the collocation techniques inadequate. Most approaches to date have focused on weakly imposing Dirichlet conditions, such as in the early work by Dolbow et al. (2001) and Belytschko et al. (2003) using Lagrange multipliers. Ji and Dolbow (2004) subsequently showed that the most convenient choice of Lagrange multiplier basis triggered instabilities. Traction oscillations for stiff cohesive laws on embedded surfaces was also reported by Simone (2004).

For moving boundary problems, a significant amount of work has also been done in the immersed boundary communities that is related to the issues of concern here, but that has largely developed independently. In particular, the Ghost Fluid Method of Fedkiw et al. (1999), the Immersed Interface Method of Leveque and Li (1994) and the Immersed Boundary Methods of Peskin (2002) are some of the most popular finite-difference based approaches. These methods were developed for applications ranging from modeling multi-component flows to fluid-structure interaction and allowed for interfaces to be embedded in the background grid. In their classical forms, however, these methods are reported to be only first order accurate in the vicinity of the interface. Additionally, these methods are also, in general, known to return poor approximations to gradients at the interface. Recent extensions of Bedrossian et al. (2010) address these shortcomings to some extent.

The stability issues experienced by finite element methods with embedded interfaces can be traced back to the theory of saddle point problems in mixed finite

element methods. In the finite-element community, the stability of mixed variational problems is explained by the classical results of Babuška (1973) and Brezzi (1974). In essence, these results prove that stability can only be ensured if the interpolating spaces for the primary variable and the multiplier satisfy the inf-sup condition. In practice, Pitkäranta (see Pitkäranta, 1979, 1980, 1981) proved that this severely restricts the choice of approximating spaces for the multiplier field and often rules out the most-convenient choice of the multiplier basis.

Mixed formulations that are unstable have classically been resolved using one of two approaches: developing stable and stabilized methods. Stable methods seek to develop elements where the interpolating spaces for the bulk field and the multipliers satisfy the inf-sup condition by construction. Stabilized methods, on the other hand, consider elements that violate the inf-sup condition by construction and add stabilizing terms to restore stability. Early work on stabilized Lagrange multipliers on fitted surfaces was performed by Barbosa and Hughes (1991).

With respect to constraints on embedded interfaces, perhaps some of the earliest work using Lagrange multipliers to enforce constraints on embedded surfaces with finite elements was the fictitious domain method introduced by Glowinski et al. (1994). Recently, both stable and stabilized approaches have been examined for embedded interfaces. Moës et al. (2006) developed a stable Lagrange multiplier method by algorithmically coarsening the multiplier space and using a piecewise linear interpolation over this coarsened space. The use of a separate, coarse interfacial mesh to discretize the multiplier in a mortar-like approach was proposed by Kim et al. (2007). Both of these approaches effectively coarsen the discretization of the multiplier with respect to the bulk mesh. However neither is particularly convenient to extend to three-dimensional problems. Gerstenberger and Wall (2010) developed an alternative mixed approach in 3D where the constraining Lagrange multiplier represented the stress field as opposed to the interfacial traction of traditional mixed

formulations. While the developed method performed well for the chosen numerical examples, the mathematical basis for the stability of this approach is still being developed. More recently, Béchet et al. (2009) developed an efficient algorithm to effect a coarsening of the multiplier space. They also proposed the use of the trace of the bulk-shape functions as a basis for the multipliers. Both the proposed coarsening and the choice of basis are particularly attractive as opposed to previously developed methods because they can be readily extended to three-dimensional problems.

Stabilized approaches have largely relied on Nitsche's method (Nitsche, 1971), such as the work of Hansbo and Hansbo (2002) and Dolbow and Harari (2009). More recently, Burman and Hansbo (2010) developed a stabilized Lagrange multiplier method for fictitious domains using cut elements. Codina and co-workers examined various stabilized methods (including Nitsche's) for imposing Dirichlet boundary conditions in immersed boundary methods (see Codina and Baiges, 2009; Baiges et al., 2012). However, all of the aforementioned studies were limited to two-dimensional problems.

Recent work by Lew and Buscaglia (2008) (see also Rangarajan et al., 2009) developed an immersed boundary method for three-dimensional problems by replacing intersected elements with a layer of discontinuous-Galerkin elements. It is noteworthy that many of these stabilized approaches involve a mesh-dependent free-parameter that controls the stability of the method. Also, for interface orientations that result in arbitrarily small element configurations, problems associated with the conditioning of the discrete system of equations are also quite common.

In this Chapter, we extend the works of Béchet et al. (2009) and Dolbow and Harari (2009) to three-dimensional problems. In principle, both the stable method of Béchet et al. (2009) and the stabilized method of Dolbow and Harari (2009) readily extend to three-dimensional problems, but no convergence studies had been published prior to this work. While the stabilized methods are relatively straightforward

to extend to three dimensions, stable Lagrange multiplier methods based on the vital vertex method of Béchet et al. (2009) are more complex. We provide essential algorithmic and implementational details for the vital vertex method in three dimensions and introduce a new discontinuous basis for the corresponding Lagrange multiplier on the embedded surface.

This Chapter is outlined as follows. In the next Section, we describe the model problem and variational formulations for the Lagrange multiplier and Nitsche-based approaches. Section 2.3 introduces the discretized forms and discusses their numerical implementation. Particular emphasis is placed on the newly proposed basis for the Lagrange multiplier in three dimensions. We then present the results of several benchmark problems for three-dimensional embedded surfaces in Section 2.4. A discussion of the results follows in Section 2.5, followed by a summary and concluding remarks.

## 2.2 Model Problem and Variational Formulation

As a model problem, we consider an idealized diffusion problem defined on  $\Omega$ , a bounded domain of  $\mathbb{R}^3$ , that is partitioned into two domains,  $\Omega^1$  and  $\Omega^2$ , by an embedded surface  $\Gamma^*$ , as depicted in Figure 2.1(a). Both  $\Omega^1$  and  $\Omega^2$  contain a homogeneous and an isotropic material and have unitary and null conductivity respectively, thus leading to an idealized one-sided problem. The unit normal  $\mathbf{n}$  to  $\Gamma^*$  is chosen to point outward of  $\Omega^1$ . We write  $u$  for the primal variable of this problem in  $\Omega^1$ . It can be seen as representing the hydraulic head or temperature if flow in porous media or thermal problems were studied, respectively.

Dirichlet boundary conditions are applied such that  $u = u_0$  on the embedded surface  $\Gamma^*$ , and such that  $u = \bar{u}$  on  $\Gamma_D$ , a portion of the outer surface,  $\Gamma$ , of  $\Omega$ . The complementary part of  $\Gamma$  is flux-free. The primal variable  $u$  is defined in  $\mathbb{U}$  and its

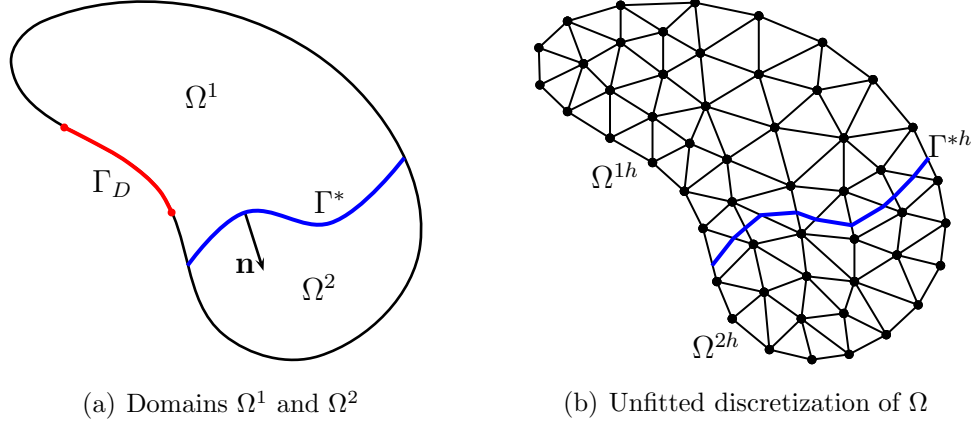


FIGURE 2.1 – The domain  $\Omega$  is divided in two sub-domains,  $\Omega^1$  and  $\Omega^2$ , by an embedded surface  $\Gamma^*$ . It is later discretized with a mesh which does not conform with the surface resulting in the two discretized sub-domains  $\Omega^{1h}$  and  $\Omega^{2h}$ . The embedded discretized surface  $\Gamma^{*h}$  is defined by the zero-level set of an approximate signed distance function to  $\Gamma^*$ .

variation  $\delta u$  is an element of  $\mathbb{U}_0$ :

$$\mathbb{U} = \{u \in H^1(\Omega), u = \bar{u} \text{ on } \Gamma_D\}$$

$$\mathbb{U}_0 = \{u \in H^1(\Omega), u = 0 \text{ on } \Gamma_D\}$$

The potential energy of such a system subjected to a body force  $\bar{f}$  can be constructed by subtracting the work done by the body force from the internal potential energy of the system  $\Pi^{\text{int}}$ :

$$\Pi(u) = \Pi^{\text{int}}(u) - \Pi^{\text{ext}}(u) = \frac{1}{2} \int_{\Omega} \nabla u \cdot \nabla u \, d\Omega - \int_{\Omega} \bar{f} u \, d\Omega \quad (2.1)$$

From a variational perspective, the solution  $u$  minimizes this potential energy under the specified Dirichlet constraints. Clearly, the definitions of the solution space and its variations account for the Dirichlet constraint on the external boundary. Therefore, we emphasize on the ways to treat the constraint on the embedded surface  $\Gamma_*$ .

### 2.2.1 Variational Methods for Constraint Enforcement

The problem described above can be seen as a constrained minimization problem where we seek to minimize the potential energy of the system subject to the constraint on the embedded surface. A standard way to transform such a constrained minimization problem into an unconstrained one is through the use of Lagrange multipliers. In order to enforce the constraint, the Lagrangian of the system,  $\mathcal{L}$ , may be built by adding the work of the Lagrange multipliers,  $\lambda$  in  $\mathbb{L} = H^{-1/2}(\Gamma^*)$ , to the potential energy of the system:

$$\mathcal{L}(u, \lambda) = \Pi(u) + \int_{\Gamma^*} \lambda(u - u_0) \, d\Gamma \quad (2.2)$$

The stationarity of  $\mathcal{L}$  yields a dual variational formulation: for all  $(\delta u, \delta \lambda) \in \mathbb{U}_0 \times \mathbb{L}$ , find  $(u, \lambda) \in \mathbb{U} \times \mathbb{L}$ , such that:

$$\delta \mathcal{L} = \int_{\Omega} \nabla \delta u \cdot \nabla u \, d\Omega - \int_{\Omega} \delta u \bar{f} \, d\Omega + \int_{\Gamma^*} \delta u \lambda \, d\Gamma + \int_{\Gamma^*} \delta \lambda (u - u_0) \, d\Gamma = 0 \quad (2.3)$$

Standard arguments regarding the arbitrariness of  $\delta u$  and  $\delta \lambda$  result in the well known mixed system of equations. When discretizing (2.3), it is well-established (see Babuška, 1973; Brezzi, 1974) that  $u$  and  $\lambda$  cannot be approximated independently. In order for the discretized form to remain stable, the choice of interpolation for both the primal and the dual variables has to satisfy the inf-sup condition.

### 2.2.2 Augmented Lagrangian and Nitsche's method

An alternative way to circumvent the stability issues in a Lagrange multiplier method applied to a constrained minimization problem is to stabilize the multiplier with a penalty term. An augmented Lagrangian formulation follows from adding this new

term to the initial Lagrangian of the system:

$$\mathcal{L}^{\text{aug}}(u, \lambda) = \Pi(u) + \int_{\Gamma^*} \lambda(u - u_0) \, d\Gamma + \int_{\Gamma^*} \frac{\alpha}{2}(u - u_0)^2 \, d\Gamma \quad (2.4)$$

The free parameter  $\alpha$  can be seen as a stabilization parameter which unlike a penalty parameter of the classical penalty methods is not required to tend to a very large value in order to asymptotically enforce the Dirichlet constraint. The stationarity of  $\mathcal{L}^{\text{aug}}$  leads to the dual variational form: for all  $(\delta u, \delta \lambda) \in \mathbb{U}_0 \times \mathbb{L}$ , find  $(u, \lambda) \in \mathbb{U} \times \mathbb{L}$ , such that:

$$\delta \mathcal{L}^{\text{aug}} = \delta \Pi + \int_{\Gamma^*} \delta u (\lambda + \alpha(u - u_0)) \, d\Gamma + \int_{\Gamma^*} \delta \lambda (u - u_0) \, d\Gamma = 0 \quad (2.5)$$

As pointed out in Wriggers and Zavarise (2008) in the case of frictionless contact, Nitsche's variational form can be related to the augmented Lagrangian one. On sequentially using integration by parts and Green's theorem on the first variation of  $\Pi^{\text{int}}$  and then grouping together the terms on the boundary, one obtains: for all  $(\delta u, \delta \lambda) \in \mathbb{U}_0 \times \mathbb{L}$ :

$$\int_{\Gamma^*} \delta u (\nabla u \cdot \mathbf{n} + \lambda + \alpha(u - u_0)) \, d\Gamma + \int_{\Gamma^*} \delta \lambda (u - u_0) \, d\Gamma = 0 \quad (2.6)$$

On using the arguments related to the arbitrariness of  $\delta u$  and  $\delta \lambda$  in (2.6), we ensure the satisfaction of the Dirichlet condition  $u = u_0$  on  $\Gamma^*$  as well as the equivalence between the multiplier and the flux on the interface such that  $\lambda = -\nabla u \cdot \mathbf{n}$ .

Now if we use this equivalence and simply replace the Lagrange multiplier with the negative of the flux in the augmented Lagrangian potential (2.5), we obtain the potential function originally proposed by Nitsche:

$$\Pi^{\text{Nit}}(u) = \Pi(u) - \int_{\Gamma^*} (u - u_0) \nabla u \cdot \mathbf{n} \, d\Gamma + \int_{\Gamma^*} \frac{\alpha}{2}(u - u_0)^2 \, d\Gamma \quad (2.7)$$

The stationarity of  $\Pi^{\text{Nit}}$  leads to a one-field symmetric variational formulation: for all  $\delta u \in \mathbb{U}_0$ , find  $u \in \mathbb{U}$ , such that:

$$\delta \Pi^{\text{Nit}} = \delta \Pi - \int_{\Gamma^*} \delta u \nabla u \cdot \mathbf{n} \, d\Gamma - \int_{\Gamma^*} (u - u_0) \nabla \delta u \cdot \mathbf{n} \, d\Gamma + \int_{\Gamma^*} \delta u \alpha (u - u_0) \, d\Gamma = 0 \quad (2.8)$$

## 2.3 Discretization and Implementation

In this section, we describe both the discretization and the implementation of Nitsche's method and the stable Lagrange multiplier method based on the vital vertex algorithm. We first deal with the discretization of the bulk field that is used for both methods. We then focus on the details regarding the implementational aspects for each of them separately.

### 2.3.1 Discretization of the bulk field

The domain  $\Omega$  is discretized using standard non-overlapping linear tetrahedra and the surface  $\Gamma^*$  is embedded in the resulting mesh as shown in the Figure 2.1(b). The embedded surface is located by the zero level-set of  $\psi(\mathbf{x})$  which specifies a signed distance to the surface from any nodal position  $\mathbf{x}$  (see Osher and Fedkiw, 2002). If the value of the level-set is positive at a node, this node is inside  $\Omega^1$ , otherwise it belongs to  $\Omega^2$  (see Figure 2.2(a)). To capture the local nature of the solution in the vicinity of the interface, we modify the kinematics of cut elements. Following the partition of unity method proposed in Babuška and Melenk (1997), the shape functions,  $\tilde{N}$ , are computed as the product of the standard Finite Element shape functions,  $N$ , multiplied by a characteristic function  $H(\mathbf{x})$ :

$$H(\mathbf{x}) = \begin{cases} 1 & \text{if } \mathbf{x} \in \Omega^1 \\ 0 & \text{otherwise} \end{cases} \quad (2.9)$$

This technique, referred in the literature as eXtended Finite Element method, XFEM (see Moës et al., 1999; Dolbow et al., 2000; Hansbo and Hansbo, 2002), allows



for the representation of elements partially filled with matter as described in Figure 2.2(b).

We introduce  $\Gamma^{*h}$  as the discretized embedded surface. This surface is constructed in the following manner. First, we calculate the signed distance function  $\psi$  at each of the nodes in the mesh, and gather them in an array  $\Psi$ . Then, an approximate signed distance function is constructed with the same shape functions of the background mesh via

$$\psi \approx \psi^h = N^T \Psi. \quad (2.10)$$

The discretized surface  $\Gamma^{*h}$  is then given by the zero-level set of  $\psi^h$ . With linear tetrahedra, this process yields a piecewise-planar approximation of  $\Gamma^*$  resulting in either triangular (see Figure 2.2(c)) or quadrilateral surfaces  $\Gamma_e^*$  in each element. Note that only degrees of freedom associated with elements cut by the embedded

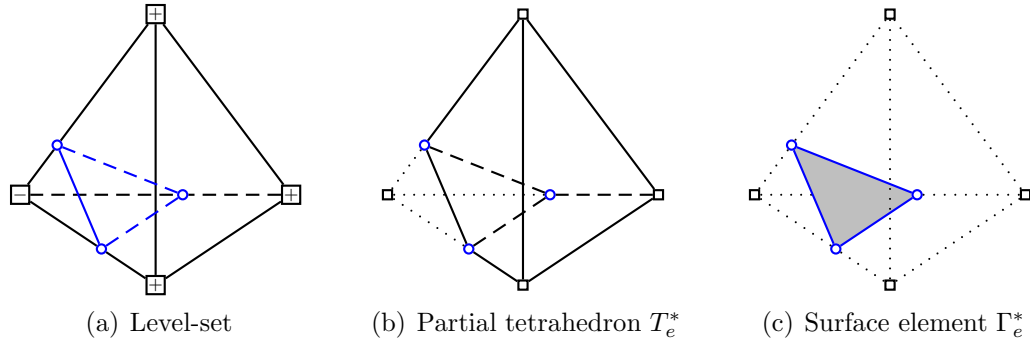


FIGURE 2.2 – Representation of partial elements using X-FEM. The black squares are the nodes of the bulk mesh and the blue circles are the vertices of the mesh of the surface.

surface are affected by the Heaviside function. Note as well that, for a one-sided problem, the degrees of freedom belonging to elements which are completely in the void phase ( $\Omega^2$ ) – are not accounted for in the assembly procedure.

### 2.3.2 Implementation of Nitsche's method

This subsection is devoted to the implementation of Nitsche's variational form (2.8). To that end, we introduce the following discretized approximations of the primal variable and its variation respectively:

$$u \approx u^h = \tilde{N}^T U \quad \delta u \approx \delta u^h = \tilde{N}^T W \quad (2.11)$$

where  $U$  and  $W$  are the vectors of nodal values of the primal variable and its variation respectively. The gradient of the modified shape function  $\tilde{N}$  is denoted by  $\tilde{B}$ .

#### *The three terms of the stiffness matrix*

Introducing the aforementioned approximations (2.11) in Nitsche's variational form (2.8) yields the following matrix system:

$$(K_b - K_n + K_s)U = f_b - f_n + f_s \quad (2.12)$$

where  $K_b$ ,  $K_n$  and  $K_s$  are the bulk contribution, the flux contribution and the stabilization contribution to the global stiffness matrix:

$$K_b = \int_{\Omega^h} \tilde{B} \tilde{B}^T d\Omega \quad (2.13)$$

$$K_n = \int_{\Gamma^{*h}} \tilde{N} \mathbf{n}^T \tilde{B}^T d\Gamma + \int_{\Gamma^{*h}} \tilde{B} \mathbf{n} \tilde{N}^T d\Gamma \quad (2.14)$$

$$K_s = \int_{\Gamma^{*h}} \tilde{N} \alpha \tilde{N}^T d\Gamma \quad (2.15)$$

and  $f_b$ ,  $f_n$  and  $f_s$  are the bulk contribution, the flux contribution and the stabilization contribution to the right side of the system:

$$f_b = \int_{\Omega^h} \tilde{N} \bar{f} \, d\Omega \quad (2.16)$$

$$f_n = \int_{\Gamma^{*h}} \tilde{B} \mathbf{n} u_0 \, d\Gamma \quad (2.17)$$

$$f_s = \int_{\Gamma^{*h}} \tilde{N} \alpha \tilde{N}^T \, d\Gamma \quad (2.18)$$

*Choice of the stabilization parameter*

The choice of stabilization parameter is important for two reasons. Firstly, in order to solve (2.12) using the large amount of effective solvers designed for symmetric positive definite systems, such as preconditioned conjugate gradient, the stabilization parameter needs to be large enough to ensure the coercivity of the bilinear form. More importantly, coercivity of the bilinear form allows us to use Lax-Milgram type arguments to establish existence, uniqueness and convergence. For the model problem considered, the discretized bilinear form is given by:

$$a(u^h, v^h) = \int_{\Omega^h} \nabla v^h \cdot \nabla u^h \, d\Omega - \int_{\Gamma^{*h}} v^h \nabla u^h \cdot \mathbf{n} \, d\Gamma - \int_{\Gamma^{*h}} u^h \nabla v^h \cdot \mathbf{n} \, d\Gamma + \alpha \int_{\Gamma^{*h}} v^h u^h \, d\Gamma \quad (2.19)$$

We can now establish a lower bound on the parameter  $\alpha$  by finding the smallest convenient choice that ensures coercivity of  $a(v^h, v^h)$  for all  $v^h \in \mathbb{U}_0^h$ , where  $\mathbb{U}_0^h$  is a finite dimensional approximation to  $\mathbb{U}_0$ . We introduce, for all  $v^h \in H^1(\Omega^h)$  and  $\mathcal{D}$  a subset of  $\Omega^h$ , the energy seminorm  $\|v^h\|_{E, \mathcal{D}}$ , the  $L_2$  norm  $\|v^h\|_{L_2, \mathcal{D}}$  and, for all

$t \in H^{-1/2}(\Gamma_h^*)$ , the duality pairing  $\langle v^h, t \rangle$  on  $\Gamma_h^*$  as:

$$\|v^h\|_{E,\mathcal{D}}^2 = \int_{\mathcal{D}} \nabla v^h \cdot \nabla v^h \, d\mathcal{D}, \quad \|v^h\|_{L_2,\mathcal{D}}^2 = \int_{\mathcal{D}} v^h v^h \, d\mathcal{D}, \quad \langle v^h, t \rangle = \int_{\Gamma^{*h}} v^h t \, d\Gamma. \quad (2.20)$$

Following Dolbow and Harari (2009), there exists a configuration-dependent positive constant  $C_1$  such that the following inequality holds:

$$\|\nabla v^h \cdot \mathbf{n}\|_{L_2,\Gamma^{*h}} \leq C_1 \|v^h\|_{E,\Omega^h} \quad (2.21)$$

By applying the Cauchy-Schwarz inequality to  $\langle v^h, \nabla v^h \cdot \mathbf{n} \rangle$  in the expression of  $a(v^h, v^h)$  and employing the previous inequality (2.21), it follows that:

$$\begin{aligned} a(v^h, v^h) &\geq \|v^h\|_{E,\Omega^h}^2 - 2C_1 \|v^h\|_{L_2,\Gamma^{*h}} \|v^h\|_{E,\Omega^h} + \alpha \|v^h\|_{L_2,\Gamma^{*h}}^2 \\ &\geq (\|v^h\|_{E,\Omega^h} - C_1 \|v^h\|_{L_2,\Gamma^{*h}})^2 + (\alpha - C_1^2) \|v^h\|_{L_2,\Gamma^{*h}}^2 \end{aligned} \quad (2.22)$$

Although  $C_1$  may be computed globally (see Griebel and Alexander, 2002; Sanders et al., 2009), for problems with evolving interfaces this necessitates the solution of a large eigenvalue problem at every time step and thus adds significantly to the computational expense.

A more efficient approach as advocated in Dolbow and Harari (2009) is to choose a stabilization parameter  $\alpha_e$  in each partial element  $T_e^*$  greater than a constant  $C_1^e$  computed in each of these elements according to (2.21). For linear tetrahedral elements, fluxes are constant in each element, thus leading to the following equalities:

$$\|\nabla v^h \cdot \mathbf{n}\|_{L_2,\Gamma_e^*}^2 = \text{meas}(\Gamma_e^*) |\nabla v^h \cdot \mathbf{n}|^2 \quad (2.23)$$

$$\|v^h\|_{E,T_e^*}^2 = \text{meas}(T_e^*) \|\nabla v^h\|^2 \quad (2.24)$$

Hence,  $C_1^e$  can be given a lower bound in terms of the physical volume of the partial tetrahedron  $T_e^*$  and the area of the embedded surface  $\Gamma_e^*$ , depicted in Figure 2.2(c), as  $|\nabla v^h \cdot \mathbf{n}|^2 \leq \|\nabla v^h\|^2$ :

$$\alpha_e \geq C_1^{e2} \geq \frac{\text{meas}(\Gamma_e^*)}{\text{meas}(T_e^*)} \quad (2.25)$$

We thus obtain a simple algebraic expression for the stabilization parameter. For the current study,  $\alpha_e$  is chosen as  $2C_1^e{}^2$ . The factor of 2 is chosen to obtain good performance in computations and is based on previous numerical experience and parametric studies (Embar et al., 2010). For higher order elements, we no longer have a piecewise constant flux in each element so analytical expressions akin to (2.25) for  $C_1^e$  cannot be obtained. However, we can estimate lower bounds for  $C_1^e$  through a series of local eigenvalue problems.

**Remark:** Clearly, from (2.25), as  $\text{meas}(T_e^*) \rightarrow 0$ , the stabilization parameter becomes unbounded. Similar issues were reported by Lew and Negri (2011) where they showed sub-optimal convergence behavior for pathological cases. While we did not encounter sub-optimal convergence behavior in  $L_2$  or  $H_1$  norms in any of the numerical examples reported in Section 2.4, we did see a poor approximation to the flux in the  $L_\infty$  norm. We will discuss this issue further in Section 2.4. For the sake of robustness, techniques which snap the surface to a vertex when it is too close (as in Lew and Buscaglia, 2008) or employ other tolerancing schemes as in Embar et al. (2010) are useful for such cases.

*Estimation of the flux with domain integrals*

Fluxes may be directly evaluated on the embedded surface by multiplying the nodal values of the bulk field by  $\tilde{B}$ , the gradient of the shape functions. However, such a method could yield inaccurate results (Mourad et al., 2007; Dolbow and Franca, 2008). We briefly recall here an alternative method using domain integrals introduced in Ji and Dolbow (2004) as a generalization of the work originally proposed in Carey et al. (1985). By projecting the local equilibrium of the Poisson's problem on any

test function  $v \in \mathbb{U}_0$  and by using integration by parts, it holds that:

$$\int_{\Gamma^*} v \nabla u \cdot \mathbf{n} \, d\Gamma = \int_{\Omega} \nabla v \cdot \nabla u \, d\Omega - \int_{\Omega} v \bar{f} \, d\Omega \quad (2.26)$$

We write  $\phi^h$  for the discrete approximation of the flux on the embedded surface and  $\Phi$  the vector at each bulk node  $I$  of elements cut by the embedded surface:

$$\nabla u \cdot \mathbf{n} \approx \phi^h = \tilde{N}^T \Phi. \quad (2.27)$$

$\phi^h$  may be interpolated by any given shape functions defined on this set of nodes. We choose the most convenient one: the readily available bulk shape functions  $\tilde{N}$

The discretized form of (2.26) can be expressed in terms of the bulk contribution to the stiffness matrix and the right side, the primal variable and an integral of the outer product of the bulk shape functions on the embedded surface:

$$\int_{\Gamma^{*h}} \tilde{N} \tilde{N}^T \, d\Gamma \, \Phi = K_b U - f_b. \quad (2.28)$$

The resulting matrix has a structure very similar to  $K_s$  (2.15). In order to simplify the resolution of the previous system, the matrix of the left term may be lumped. The resulting diagonal coefficients may be simplified as the bulk shape functions form a partition of unity. At each bulk node  $I$  of an element cut by the embedded surface, the flux  $\Phi_I$  can thus be directly evaluated as:

$$\Phi_I = \frac{1}{\int_{\Gamma^{*h}} \tilde{N}_I \, d\Gamma} [K_b U - f_b]_I \quad (2.29)$$

The domain integral method thus comes down to a simple post-treatment of the bulk quantities in the vicinity of  $\Gamma^*$ . The resulting flux is known at the bulk nodes of the cut elements and may then be interpolated using the bulk shape functions. It is worth noticing that this method is totally independent from any of the Nitsche specific terms and may also be applied for a Lagrange multiplier method.

### 2.3.3 Discretization of the Lagrange multiplier

The first step in building a stable Lagrange multiplier basis is to select a subset of vertices to prevent over-constraining the system. By a vertex, we refer to an intersection point between the bulk mesh and the embedded surface. We follow the vital vertex algorithm introduced in Béchet et al. (2009) and then build an interpolation for the Lagrange multipliers at each of these locations. All of these interpolating shape functions have to comply with the partition of unity on  $\Gamma^*$ . This condition may be enforced globally as proposed in Béchet et al. (2009) or locally in each element as described here.

#### *Free basis of Lagrange multipliers with the vital vertex algorithm*

The vital vertex algorithm is a method to coarsen the grid of vertices on the embedded surface. It relies on graph theory to do so. A vertex graph, distinct from the interfacial mesh, is built between the vertices and then marched in order to identify among all of them, a subset of vital vertices.

In order to build the vertex graph, a criterion needs to be given to specify how vertices are connected with each other. Two vertices are connected in the vertex graph when each of them belongs to an edge of the bulk mesh that meet at a node of the bulk mesh. Such a node is referred to as an end-point. In Figure 2.3(a), end-points are represented as black squares whereas nodes that are not end-points are represented as white squares. In the same figure, one may see that vertices  $\textcircled{b}$  and  $\textcircled{g}$  are connected according to the aforementioned criterion because they respectively belong to edges that meet at  $\boxed{7}$ . The vertex graph corresponding to the case depicted in Figure 2.3(a) is presented in Figure 2.3(b). Algorithm 1 details the pseudo-code to build the vertex graph.

Once the vertex graph has been built, we need to select a subset of vertices and mark them as vital. These vital vertices now represent the vector of unknowns

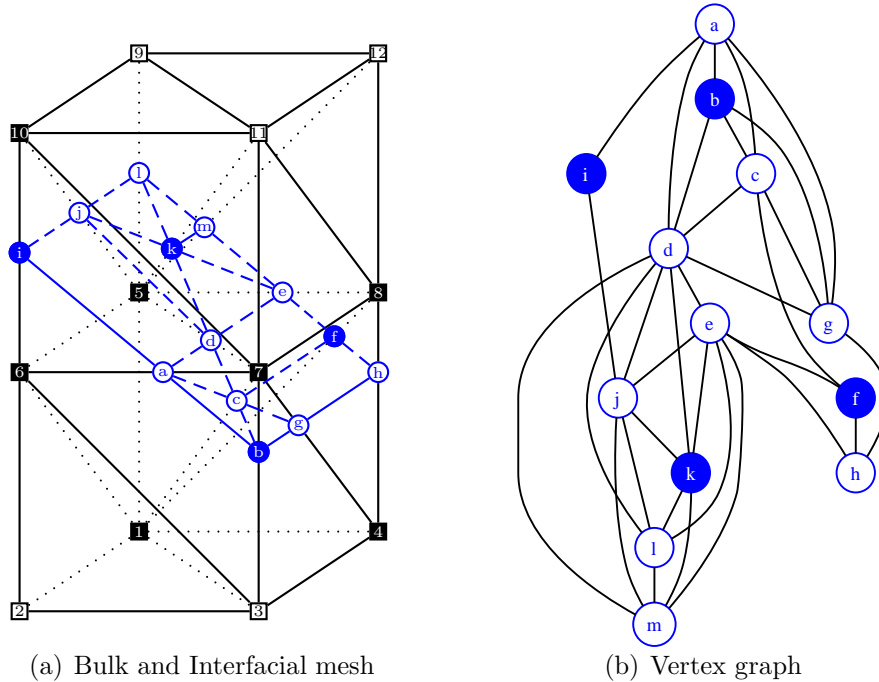


FIGURE 2.3 – Simple example of two superposed cubes, each of them being meshed by 6 tetrahedra, cut by an embedded surface. Black squares are end-points and white squares are non-end-points. The circles are the vertices of the interfacial mesh, intersection of the bulk mesh  $\Gamma^*$ . Blue circles are vital vertices and white ones are non-vital vertices. The vertex graph is generated with Graphviz (Gansner and North, 2000).

for Lagrange multipliers and are identified by marching the graph and sequentially testing the following two rules:

**Rule 1:** Two vital vertices cannot be connected in the vertex graph

**Rule 2:** Each non-vital vertex is connected to at least one vital vertex.

The first rule ensures that the graph-distance between any two vital vertices is greater than one. The second rule makes sure that this graph-distance is equal to two or three. Indeed, any vertex can be connected to a non-vital vertex. However, this non-vital vertex has to be connected to at least one vital vertex according to the second rule. Therefore, the graph-distance between any two vital vertices is at most three. A simple way of interpreting these rules is to think of the first rule as a criterion to prevent an over-constraining of the system. At the same time, the second rule



---

**Algorithm 1** Building the vertex graph

---

Inputs:  $\star$  edge connectivity of the bulk mesh  
           $\star$  list of vertices indexed by the edge they belong to

1. Identify the end-points, bulk nodes that are connected to more than 2 edges containing a vertex
2. Map vertices to end-points and end-points to vertices
3. Draw the vertex connectivity

```
for all vertices  $v$  do  
  for all end-points  $ep$  mapped to  $v$  do  
    for all vertices  $w$  mapped to  $ep$  do  
      link  $v$  to  $w$  in the vertex graph  
    end for  
  end for  
end for
```

---

ensures that the coarsening does not result in an obviously underconstrained system.

The procedure presented thus far results in several sets of vital vertices, each of them with a different cardinality. We choose a set of maximum cardinality from all permissible sets which further ensures that the constraint is implemented well. The following criteria define a total order on the vertices and enforce that the graph is only walked once:

**Criterion 1:** Vertices are sorted according to their number of connections.

**Criterion 2:** Equally connected vertices are sorted according to their indices.

According to these two criteria, ① is the smallest vertex, thus the one to start with and ④ is the biggest one, thus the last one. ⑥ and ⑧ are both connected to three other vertices and in lexicographic order ⑥ precedes ⑧, thus ⑥ is smaller than ⑧.

For the case depicted in Figure 2.3(a), the vital vertices are represented by blue circles in Figure 2.3(b). The following pseudo code (see Algorithm 2) sums up the search procedure of the vital vertex within the vertex graph.

---

**Algorithm 2** Finding the vital vertices in the vertex graph

---

1. Compute a lexicographic order for the vertex of the graph according to their number of connections and their index
  2. March the sorted graph and identify the vital vertices
- for all** vertices  $v$  in the sorted vertex graph **do**  
  **if**  $v$  is connected to at least a vital vertex **then**  
     $v$  is not a vital vertex  
  **else**  
     $v$  is a vital vertex  
  **end if**  
**end for**
- 

*Interpolation of the Lagrange multipliers consistent with the partition of unity*

A shape function has to be built at each vital vertex location for the corresponding Lagrange multipliers. In 2D for a straight surface, a straightforward choice is to use standard hat functions defined on the segment joining two neighboring vital vertices (see Moës et al., 2006). However, for more complex surfaces and in 3D, this technique implies that the embedded surface should be re-meshed keeping only the vital vertices. Even if a Delaunay algorithm can achieve this triangulation, evaluating the product of shape functions from the bulk mesh with these hat functions defined on this triangulation is quite challenging considering the non-local support of these hat functions. An alternative strategy which simplifies the implementation a great deal would be to keep the surface mesh as the intersection of the bulk mesh with the embedded surface and to build the shape functions of the multipliers from the bulk shape functions naturally defined on this mesh. The method introduced in Béchet et al. (2009) thus computes the shape function of the Lagrange multiplier at vital vertex  $v$ ,  $N_v^\lambda$ , as a weighted sum of traces onto the embedded surface,  $\Gamma^*$ , of bulk shape functions:

$$N_v^\lambda = \sum_n \omega_n^v \mathbf{T}_{\Gamma^*}(\tilde{N}_n) \quad (2.30)$$

The weights,  $\omega_n^v$ , should be chosen in order for the shape functions of the Lagrange

multipliers to satisfy the partition of unity on  $\Gamma^*$ . By enforcing the partition of unity for all the  $N_v^\lambda$  at the global level, thus using the fact that the bulk shape functions form globally a partition of unity, the interfacial shape functions inherit the smoothness of the bulk shape functions. However such a  $C_0$ -continuous interpolation for the Lagrange multipliers is not a convenient choice from an implementational perspective. Any choice of weights,  $\omega_n^v$ , that enforces the partition of unity of the  $N^\lambda$  locally in each element, results in a basis of piecewise linear shape functions on  $\Gamma^*$ . Hence, keeping the computation of the weights,  $\omega_n^v$ , at the element level gives both more latitude in their choice and a more convenient implementation in a finite element context without introducing any theoretical inconsistency.

Knowing the support of each Lagrange multiplier shape function is of primary importance in order to compute the local weights,  $\omega_n^v$ , only where it is required. To that end, we introduce the one-ring of a node which consists of all elements that encompass the considered node in their connectivity list. The support of a Lagrange multiplier shape function is not restricted to the one-ring of its associated vital vertex as is usually the case with the finite element method. Indeed, each  $N_v^\lambda$  is defined over the intersection of the embedded surface with the one-ring of the end-points it is related to. We recall that an end-point is a node of the bulk mesh that meets at least two edges that have been cut by  $\Gamma^*$ , thus containing a vertex. For instance, Figure 2.4 shows the support of each of the Lagrange multiplier shape functions corresponding to the example described in Figure 2.3(a).

When computing the Lagrange multiplier shape functions,  $N_v^\lambda$ , at the vital vertex  $v$ , in an element  $\Gamma_e^*$  of the interfacial mesh, each weight  $\omega_n^v$  is related to the trace of one of the four shape functions of the cut tetrahedron of the bulk mesh. Each weight is computed according to the nature of the related bulk node. Bulk nodes may belong to four different categories. They can either be not an end-point, an end-point connected to  $v$  or an end-point not connected to  $v$ . The latter category

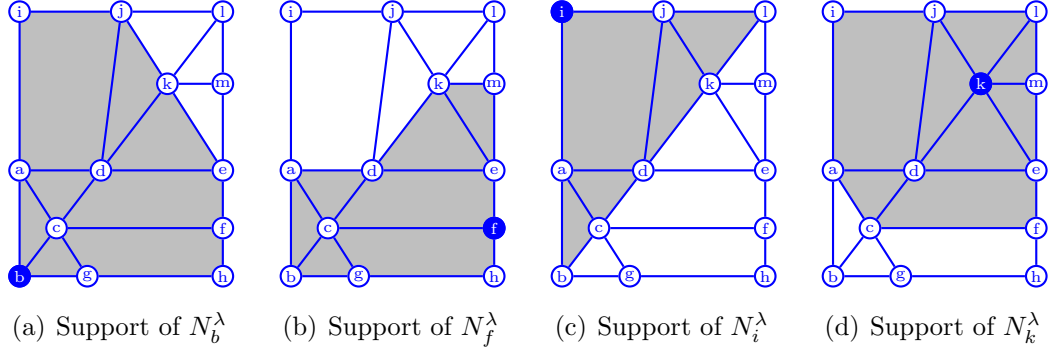


FIGURE 2.4 – Lagrange multiplier shape functions of the example described in Figure 2.3(a) are defined on the shaded areas which represent their respective support. The blue circles are the corresponding vital vertices.

encompasses two distinct categories owing to the fact that not all end-points may be connected to a vital vertex. For instance in Figure 2.3(a), the end-point  $\boxed{4}$  is not connected to any vital vertex. We thus introduce the notion of an active end-point: an active end-point is connected to at least one vital vertex whereas a non-active end-point is not connected to any vital vertex. Hence, any node of a bulk element cut by the embedded surface can either be an active end-point connected to  $v$ , an active end-point not connected to  $v$ , a non-active end-point or not an end-point at all.

The key idea introduced in Béchet et al. (2009) regarding the interpolation of the Lagrange multipliers with the trace of bulk shape functions is that the shape function of an active end-point only fully contributes to the Lagrange multiplier shape functions at the vital vertex it is connected to. Thus, the  $N_v^\lambda$ , shape functions of active end-points connected to  $v$  are given a unitary weight  $\omega_n^v$  whereas other active end-points are given a null weight. Bulk nodes of a tetrahedron  $e$  that are not active end-points should contribute to all the Lagrange multiplier shape functions that are defined on  $\Gamma_e^*$ . We write  $n_e$  for the number of Lagrange multiplier shape functions defined over  $\Gamma_e^*$ . In order for the Lagrange multiplier shape functions to form a partition of unity over  $\Gamma^*$ , a straightforward way to compute the weight related to

the trace of bulk shape functions of not active end-points is to consider it equal to  $1/n_e$ . Hence, in each element of the mesh of the embedded surface, the restriction of the Lagrange multiplier shape function  $N_v^\lambda|_{\Gamma_e^*}$  of the vital vertex  $v$  is computed as the weighted sum of the four bulk shape functions where the local weights  $\omega_n^{v,e}$  are computed as follows:

$$\omega_n^{v,e} = \begin{cases} 1 & \text{if } n \text{ is connected to } v \\ 0 & \text{if } n \text{ is an active end-point not connected to } v \\ \frac{1}{n_e} & \text{otherwise} \end{cases} \quad (2.31)$$

As a means of comparison between the shape functions of the Lagrange multipliers at vital vertices as introduced in Béchet et al. (2009) and the one proposed in the present contribution, we plot both alternatives in Figure 2.5 for the example depicted in Figure 2.3(a). In both cases, the shape functions look fairly similar except for the

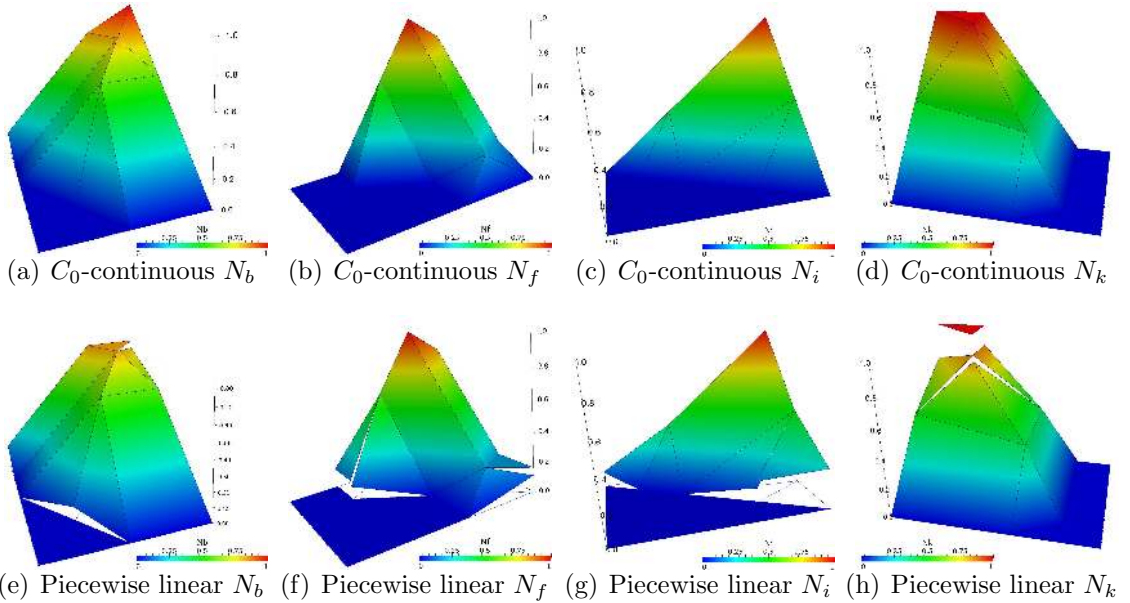


FIGURE 2.5 – Comparison between  $C_0$ -continuous, as proposed in Béchet et al. (2009), and piecewise continuous shape functions of the Lagrange multipliers defined at vital vertices for the example problem depicted in Figure 2.3(a).

fact that the shape functions introduced in this work are not  $C_0$ -continuous over their support but only piecewise  $C_0$ -continuous.

It bears emphasis that all the pieces of information required to compute the local weights for each of the Lagrange multiplier shape functions - namely: active end-points, non-active end-points and  $n_e$ , the number of shape functions defined on each element  $\Gamma_e^*$  of the embedded surface - may be computed on the fly while running the vital vertex algorithm providing added efficiency and simplicity to this approach.

#### 2.3.4 Implementation of the stable Lagrange multiplier method

In order to discretize the dual variational form (2.3), we introduce the following approximations of both the Lagrange multipliers and their variations in terms of the stable basis functions introduced in the previous subsection:

$$\lambda \approx \lambda^h = \Lambda^T N^\lambda \quad \delta\lambda \approx \delta\lambda^h = L^T N^\lambda \quad (2.32)$$

Together with the approximations of the bulk field introduced in (2.11), it is straightforward to derive the following saddle-point system:

$$\begin{bmatrix} K_b & G^T \\ G & 0 \end{bmatrix} \begin{bmatrix} U \\ \Lambda \end{bmatrix} = \begin{bmatrix} f_b \\ f_\lambda \end{bmatrix} \quad (2.33)$$

where  $G$  and  $f_\lambda$  are respectively the off-diagonal and complementary contribution to both the matrix and the right side:

$$G = \int_{\Gamma^{*h}} N^\lambda \tilde{N}^T \, d\Gamma \quad (2.34)$$

$$f_\lambda = \int_{\Gamma^{*h}} N^\lambda u_0 \, d\Gamma \quad (2.35)$$

and  $K_b$  and  $f_b$  are respectively the upper diagonal bulk contribution introduced in (2.13) and the bulk contribution to the right side introduced in (2.16).

Due to the specific expression of the shape functions of the Lagrange multipliers, the standard assembly procedure needs to be modified in order to compute  $G$  and

$f_\lambda$ . Algorithm 3 presents a pseudo-code of the strategy we implemented in order to compute  $f_\lambda$ . It is straightforward to extend this procedure to  $G$  using another loop on the bulk nodes of elements cut by the embedded surface. In Algorithm 3, non-active end-points are being looped over several times for a vital vertex connected to more than one end-point. Therefore, their weights should be divided by  $n_v$ , the number of end-points that are connected to the vital vertex  $v$ .

---

**Algorithm 3** Pseudo-code to compute  $[f_\lambda]_v$ , the  $v^{th}$  component of  $f_\lambda$

---

```

 $[f_\lambda]_v = 0$ 
for all end-point  $e_p$  connected to the vital vertex  $v$  do
  for all bulk element  $e$  in the one-ring of  $e_p$  do
    for all node  $n$  of element  $e$  do
      if  $n$  is not an active end-point then
         $[f_\lambda]_{v+} = \frac{1}{n_v} \int_{\Gamma_e^*} \omega_n^{v,e} T_{\Gamma_e^*}(\tilde{N}_n) u_0 \, d\Gamma$ 
      else
         $[f_\lambda]_{v+} = \int_{\Gamma_e^*} \omega_n^{v,e} T_{\Gamma_e^*}(\tilde{N}_n) u_0 \, d\Gamma$ 
      end if
    end for
  end for
end for

```

---

## 2.4 Numerical Examples

In this section, we present numerical results obtained for several three-dimensional benchmark problems with Dirichlet constraints on an embedded surface treated with Nitsche's method and the stable Lagrange multiplier method based on the vital vertex algorithm (LMVV).

The main focus of the present numerical study is to investigate the relative performance of the two concurrent methods in approximating the bulk and interfacial fields by means of convergence studies. To that end, we use several relative error norms for the bulk field and the flux. The accuracy in the bulk field is evaluated using standard  $L_2$  and  $H_1$  error norms computed in terms of the  $L_2$  and energy norm

introduced in (2.20):

$$\mathcal{E}_{L_2}(u)|_{\Omega^1} = \frac{\|u^h - u\|_{L_2, \Omega^1}}{\|u\|_{L_2, \Omega^1}} \quad (2.36)$$

$$\mathcal{E}_{H_1}(u)|_{\Omega^1} = \left( \frac{\|u^h - u\|_{L_2, \Omega^1}^2 + \|u^h - u\|_{E, \Omega^1}^2}{\|u\|_{L_2, \Omega^1}^2 + \|u\|_{E, \Omega^1}^2} \right)^{\frac{1}{2}} \quad (2.37)$$

The accuracy in the bulk field on the boundary as well as the accuracy in the flux are examined using the  $L_2$  error norm on the surface:

$$\mathcal{E}_{L_2}(u)|_{\Gamma^*} = \frac{\|u^h - u\|_{L_2, \Gamma^*}}{\|u\|_{L_2, \Gamma^*}} \quad (2.38)$$

$$\mathcal{E}_{L_2}(\nabla u \cdot \mathbf{n})|_{\Gamma^*} = \frac{\|\phi^h - \nabla u \cdot \mathbf{n}\|_{L_2, \Gamma^*}}{\|\nabla u \cdot \mathbf{n}\|_{L_2, \Gamma^*}} \quad (2.39)$$

where  $\phi^h$  is an approximation of the flux on the embedded surface which can either be constructed from Lagrange multipliers or evaluated either directly or using the domain integral method for Nitsche's method. These error norms are computed using a  $4 \times 4 \times 4$  point Gauss quadrature in the bulk or, on the embedded surface, using a three-point Gauss quadrature for a triangular surface patch or  $2 \times 2$  point Gauss quadrature for a quadrilateral patch.

For comparison with methods that are not based on a variational principle such as the finite difference method, we also present results showing convergence in the maximum nodal error or the sup-norm for both the bulk field and the flux:

$$\mathcal{E}_{\infty}(u)|_{\Omega^1} = \max_{\mathbf{x}_i \in \Omega^1} (|u_i^h - u(\mathbf{x}_i)|) \quad (2.40)$$

$$\mathcal{E}_{\infty}(\nabla u \cdot \mathbf{n})|_{\Gamma^*} = \max_{\mathbf{x}_i \in \Gamma^*} (|\phi_i^h - \nabla u(\mathbf{x}_i) \cdot \mathbf{n}|) \quad (2.41)$$

Contour plots of the bulk field and the flux are also provided where illustrative.



### 2.4.1 Triple cosine field

As a first example, we consider the following one-sided problem defined in  $\Omega = (0, 1) \times (0, 1) \times (0, 1)$  divided by the embedded surface  $\Gamma^*$  into  $\Omega^1$  and  $\Omega^2$  by:

$$\begin{aligned}
 \Delta u &= \pi^2 \cos(\pi x) \cos(\pi y) \cos(\pi z) & \text{in } \Omega^1 &= \{\mathbf{x} : \psi(\mathbf{x}) > 0\} \\
 u &= -\cos(\pi x) \cos(\pi y) & \text{on } \Gamma_D &= \{\mathbf{x} : z = 1\} \\
 u &= \cos(\pi x) \cos(\pi y) \cos(\pi z_*) & \text{on } \Gamma^* &= \{\mathbf{x} : \psi(\mathbf{x}) = 0\} \\
 \nabla u \cdot \mathbf{n} &= 0 & \text{on } \Gamma_N &= \{\mathbf{x} : x = 0 \mid y = 0 \mid \psi(\mathbf{x}) > 0\}
 \end{aligned} \tag{2.42}$$

The problem has an analytical solution given by  $u = \cos(\pi x) \cos(\pi y) \cos(\pi z)$ .

#### *Planar embedded surface*

To begin with, we consider an arbitrarily chosen planar embedded Dirichlet surface  $\psi(\mathbf{x}) = 0.2x - 0.2y + z + 0.4856$ . For the convergence study, we report our results on a sequence of six unstructured meshes. The results of the convergence study are shown in Figure 2.7. We report optimal rates of convergence for both the methods in the bulk field as well as the flux. The errors in the bulk field calculated using Nitsche's method and LMVV are nearly indistinguishable in both the  $L_2$  norm as well as the  $H_1$  norm. We thus only show the approximation to the bulk solution computed with Nitsche's method in Figure 2.6(a). For the flux, we report much better accuracy using Nitsche's method as opposed to LMVV. However, on computing the flux by applying the domain integral smoothing technique to LMVV, we obtain a much improved accuracy, comparable to Nitsche's method, with the stable Lagrange multiplier method as well. The approximation to the flux calculated using Nitsche's method and LMVV are shown in Figures 2.6(c) and 2.6(e). Due to the specific feature of the chosen Lagrange multiplier shape functions that have a support larger than the one-ring of the vertex they are built on, the surface has been remeshed retaining the vital vertices alone for visualizing the contour plots. The remeshing

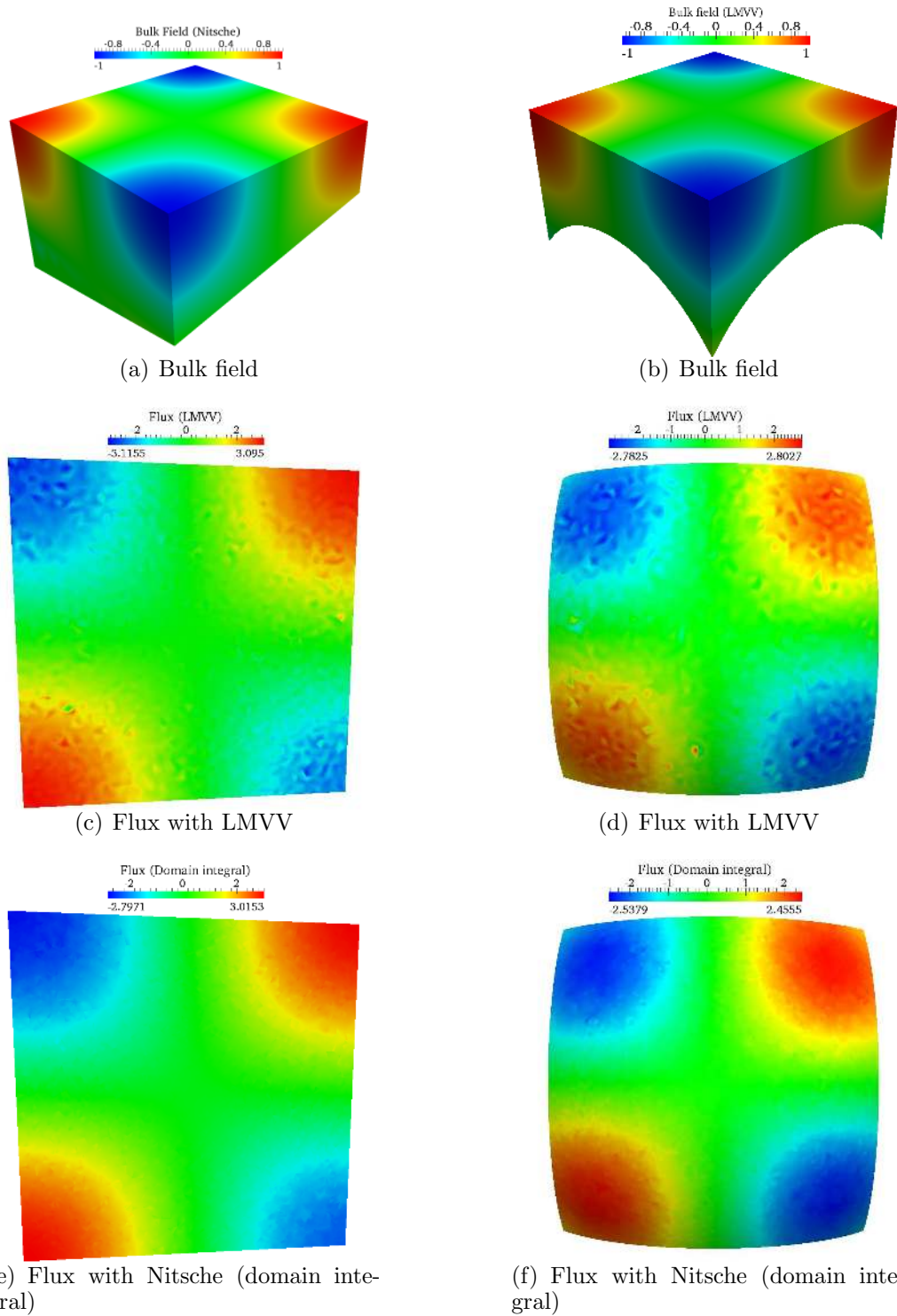


FIGURE 2.6 – Finite element approximation of the bulk and the flux for the triple cosine problem (2.42) with a planar (left) and a spherical (right) embedded surface.

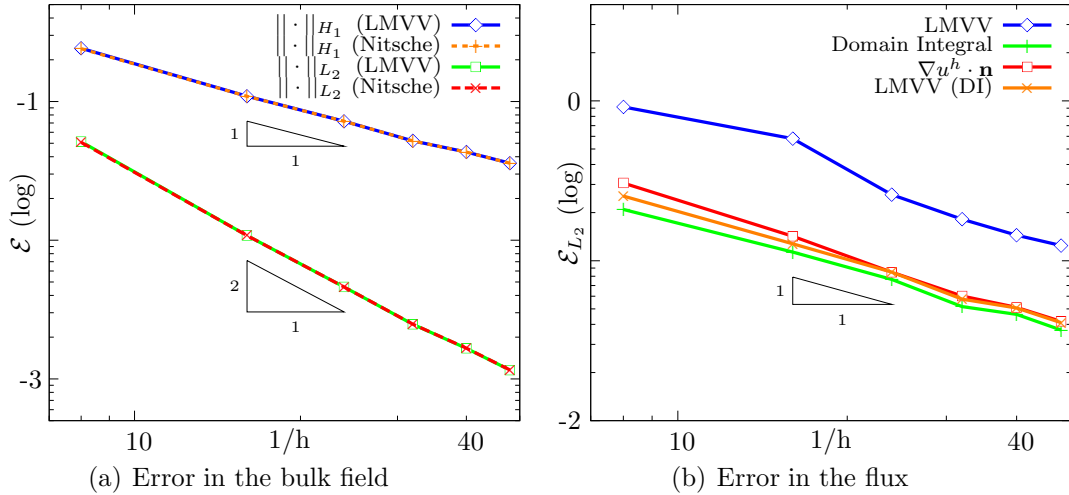


FIGURE 2.7 – Convergence study for the triple cosine problem (2.42) solved on a unit cubic domain with an embedded planar surface.

has been done only for visualizing the results and bears no significance to the way we evaluate the flux on the interface.

*Sensitivity analysis on the position of the surface relative to the bulk mesh*

In order to examine the sensitivity of the methods to the relative position of the surface with respect to the background mesh, we carry out a sensitivity study on a  $16 \times 16 \times 16$  structured mesh in the same spirit as Dolbow and Franca (2008). In essence, for a given background mesh, we plot the error in  $L_2$  norm for the flux as the surface location  $d_z$  is varied with respect to the given background mesh of size  $h$  as exemplified in Figure 2.8(a). From the results reported in Figure 2.8(b), we can see that both domain integral and LMVV exhibit minimal sensitivity to surface location while direct evaluation is quite sensitive to surface location. As regards to accuracy, the flux obtained using domain integral is clearly more accurate than the flux calculated from LMVV or direct evaluation.

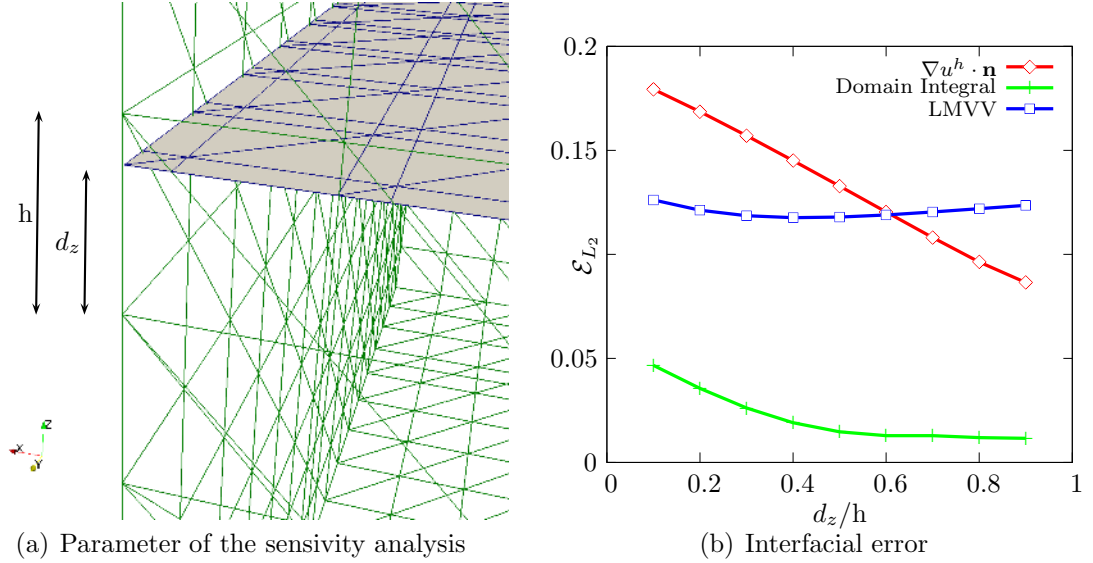


FIGURE 2.8 – Sensitivity analysis for the triple cosine problem. Variation in the flux error as the surface is moved with respect to the background mesh.

### *Spherical surface*

As a next example, we modify the surface geometry in the above considered problem such that we have an embedded spherical surface defined by:

$\psi(\mathbf{x}) = \sqrt{(x - 0.5)^2 + (y - 0.5)^2 + z^2} - 0.81$ . The results for the convergence study are shown in Figure 2.9. We again report optimal rates of convergence in the bulk field as well as the flux. As with the planar surface, the error plots for Nitsche’s method and LMVV overlap in both the  $L_2$  and  $H_1$  norms. We choose this time to show in Figure 2.6(b) the finite element approximation to the bulk field computed with LMVV. The flux on  $\Gamma^*$  is again better approximated with the domain integral applied to Nitsche’s method than for LMVV except for the coarsest bulk mesh investigated. We approximate the embedded surface  $\Gamma^*$  as a union of all piecewise planar surface elements  $\Gamma_e^*$ . As a result, for a coarse discretization, we get a very poor approximation for the interface geometry. We suspect the loss in accuracy for the coarsest case investigated is due to this error introduced while discretizing the embedded surface. When the flux is evaluated using the domain integral method

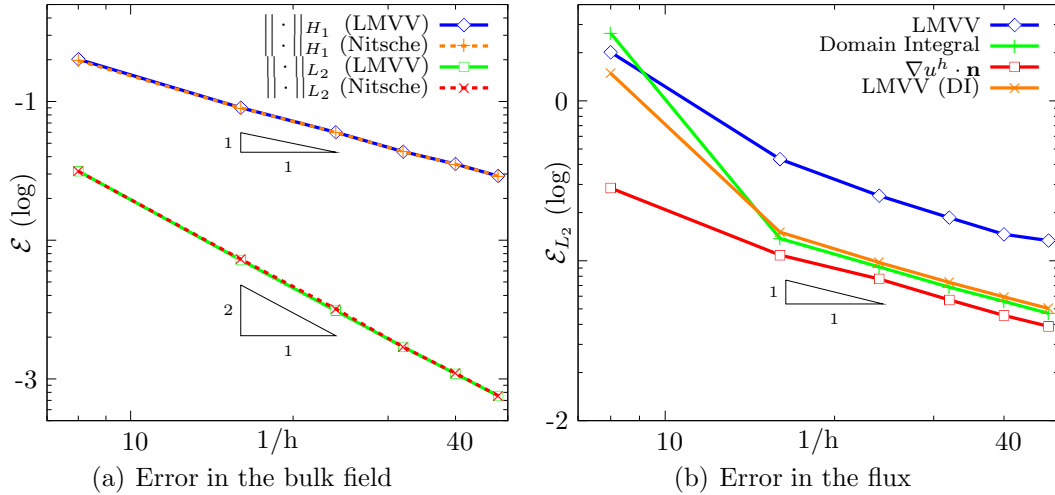


FIGURE 2.9 – Convergence study for the triple cosine problem solved on a unit cubic domain with an embedded spherical surface.

in conjunction with the stable Lagrange multiplier method, the results are almost identical to those evaluated using domain integral method with Nitsche’s method. Contour plots of the flux are shown in Figures 2.6(d) and 2.6(f).

#### 2.4.2 Logarithmic field

We next extend the problem modelling a logarithmic field considered in Dolbow and Franca (2008) to a three dimensional case. The problem is defined on the domain  $\Omega = (0, 1) \times (0, 1) \times (0, 1)$  divided into two disjoint sets  $\Omega^1$  and  $\Omega^2$  by an embedded surface  $\Gamma^*$  located thanks to the level-set  $\psi$ . The exact solution for the bulk variable is given as  $u = \log r$  with  $r$ , a radial coordinate defined with respect to the center of the field  $\mathbf{x}_c$ .  $u$  obeys the following equations:

$$\begin{aligned}
 \Delta u &= 1/r^2 & \text{in } \Omega^1 &= \{\mathbf{x} : \psi(\mathbf{x}) > 0\} \\
 u &= \log r & \text{on } \Gamma_D & \\
 u &= \log r & \text{on } \Gamma^* &= \{\mathbf{x} : \psi(\mathbf{x}) = 0\} \\
 \nabla u \cdot \mathbf{n} &= 0 & \text{on } \Gamma_N &
 \end{aligned}
 \tag{2.43}$$

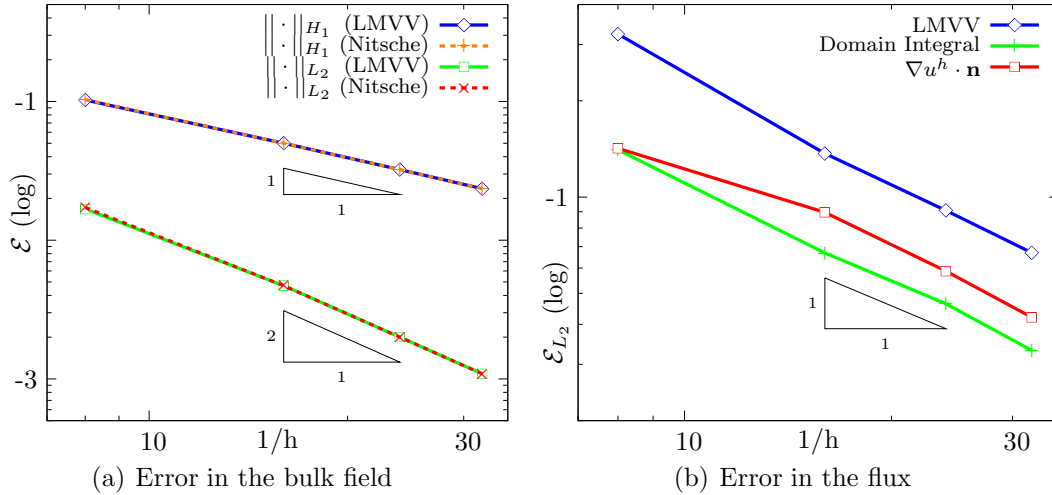


FIGURE 2.10 – Convergence study for the logarithmic field problem (2.43) solved on a unit cubic domain with an embedded spherical surface.

### *Spherical surface*

For this first example, we consider a spherical surface. The symmetry of the problem allows us to solve it only in one-eighth of the entire sphere of radius  $r_0 = 0.41$  which is centered at the origin of the coordinate system as is the logarithmic field. The level-set has thus the simple expression  $\psi = r - r_0$ . The outer Dirichlet boundary  $\Gamma_D$  where collocation is employed and the flux-free Neumann boundary are defined as follows:

$$\Gamma_D = \{\mathbf{x} : x = 1; y = 1; z = 1\} \quad (2.44)$$

$$\Gamma_N = \{\mathbf{x} : x = 0; y = 0; z = 0; \psi(\mathbf{x}) > 0\} \quad (2.45)$$

The domain is discretized with an unstructured tetrahedral mesh. The results of the convergence study are shown in Figure 2.10 and we report optimal rates of convergence in the bulk field as well as the flux for both the methods. As before, global errors for the bulk field are indistinguishable. Thus, only the contour plot of the bulk field for the stable Lagrange multiplier method is presented in Figure 2.14(a). For the flux, we obtain a much better accuracy on using domain integral method in conjunction with Nitsche’s method than for the other two, the flux computed with

LMVV being the least accurate. The contour plots of the relative error for the flux for both methods is shown in Figure 2.14(c) and 2.14(e). It is worth noticing that no clear pattern exists in the contour of the flux for LMVV whereas the error in flux with domain integral is much smoother and smaller.

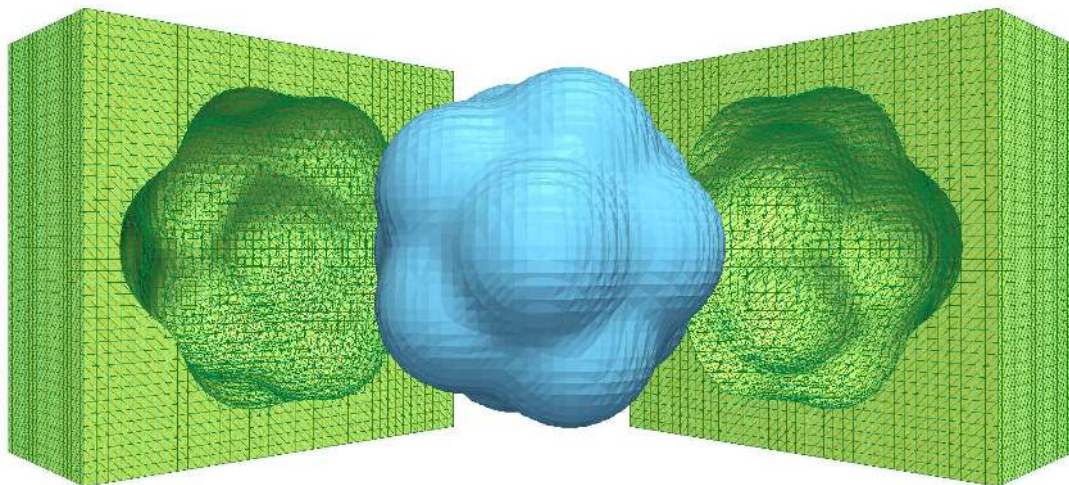


FIGURE 2.11 – Geometry of the computational domain (in green) for the logarithmic field problem with an embedded popcorn shaped surface (in blue).

### *Popcorn shaped surface*

As a final example, we consider an embedded surface in the form of a popcorn as defined in Chern and Shu (2007) centered as the logarithmic field in the middle –  $(x_c, y_c, z_c) = (0.5, 0.5, 0.5)$  – of the domain  $\Omega$ . The surface geometry is defined by the following modified level-set of a sphere of radius  $r_0 = 0.25$ :

$$\psi(\mathbf{x}) = r - r_0 - \sum_{k=0}^{k=11} 4 \exp\left(\frac{(x - 0.5 - x_k)^2 + (y - 0.5 - y_k)^2 + (z - 0.5 - z_k)^2}{0.1^2}\right) \quad (2.46)$$

where:

$$(x_k, y_k, z_k) = \begin{cases} r_0/\sqrt{5} (2 \cos (2k\pi/5), 2 \sin (2k\pi/5), 1); & 0 \leq k \leq 4 \\ r_0/\sqrt{5} (2 \cos ((2(k-5)-1)\pi/5), 2 \sin ((2(k-5)-1)\pi/5), 1); & 5 \leq k \leq 9 \\ (0, 0, r_0); & k = 10 \\ (0, 0, -r_0); & k = 11 \end{cases}$$

The full outer boundary is of Dirichlet type. The domain is discretized with a structured tetrahedral grid. Its geometry is shown in Figure 2.11.

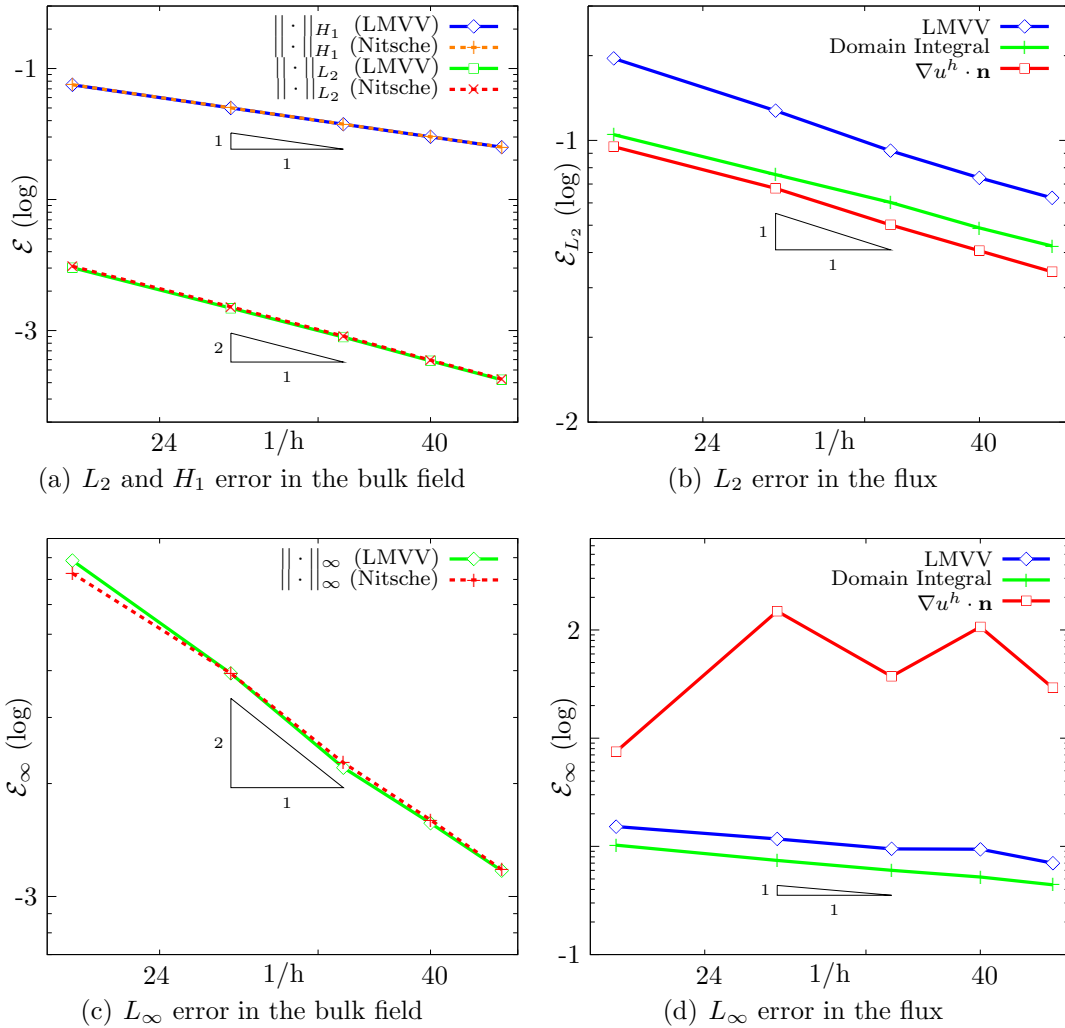


FIGURE 2.12 – Convergence study for the logarithmic field problem (2.43) solved on a unit cubic domain with an embedded popcorn shaped surface.

The results of the convergence study are reported in Figure 2.12. Both the meth-



ods yield optimal rates of convergence in the bulk field and flux. The approximation to the bulk field is indistinguishable for both methods. The contour plot of the bulk field computed with the stable Lagrange multiplier method is shown in Figure 2.14(b). The accuracy in the  $L_2$  norm for the flux is comparable for the direct evaluation and the domain integral technique, and consistent with previous examples, is better than that obtained using the Lagrange multiplier method. The contour plots of the element-wise relative error for the flux are given in Figure 2.14(d) and 2.14(f).

The plots for maximum nodal error and the maximum error in flux are presented in Figures 2.12(c) and 2.12(d). In the bulk field, we report a quadratic rate of convergence in the sup-norm as well. In the sup-norm, the flux converges at a nearly linear rate with the domain integral and LMVV. The rates of convergence are 0.8 and 0.9 respectively. The accuracy in the sup-norm for the flux calculated using domain integral method together with Nitsche’s method is again better than that obtained using LMVV. However, the difference in the obtained accuracies is much lower as compared to  $L_2$  norm. The striking result here is the non-convergence of the flux computed using direct evaluation in the sup-norm. We suspect that this non-convergence is a direct consequence of the sensitivity of the method to the interface location that we reported in Section 2.4.1.

### *2.4.3 Comparative study on constraint enforcement*

For two of the previous examples, we also carried out a convergence study for the error in  $L_2$  norm for the bulk field on the surface. The results of this study are shown in Figure 2.13. We report quadratic rates of convergence for both the methods. The accuracy obtained using LMVV is slightly better than that obtained using Nitsche’s method.

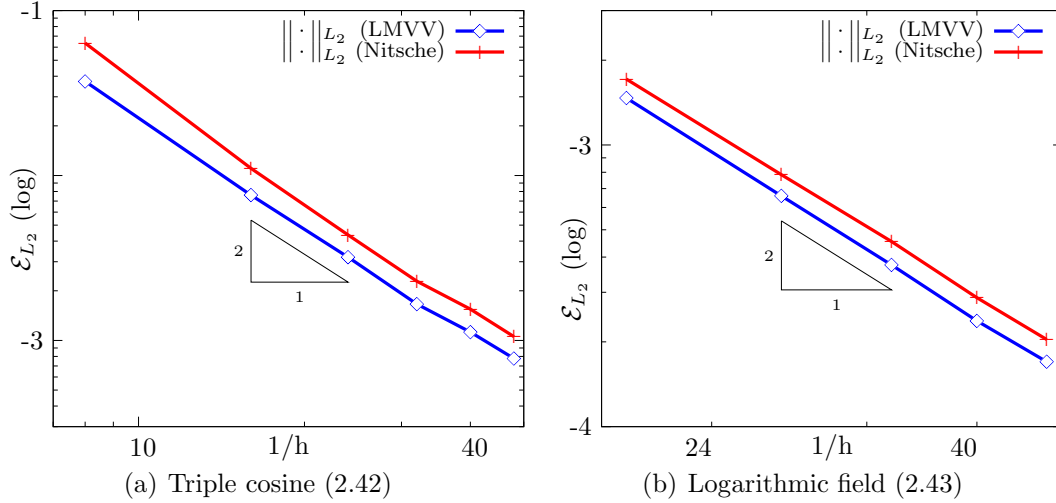
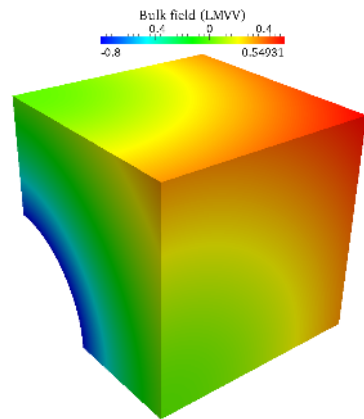


FIGURE 2.13 – Convergence in the bulk field on the surface for Poisson problem solved on a unit cubic domain with an embedded spherical surface (a) and an embedded popcorn shaped surface (b).

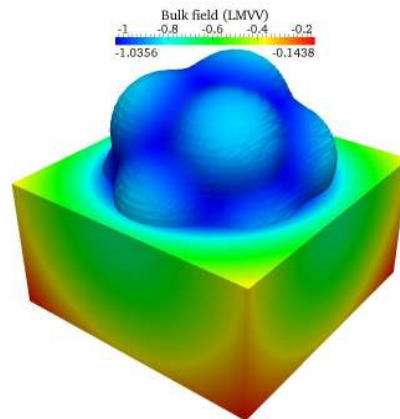
## 2.5 Discussion

A proper comparison between Nitsche’s method and the stable Lagrange multiplier method based on the vital vertex algorithm cannot be achieved only by looking at the bulk field. Indeed, these methods give indistinguishable results for the bulk field in terms of contour plots, error in  $L_2$  and  $H_1$  norms as well as rate of convergence with the mesh refinement in  $L_2$  and  $H_1$  norms, optimal in both cases, for all the proposed benchmark problems. A better means of comparison is to look at the interfacial quantities of interest, *viz*: the flux and constraint enforcement on the embedded surface.

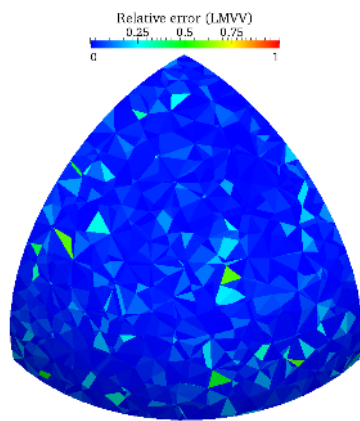
For Nitsche’s method, for reasons of robustness, we recommend that the flux should be evaluated using the domain integral post-processing technique. This assertion is fully justified by the lack of convergence of the directly evaluated flux with the mesh refinement of the error in the sup norm. For pathological cases where the surface is oriented such that the volume fraction of the element in the computational domain is really small, the direct evaluation is just not robust enough and yields



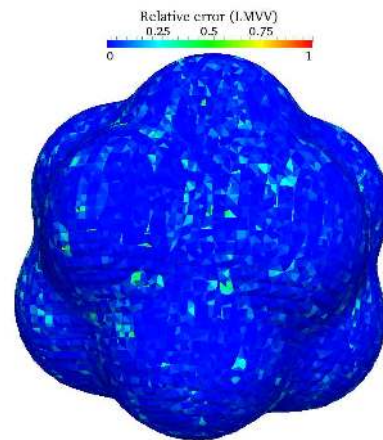
(a) Bulk field



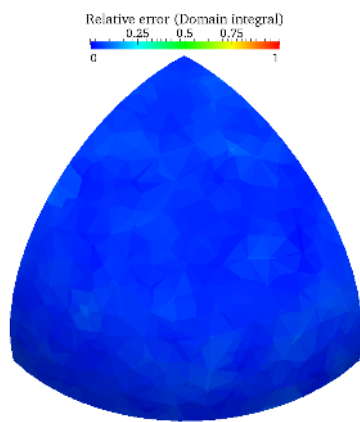
(b) Bulk field



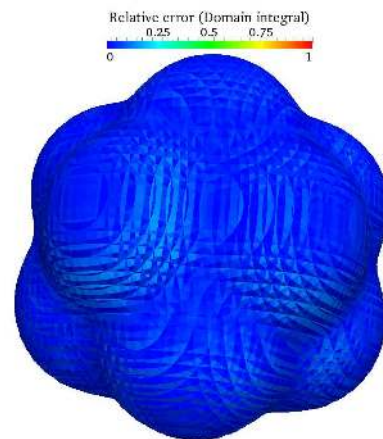
(c) Error in flux with LMVV



(d) Error in flux with LMVV



(e) Error in flux with Nitsche



(f) Error in flux with Nitsche

FIGURE 2.14 – Finite element approximation of the bulk field and the flux for the logarithmic field problem (2.43) with an embedded spherical surface (left) and a popcorn shaped surface (right). The errors in flux are relative and computed with the domain integral technique for Nitsche’s method.

non-physical values.

For the stable Lagrange multiplier method based on the vital vertex algorithm, flux converges with the mesh refinement at an optimal rate of convergence in the  $L_2$  norm and at a nearly optimal rate in the sup norm when directly constructed using the Lagrange multipliers. However, such a construction, results in a much reduced accuracy in the flux as compared to that obtained from Nitsche's method with domain integrals. We suspect that on choosing to implement the constraint at the vital vertices alone, and thereby reducing the Lagrange multiplier space, we are also reducing the approximation space for the flux. Thus, for the same degree of refinement, we get a much coarser surface mesh with LMVV than with Nitsche's method and hence the loss in accuracy. Again an efficient way to regain good accuracy for flux for this method is to post-treat the bulk field on the embedded surface using domain integrals. The flux then obtained is much more accurate and comparable to that computed with Nitsche's method.

From an implementation perspective, Nitsche's method is clearly less intrusive in standard codes as all the modifications arise at the element level. In the case of the Lagrange multiplier method even with the proposed shape functions defined in each cut element, a dedicated assembly procedure is required. Furthermore, the vital vertex algorithm needs to be extended to evolving surfaces and domain decomposition in order to be truly versatile. In terms of solvers, Nitsche's method fully benefits from the large range of methods for positive definite systems which make the resolution much faster than the one of the saddle-point system resulting from Lagrange multiplier method. Also, as opposed to Lagrange multiplier method the dimension of the resulting system remains the same on using Nitsche's method.

## 2.6 Conclusion

This Chapter presents a comparative study of two robust methods to impose Dirichlet boundary conditions on embedded surfaces. We propose a variational derivation of Nitsche’s method and recall its main features: the computation of the stabilization parameter and the method of domain integrals to compute the flux. We also investigate a concurrent stable Lagrange multiplier method and contrast its performance against Nitsche’s method. We briefly reintroduce the key feature of the vital vertex algorithm – a means to coarsen the surface mesh to prevent over-constraining the system – on which the method is based. We propose a new set of shape functions for interpolating the Lagrange multipliers on the embedded surface, defined element by element which are thus of importance due to the implementational ease they offer in a finite element context.

These two methods are compared on several benchmark examples. We report optimal rates of convergence with mesh refinement in the  $H_1$ ,  $L_2$  and  $L_\infty$  error norms in the bulk field for both the methods. The errors for the flux converge optimally in the  $L_2$  norm and quasi optimally in the  $L_\infty$  norm for Nitsche’s method in conjunction with the domain integral method and for the stable Lagrange multiplier method. Construction of flux directly using Lagrange multipliers on the interface results in poorer accuracy as opposed to that obtained using Nitsche’s method together with domain integral formulation. Therefore, we recommend post-processing with the domain integral technique even for the stable Lagrange multiplier method.

While the results for the numerical experiments in this Chapter are largely positive, the main concern lies with the non-convergence of flux (with direct calculation) for Nitsche’s method in the  $L_\infty$  norm. It needs to be established whether this is a direct consequence of the possible poor-conditioning that will result from a large value of the stabilization parameter for cut elements with arbitrarily small volume

fractions. In addition, for such elements, we also need to look more closely at the effect of the stabilization parameter on the temporal stability in explicit time-stepping. We investigate these questions in greater detail in the next Chapter, where we extend these methods to enforce perfect continuity constraints over embedded surfaces in elastodynamics with explicit time-stepping.

# Imposing Stiff Interfacial Constraints Over Embedded Surfaces in Explicit Dynamics

## 3.1 Introduction

In this Chapter, we consider two extensions to the stable and stabilized methods proposed in Chapter 2. Firstly, we now focus on enforcing a perfect continuity constraint across an embedded surface. In other words, while the previous Chapter considered the special case of treating voids, here we relax that constraint and consider bi-material interfaces. Secondly, and perhaps, more importantly, here we consider elastodynamics problems with explicit time integration. Consequently, apart from the spatial stability concerns illustrated in the previous Chapter, we also need to account for temporal stability concerns. Perhaps, the key question that needs addressing is whether the spatial and temporal stability aspects are coupled.

As discussed in Chapter 2, there has been significant progress in addressing spatial stability concerns for quasi-static problems. However, prior to this work, very little attention has been devoted to the performance of these methods for transient problems. The numerical artifacts (namely, traction oscillations) observed in quasi-

static regimes have also been reported in a dynamic setting (see Remmers et al., 2001; Simone, 2004). It is natural to think that the methods developed for quasi-static problems would resolve the spatial stability issues in dynamic problems as well. However, as a first step towards modeling more complex physical problems like dynamic cohesive fracture and polycrystalline media subjected to high strain rates, it is essential to study the effect of spatial stabilization on the temporal stability of explicit dynamic simulations.

The first study to identify the incompatibility of X-FEM with explicit dynamics was by Belytschko et al. (2003). They reported that as the discontinuity approached an element boundary, the critical time step required for stable computations in explicit dynamics approached zero. Remmers et al. (2003) and de Borst et al. (2006) also reported the critical time step problem with discontinuous enrichment. These studies then all used either fully implicit time integration or implicit-explicit methods (Hughes and Liu, 1978) with implicit time integration utilized only for cut elements.

This incompatibility was traced back to the direct extension of the row-summing mass lumping technique from the Finite Element Method to X-FEM by Menouillard et al. (2006). Accordingly, they proposed a modified lumping procedure that resulted in a critical time step of the same order of magnitude as that of an uncut element even as the discontinuity approached the boundaries of the cut element. Rozycki et al. (2008) further improved on this method for modeling voids with constant strain triangles and tetrahedra. Their approach resulted in identical time steps for cut and uncut elements. Menouillard et al. (2008) adapted this approach to model cracks with Heaviside enrichment and showed that the critical time step for constant strain cracked elements remained identical to a classical finite element. Elguedj et al. (2009) further generalized this procedure for arbitrary enrichment functions.

It is noteworthy that all the above studies assumed that the interfaces (crack or void surfaces) were traction free. Unfortunately, this assumption does not re-



main valid while modeling cohesive fracture or weak discontinuities such as material interfaces. To our knowledge, prior to this work, neither the lumping procedures developed for unconstrained problems nor the methods to enforce constraints on embedded surfaces have been examined within an explicit dynamics framework for constrained problems. Also, while the stable and stabilized methods presented in Chapter 2 stand out as the two most efficient techniques to enforce constraints in a quasi-static framework; there is a need to evaluate their relative merits/demerits in an explicit dynamics framework. To that end, here we extend both the stable and stabilized embedded methods presented in Chapter 2 to explicit dynamics scenarios.

This Chapter is organized as follows. The model problem and the variational forms are described in the next section. In Section 3.3, we outline the spatial discretization for the stable and stabilized forms and provide a lower bound on the Nitsche stabilization parameter. In Section 3.4, we perform a theoretical study on the bounds for the critical time step with particular focus on the stabilized framework. We also develop a novel mass lumping procedure for the stabilized method. In Section 3.5, we present results from several three-dimensional benchmark examples where we compare the performance of the stabilized and stable formulations against each other as well as against the penalty method. Finally, we summarize and provide concluding remarks in the last section.

## 3.2 Model Problem and Variational Formulations

We consider an elastodynamics problem defined on the domain  $\Omega$ , divided into component sub-domains  $\Omega^1$  and  $\Omega^2$  by the embedded surface  $\Gamma_*$ . We specify displacements on part of the exterior boundary such that  $u_i^m(t) = u_d^m(t)$  on the Dirichlet boundaries  $\Gamma_d^m$  and specify tractions on the Neumann boundaries such that  $\sigma_{ij}^m n_j^m = p_i(t)$  on  $\Gamma_n^m$ , where  $m = 1, 2$  is an index denoting the sub-domain  $\Omega^m$ . The complementary part of the boundary is considered traction free. A schematic

of the problem domain is shown in Figure 3.1.

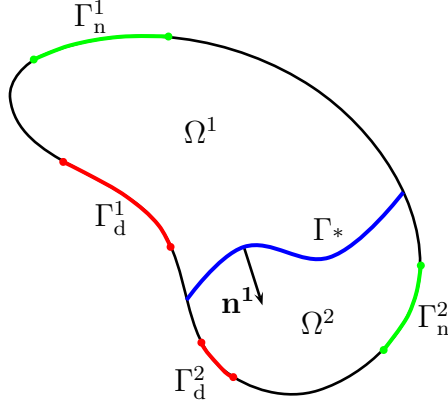


FIGURE 3.1 – Domain  $\Omega$  divided by the interface  $\Gamma_*$  into sub-domains  $\Omega^1$  and  $\Omega^2$ . The Dirichlet boundaries  $\Gamma_d^1$ ,  $\Gamma_d^2$  and the Neumann boundaries  $\Gamma_n^1$  and  $\Gamma_n^2$  are shown. The complementary part of the boundary is traction free. The normal to the boundary of each sub-domain,  $\mathbf{n}^m$ , points outwards as shown.

The governing equations are given (in indicial notation) by:

$$\begin{aligned}
 \sigma_{ij,j}^m &= \rho^m \ddot{u}_i^m & \text{in } \Omega^m \times (0, T), \\
 u_i^m(t) &= u_d^m(t) & \text{on } \Gamma_d^m \times (0, T), \\
 \sigma_{ij}^m n_j^m &= p_i(t) & \text{on } \Gamma_n^m \times (0, T), \\
 u_i^m(0) &= u_0^m & \text{in } \Omega^m \text{ at } T = 0, \\
 \dot{u}_i^m(0) &= \dot{u}_0^m & \text{in } \Omega^m \text{ at } T = 0.
 \end{aligned} \tag{3.1}$$

The interfacial coupling conditions are traction balance and displacement continuity across the interface:

$$\begin{aligned}
 \sigma_{ij}^1 n_j^1 &= -\sigma_{ij}^2 n_j^2 & \text{on } \Gamma_* \times (0, T), \\
 u_i^1 &= u_i^2 & \text{on } \Gamma_* \times (0, T).
 \end{aligned} \tag{3.2}$$

We assume that the material in both sub-domains follows a linear elastic constitutive relationship such that  $\sigma_{ij}^m = C_{ijkl}^m u_{(k,l)}^m$ .

### 3.2.1 Variational forms

Let us begin by defining the space of trial solutions,  $\mathbb{U}_t = \mathbb{U}_t^1 \times \mathbb{U}_t^2$  and their variations,  $\mathbb{V} = \mathbb{V}^1 \times \mathbb{V}^2$  such that:

$$\begin{aligned}\mathbb{U}_t^m &= \{u_i(t) \in H^1(\Omega^m), u_i(t)|_{\Gamma_d^m} = u_d^m(t)\}, \\ \mathbb{V}^m &= \{\delta u_i \in H^1(\Omega^m), \delta u_i|_{\Gamma_d^m} = 0\}.\end{aligned}$$

We formulate the Lagrangian of such a system by evaluating the difference between its kinetic and potential energies:

$$\mathcal{L} = \sum_m \left( \frac{1}{2} \int_{\Omega^m} \dot{u}_i^m \rho^m \dot{u}_i^m \, d\Omega - \frac{1}{2} \int_{\Omega^m} u_{(i,j)}^m C_{ijkl}^m u_{(k,l)}^m \, d\Omega + \int_{\Gamma_n^m} p_i u_i^m \, d\Gamma \right). \quad (3.3)$$

On employing Hamilton's principle of least action to minimize the action integral  $\mathcal{I} = \int \mathcal{L} \, dt$ , subject to the constraints (3.2), we recover the set of equations (3.1) as the Euler-Lagrange equations for the considered variational problem. Within this variational framework, we can enforce the constraints through one of the following ways:

*Lagrange multiplier approach:*

In this approach, we build in the interfacial constraints into the Lagrangian of the system by adding the work done by Lagrange multipliers  $\lambda_i(t)$ , to the potential energy of the system. We define the space of Lagrange multipliers as  $\mathbb{L}_t = \{\lambda(t) \in H^{-1/2}(\Gamma_*)\}$ . In other words,

$$\mathcal{L}^{\text{Lag}} = \mathcal{L} - \int_{\Gamma_*} \lambda_i \llbracket u_i \rrbracket \, d\Gamma, \quad (3.4)$$

where, by  $\llbracket \cdot \rrbracket$ , we denote the jump in a quantity  $(\cdot)$  across an interface, in particular,  $\llbracket \cdot \rrbracket = (\cdot)^2 - (\cdot)^1$ . Now on insisting the stationarity of the action integral, we get the

following two-field variational formulation: find  $(u_i(t), \lambda_i(t)) \in \mathbb{U}_t \times \mathbb{L}_t$ ,  $t \in (0, T)$  such that for all  $(\delta u_i, \delta \lambda_i) \in \mathbb{V} \times H^{-1/2}(\Gamma_*)$

$$\delta \mathcal{I} = \int_T (\delta \mathcal{L} - \int_{\Gamma_*} \lambda_i \llbracket \delta u_i \rrbracket \, d\Gamma - \int_{\Gamma_*} \delta \lambda_i \llbracket u_i \rrbracket \, d\Gamma) \, dt = 0, \quad (3.5)$$

where,

$$\delta \mathcal{L} = \sum_m \left( - \int_{\Omega} \delta u^m \rho^m \ddot{u}_i^m \, d\Omega - \int_{\Omega^m} \delta u_{(i,j)}^m C_{ijkl}^m u_{(k,l)}^m \, d\Omega + \int_{\Gamma_n^m} p_i \delta u_i^m \, d\Gamma \right).$$

The arbitrariness of  $(\delta u_i, \delta \lambda_i)$  returns the well known mixed system of equations. As stated before, in a discrete setting, the spatial stability of this dual formulation depends on the satisfaction of an inf-sup condition for the interpolating spaces. In Section 3.4, we extend the stable formulation presented in Chapter 2 to dynamic regimes.

*Penalty function approach:*

For dynamic problems, in addition to the inf-sup stability restrictions, it is also established that a dual formulation is incompatible with purely explicit time integration (Carpenter et al., 1991). These restrictions have motivated the development and use of penalty function based approaches. In a penalty function approach, the jump constraint given by (3.2) is replaced by a spring-like Robin constraint. In other words,

$$\sigma_{ij} n_j = \sigma_{ij}^1 n_j^1 = -\sigma_{ij}^2 n_j^2 = \alpha_{ij} \llbracket u_j \rrbracket, \quad (3.6)$$

where,  $\alpha_{ij}$  can be interpreted as the stiffness of a spring connecting the two sub-domains together. Further, on choosing  $\alpha_{ij} = \frac{1}{2} \alpha \delta_{ij}$ , where  $\delta_{ij}$  is the Kroenecker delta function, and interpreting the Lagrange multipliers as the traction at the interface,

we arrive at the Lagrangian for the primal penalty function based approach:

$$\mathcal{L}^{\text{Pen}} = \mathcal{L} - \int_{\Gamma_*} \frac{1}{2} \llbracket u_i \rrbracket \alpha \llbracket u_i \rrbracket \, d\Gamma. \quad (3.7)$$

The stationarity of the action integral now yields: find  $u_i(t) \in \mathbb{U}_t$ ,  $t \in (0, T)$  such that for all  $\delta u_i \in \mathbb{V}$

$$\delta \mathcal{I} = \int_T (\delta \mathcal{L} - \int_{\Gamma_*} \llbracket \delta u_i \rrbracket \alpha \llbracket u_i \rrbracket \, d\Gamma) \, dt = 0. \quad (3.8)$$

The main drawback of this approach is that the desired constraint is achieved exactly only in the limit  $\alpha \rightarrow \infty$ . Due to this approximation, the method is no longer variationally consistent. Often, to get accurate results, one requires the use of a very large penalty parameter which results in ill-conditioned systems and artificial oscillations in the traction field at the interface (Simone, 2004; Sanders et al., 2009). For explicit dynamics simulations, the stable time step is also inversely related to the chosen penalty parameter. This creates a three-way competition between accurate constraint enforcement, traction recovery and computational expense associated with explicit time stepping and makes the choice for the parameter all the more challenging. We discuss these issues further in Section 3.4.

*Nitsche's method:*

The Lagrange multiplier approach has the attractive property of variational consistency while the penalty method does not require building an inf-sup stable space and is compatible with explicit dynamics. Nitsche's approach is a way to combine the attractive features of both these methods to result in a variationally consistent, one-field approach that is also compatible with explicit dynamics. However, the key to the performance of this approach lies in the choices we make for certain free parameters that arise in this method. To see that further, consider the derivation of this

method from an augmented Lagrange multiplier framework. Within this framework, the Lagrangian can be constructed as:

$$\mathcal{L}^{\text{Aug}} = \mathcal{L} - \int_{\Gamma_*} (\lambda_i + \frac{1}{2}\alpha[[u_i]])[[u_i]] \, d\Gamma. \quad (3.9)$$

The stationarity of the action integral yields:

$$\delta\mathcal{I} = \int_T (\delta L - \int_{\Gamma_*} \delta\lambda_i[[u_i]] \, d\Gamma - \int_{\Gamma_*} [[\delta u_i]](\lambda_i + \alpha[[u_i]]) \, d\Gamma) \, dt = 0. \quad (3.10)$$

Now, on using the product rule of differentiation, applying the divergence theorem and grouping together the terms on the embedded boundary, we obtain:

$$- \sum_m \int_{\Gamma_*} \delta u_i^m \sigma_{ij}^m n_j^m \, d\Gamma - \int_{\Gamma_*} \delta\lambda_i[[u_i]] \, d\Gamma - \int_{\Gamma_*} [[\delta u_i]](\lambda_i + \alpha[[u_i]]) \, d\Gamma = 0. \quad (3.11)$$

Further, on using the traction continuity at the interface in (3.11), for any  $\gamma^m \in [0, 1]$  such that  $\gamma^1 + \gamma^2 = 1$ , we obtain

$$- \int_{\Gamma_*} \delta\lambda_i[[u_i]] \, d\Gamma - \int_{\Gamma_*} [[\delta u_i]](\lambda_i + \langle \sigma_{ij} \rangle_\gamma n_j^1 + \alpha[[u_i]]) \, d\Gamma = 0, \quad (3.12)$$

where,  $\langle \sigma_{ij} \rangle_\gamma = \gamma^1 \sigma_{ij}^1 + \gamma^2 \sigma_{ij}^2$  is the weighted average of the stress on the interface. Now, the arbitrariness of  $(\delta u_i, \delta\lambda_i)$  returns the displacement continuity constraint at the interface and also provides us with the physical interpretation of the multiplier as the weighted average of the traction at the interface such that  $\lambda_i = -\langle \sigma_{ij} \rangle_\gamma n_j^1$ . On using this interpretation of  $\lambda_i$  in (3.10) we arrive at a Nitsche stabilized and consistent primal variational form: find  $u_i(t) \in \mathbb{U}_t$ ,  $t \in (0, T)$  such that for all  $\delta u_i \in \mathbb{V}$

$$\delta\mathcal{I} = \int_T (\delta L + \int_{\Gamma_*} [[u_i]] \langle \delta \sigma_{ij} \rangle_\gamma n_j^1 \, d\Gamma + \int_{\Gamma_*} [[\delta u_i]] \langle \sigma_{ij} \rangle_\gamma n_j^1 \, d\Gamma - \int_{\Gamma_*} \alpha [[\delta u_i]] [[u_i]] \, d\Gamma) \, dt = 0. \quad (3.13)$$

The two additional boundary integrals in the Nitsche weak form that we see as opposed to the penalty approach provide the method with variational consistency and symmetry while the penalty like term is used to stabilize the weak form and restore its coercivity. It must be recalled that coercivity here not only ensures spatial stability but for transient problems also guarantees that the discrete energy never grows with time (Hughes, 2000). At the same time, as in the penalty method, the stabilization parameter adversely affects the stable time step required for explicit dynamics calculations. In Nitsche’s method, the stabilization parameter then serves to couple the spatial and temporal stability requirements. A judicious choice for this parameter is therefore critical for the performance of the method from both spatial and temporal perspectives.

In addition to the stabilization parameter, we also have to make a suitable choice for the interfacial weights,  $\gamma^m$ . We have a lot of latitude in our choice of these weights provided  $\gamma^1 + \gamma^2 = 1$ . However, the key thing to appreciate is that once we have made a choice for these weights, the stabilization parameter can no longer be arbitrarily chosen if we desire stability. As a quick remark, we also point out that the choice of  $\gamma^1 = \gamma^2 = 0.5$  returns the most commonly used “unbiased” form of Nitsche’s method (see Arnold et al., 2002). As we shall see in the next section, this is not necessarily the best choice. By contrast, choosing  $\gamma^m$  at an element level allows us to arrive at an estimate for the stabilization parameter which conforms to both spatial and temporal stability requirements.

### 3.3 Spatial Discretization

We discretize the problem domain  $\Omega$  by a set of non overlapping elements  $\Omega_e$ . The interface  $\Gamma_*$  is allowed to be embedded in the underlying background mesh allowing for a natural discretization of the interface by the vertex set of intersection points between the interface and the element edges. We adapt a level-set approach to

identify the location of the interface within the background mesh. Any element with positive level-set values for all its nodes belongs to  $\Omega^1$  and with negative level-set values for all its nodes belongs to  $\Omega^2$ . Elements which have nodes with both positive and negative level-set values are considered to contain the embedded discontinuity. We follow the approach suggested in Hansbo and Hansbo (2002) and consider every such element with an embedded discontinuity to be a superposition of two elements, one contributing to the discretization in  $\Omega^1$  and the other in  $\Omega^2$ . In other words, the discretized domain results from two separate meshes, one for each sub-domain, that overlap at the interface. For concreteness, the overlapping element formulation for a tetrahedral element, with a triangular surface element, is shown in Figure 3.2. A similar element can also be constructed when the resulting surface element is a quadrilateral.

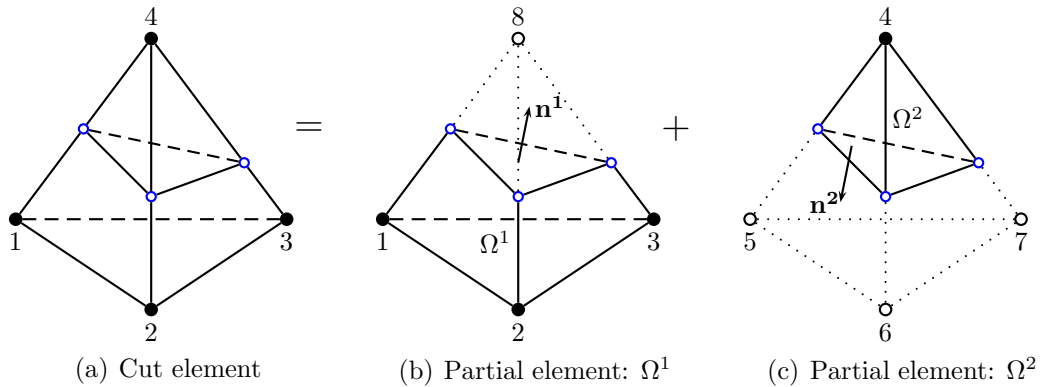


FIGURE 3.2 – Overlapping element formulation for a tetrahedral element cut by an embedded interface. The black circles are the physical nodes corresponding to the background mesh and the hollow circles are the ghost nodes. The blue circles represent the discretization of the embedded interface within an element. The discrete interface is constructed to be piecewise-planar throughout the mesh, and is based on a zero level-set. The discrete interfacial normal  $\mathbf{n}$  in each element is then given by that of the discrete interfacial plane.

A similar reasoning for constructing the discrete approximation leads us to consider the finite element approximation to the solution as an additive decomposition of the solution field existing entirely in  $\Omega^1$  and  $\Omega^2$ . Alternatively, for the solution



field and its variation, we write

$$\mathbf{u}^h = \sum_m \sum_{i \in I^m} H^m \mathbf{N}_i \mathbf{u}_i, \quad \delta \mathbf{u}^h = \sum_m \sum_{i \in I^m} H^m \mathbf{N}_i \delta \mathbf{u}_i, \quad (3.14)$$

where,  $I^m$  is the set of all nodes whose supports overlap the domain  $\Omega^m$ . The nodal degrees of freedom are denoted by  $\mathbf{u}_i$  and the interpolating functions are denoted by  $\mathbf{N}_i$ .  $H^m$  is the characteristic function given by

$$H^m(\mathbf{x}) = \begin{cases} 1 & \text{if } \mathbf{x} \in \Omega^m, \\ 0 & \text{otherwise.} \end{cases} \quad (3.15)$$

### 3.3.1 Dual approach

On using Lagrange multipliers to enforce the constraints, the system of semi-discrete equations of motion is given by:

$$\begin{aligned} \mathbf{M}\ddot{\mathbf{u}} + \mathbf{K}_b \mathbf{u} + \mathbf{G}\lambda &= \mathbf{F}_{\text{ext}}, \\ \mathbf{G}^T \lambda &= \mathbf{0}, \\ \mathbf{u}(0) &= \mathbf{u}_0, \\ \dot{\mathbf{u}}(0) &= \dot{\mathbf{u}}_0. \end{aligned} \quad (3.16)$$

Here,  $\ddot{\mathbf{u}}$ ,  $\dot{\mathbf{u}}$  and  $\mathbf{u}$  represent the nodal values of accelerations, velocities and displacements respectively. The mass matrix  $\mathbf{M}$  and the bulk stiffness matrix  $\mathbf{K}_b$  inherit the overlapping structure from the discrete formulation and are given as:

$$\mathbf{M} = \begin{bmatrix} \mathbf{M}_b^1 & \mathbf{0} \\ \mathbf{0} & \mathbf{M}_b^2 \end{bmatrix}, \quad \mathbf{K} = \begin{bmatrix} \mathbf{K}_b^1 & \mathbf{0} \\ \mathbf{0} & \mathbf{K}_b^2 \end{bmatrix}. \quad (3.17)$$

The consistent mass matrix can be evaluated in a straightforward manner as:

$$\mathbf{M}_b^m = \sum_e \int_{\Omega_e^m} \mathbf{N}^T \rho^m \mathbf{N} \, d\Omega_e \quad \text{for } m = 1, 2, \quad (3.18)$$

where  $\mathbf{N}$  is the shape function matrix and  $\Omega_e^m$  denotes the part of the element that belongs to  $\Omega^m$ . A detailed discussion on mass lumping techniques for constrained embedded problems is provided in Section 3.4.

The bulk stiffness terms in each of the sub-domains can be calculated as:

$$\mathbf{K}_b^m = \sum_e \int_{\Omega_e^m} \mathbf{B}^T \mathbf{D}^m \mathbf{B} \, d\Omega_e \quad \text{for } m = 1, 2, \quad (3.19)$$

where, the matrix  $\mathbf{B}$  contains the gradient of shape functions and the matrix  $\mathbf{D}$  denotes the discrete counterpart of the elasticity tensor. The  $\mathbf{G}$  matrix is the contribution to the stiffness matrix due to the Lagrange multipliers:

$$\mathbf{G} = \sum_e \int_{\Gamma_*^e} \mathbf{N}^T \mathbf{N}_\lambda \, d\Gamma_e, \quad (3.20)$$

where  $\mathbf{N}_\lambda$  are the interpolating functions for the Lagrange multipliers and  $\Gamma_*^e$  denotes the portion of the interface  $\Gamma_*$  residing in the element. The interpolation for the Lagrange multipliers and the coarsening of the multiplier space is done according to Chapter 2.

### 3.3.2 Primal approaches

For the primal formulation, on introducing the spatial discretization described above and using standard arguments regarding the arbitrariness of  $\delta u^h$ , we recover a set of semi-discrete equations of motion given by:

$$\begin{aligned} \mathbf{M}\ddot{\mathbf{u}} + \mathbf{K}\mathbf{u} &= \mathbf{F}_{\text{ext}}, \\ \mathbf{u}(0) &= \mathbf{u}_0, \\ \dot{\mathbf{u}}(0) &= \dot{\mathbf{u}}_0. \end{aligned} \quad (3.21)$$

The stiffness matrix  $\mathbf{K}$  is now given as:

$$\mathbf{K} = \begin{bmatrix} \mathbf{K}_b^1 + \mathbf{K}_n^1 + \mathbf{K}_s^1 & \mathbf{K}_c \\ \mathbf{K}_c^T & \mathbf{K}_b^2 + \mathbf{K}_n^2 + \mathbf{K}_s^2 \end{bmatrix}. \quad (3.22)$$

The mass matrix  $\mathbf{M}$  and the bulk stiffness  $\mathbf{K}_b$  remain identical to those described by equations (3.18) and (3.19). However, the stiffness matrix for the primal methods

differs from the stiffness matrix for the dual method due to the presence of interfacial terms on the leading diagonal and the off-diagonal blocks. We can write the Nitsche contribution to the leading diagonal block of the stiffness matrix as

$$\mathbf{K}_n^m = -\gamma^m \sum_e \int_{\Gamma_*^e} \mathbf{N}^T ((\mathbf{n}^m)^T \mathbf{D}^m) \mathbf{B} \, d\Gamma_e - \gamma^m \sum_e \int_{\Gamma_*^e} \mathbf{B}^T ((\mathbf{D}^m)^T \mathbf{n}^m) \mathbf{N} \, d\Gamma_e \text{ for } m = 1, 2, \quad (3.23)$$

while the contribution of the stabilization terms to the leading diagonal block is given by:

$$\mathbf{K}_s^m = \sum_e \alpha_e \int_{\Gamma_*^e} \mathbf{N}^T \mathbf{N} \, d\Gamma_e \text{ for } m = 1, 2. \quad (3.24)$$

In Section 3.3.3, we leverage numerical analysis to establish an algebraic expression for the stabilization parameter  $\alpha_e = \alpha|_{\Omega_e}$ , for constant strain triangular and tetrahedral elements.

The Nitsche and the stabilization terms also contribute to the off-diagonal block of the stiffness matrix denoted by  $\mathbf{K}_c$ . The off-diagonal block can be evaluated as

$$\mathbf{K}_c = -\sum_e \alpha_e \int_{\Gamma_*^e} \mathbf{N}^T \mathbf{N} \, d\Gamma_e + \gamma^1 \sum_e \int_{\Gamma_*^e} \mathbf{N}^T ((\mathbf{n}^1)^T \mathbf{D}^1) \mathbf{B} \, d\Gamma_e + \gamma^2 \sum_e \int_{\Gamma_*^e} \mathbf{B}^T ((\mathbf{D}^2)^T \mathbf{n}^2) \mathbf{N} \, d\Gamma_e. \quad (3.25)$$

Finally, we notice that it is convenient to use the above discrete form to implement the penalty method as well. On omitting the Nitsche specific terms from the leading diagonal and the off-diagonal blocks of the stiffness matrix, one obtains the discretized form of the penalty method. The only other difference would then lie in the choice of the stabilization parameter  $\alpha_e$ , which is now completely at an analyst's discretion.

### 3.3.3 Estimate for the stabilization parameter

In arriving at the element-level stabilization parameter  $\alpha_e$ , we insist on the positive-ness of discrete energy, locally, for elements intersected by the embedded surface. To facilitate this analysis, we proceed by defining an “energy” norm of a quantity over a domain  $\Omega_e$  and  $L_2$  norm of a quantity over a surface  $\Gamma_*^e$  as

$$\|\mathbf{u}\|_{\Omega_e}^2 = \int_{\Omega_e} u_{(i,j)} C_{ijkl} u_{(k,l)} \, d\Omega_e, \quad \|\mathbf{u}\|_{\Gamma_*^e}^2 = \int_{\Gamma_*^e} u_i u_i \, d\Gamma_e. \quad (3.26)$$

From (3.13), we can write the discrete energy for the considered model problem as

$$\sum_m (\mathbf{u}^m, \rho^m \ddot{\mathbf{u}}^m)^e + a^e(\mathbf{u}, \mathbf{u}), \quad (3.27)$$

where,

$$a^e(\mathbf{u}, \mathbf{u}) = \sum_m \int_{\Omega_e^m} u_{(i,j)}^m C_{ijkl}^m u_{(k,l)}^m \, d\Omega_e - 2 \int_{\Gamma_*^e} \llbracket u_i \rrbracket \langle C_{ijkl} u_{(k,l)} \rangle_{\gamma} n_j^1 \, d\Gamma_e + \int_{\Gamma_*^e} \alpha \llbracket u_i \rrbracket \llbracket u_i \rrbracket \, d\Gamma_e. \quad (3.28)$$

The mass matrix as seen from (3.18) admits a quadratic form and hence does not affect the spatial stability requirements of the method. Therefore, the stability requirements are solely governed by the coercivity of the bilinear form  $a^e(\mathbf{u}, \mathbf{u})$ . Rewriting (3.28) in a discrete setting by employing the Cauchy-Schwarz inequality on the second term, we obtain

$$a^e(\mathbf{u}^h, \mathbf{u}^h) \geq \|\mathbf{u}^h\|_{\Omega_e}^2 + \alpha_e \|\llbracket \mathbf{u}^h \rrbracket\|_{\Gamma_*^e}^2 - 2 \|\llbracket \mathbf{u}^h \rrbracket\|_{\Gamma_*^e} \|\langle C_{ijkl} u_{(k,l)}^h \rangle_{\gamma} n_j\|_{\Gamma_*^e}. \quad (3.29)$$

From the well-known generalized inverse estimate (Brezzi and Fortin, 1991), there exists a mesh-dependent positive constant  $C_I$  such that

$$\|\langle C_{ijkl} u_{(k,l)}^h \rangle_{\gamma} n_j\|_{\Gamma_*^e} \leq C_I \|\mathbf{u}^h\|_{\Omega_e}. \quad (3.30)$$

From (3.29) and (3.30),

$$a^e(\mathbf{u}^h, \mathbf{u}^h) \geq (\|\mathbf{u}^h\|_{\Omega_e} - C_I \|\llbracket \mathbf{u}^h \rrbracket\|_{\Gamma_*^e})^2 + (\alpha_e - C_I^2) \|\llbracket \mathbf{u}^h \rrbracket\|_{\Gamma_*^e}^2. \quad (3.31)$$

From (3.31), we can ensure coercivity if  $\alpha_e \geq C_I^2$ .

We use the inverse inequality (3.30) to bound the mesh-dependent parameter  $C_I^2$  from below. Squaring both sides of (3.30) and employing direct notation:

$$\|\mathbf{n}^T \langle \mathbf{D} \epsilon \rangle_\gamma\|_{\Gamma_*^e}^2 \leq C_I^2 \|\mathbf{u}^h\|_{\Omega_e}^2, \quad (3.32)$$

where, we denote the discrete counterpart of the elasticity tensor by  $\mathbf{D}$  and strain by  $\epsilon$ . For constant strain triangular and tetrahedral elements, assuming  $\mathbf{D}$  remains piecewise constant within each element, we can write

$$\begin{aligned} \|\mathbf{u}^h\|_{\Omega_e}^2 &= \text{meas}(\Omega_e^1) |(\epsilon^1)^T \mathbf{D}^1 \epsilon^1| + \text{meas}(\Omega_e^2) |(\epsilon^2)^T \mathbf{D}^2 \epsilon^2|, \\ &\leq \text{meas}(\Omega_e^1) |(\epsilon^1)^T| |\mathbf{D}^1| |\epsilon^1| + \text{meas}(\Omega_e^2) |(\epsilon^2)^T| |\mathbf{D}^2| |\epsilon^2|, \\ &\leq \text{meas}(\Omega_e^1) |\mathbf{D}^1| |\epsilon^1|^2 + \text{meas}(\Omega_e^2) |\mathbf{D}^2| |\epsilon^2|^2, \end{aligned} \quad (3.33)$$

where we have denoted a 2-norm of a matrix by  $|(\cdot)|$ . The first line follows from the definition of the energy norm (3.26) and the second and third lines from standard properties of matrix norms. Similarly, for the average traction at the interface, we write

$$\begin{aligned} \|\mathbf{n}^T \langle \mathbf{D} \epsilon \rangle_\gamma\|_{\Gamma_*^e}^2 &= \text{meas}(\Gamma_*^e) |\mathbf{n}^T \langle \mathbf{D} \epsilon \rangle_\gamma|^2, \\ &\leq \text{meas}(\Gamma_*^e) |\mathbf{n}^T|^2 |\langle \mathbf{D} \epsilon \rangle_\gamma|^2, \\ &\leq \text{meas}(\Gamma_*^e) ((\gamma^1)^2 |\mathbf{D}^1|^2 |\epsilon^1|^2 (1 + \beta) + (\gamma^2)^2 |\mathbf{D}^2|^2 |\epsilon^2|^2 (1 + 1/\beta)), \end{aligned} \quad (3.34)$$

where, the first line follows from the definition of the  $L_2$  norm (3.26), and the third line from the definition of a unit normal and on employing Young's inequality with  $\beta$ . Now on choosing

$$\beta = \frac{\text{meas}(\Omega_e^1) |\mathbf{D}^2| (\gamma^2)^2}{\text{meas}(\Omega_e^2) |\mathbf{D}^1| (\gamma^1)^2},$$

$$\|\mathbf{n}^T \langle \mathbf{D} \epsilon \rangle_\gamma\|_{\Gamma_*^e}^2 \leq \text{meas}(\Gamma_*^e) \left( \frac{|\mathbf{D}^1|(\gamma^1)^2}{\text{meas}(\Omega_e^1)} + \frac{|\mathbf{D}^2|(\gamma^2)^2}{\text{meas}(\Omega_e^2)} \right) (\text{meas}(\Omega_e^1) \|\mathbf{D}^1\| \epsilon^1|^2 + \text{meas}(\Omega_e^2) \|\mathbf{D}^2\| \epsilon^2|^2). \quad (3.35)$$

Thus, from (3.35) and (3.33), for constant strain triangular and tetrahedral elements the generalized inverse estimate is satisfied for

$$C_I^2 \geq \text{meas}(\Gamma_*^e) \left( \frac{|\mathbf{D}^1|(\gamma^1)^2}{\text{meas}(\Omega_e^1)} + \frac{|\mathbf{D}^2|(\gamma^2)^2}{\text{meas}(\Omega_e^2)} \right). \quad (3.36)$$

Based on parametric studies conducted in Embar et al. (2010), for good performance in computations we choose  $\alpha_e = 2C_I^2$ , where  $C_I^2$  is given by the equality sign in equation (3.36). A choice of  $\gamma^1 = \gamma^2 = 0.5$  returns an identical result as that proposed in Dolbow and Harari (2009) for scalar elliptic problems.

For arbitrary orientations of the interface, it is not inconceivable to have elements where either  $\text{meas}(\Omega_e^1)$  or  $\text{meas}(\Omega_e^2)$  approach a very small value. In such cases, the stabilization parameter is not bounded from above. This can have an adverse effect on the conditioning of the system for quasi-static problems as reported in Chapter 2 and also in several earlier studies (see, for example, Lew and Buscaglia, 2008; Embar et al., 2010).

For dynamic problems, as illustrated further in Section 3.4, a large stabilization parameter also places severe constraints on the stable time step required for time stepping schemes. In addition, the coercivity of the bilinear form is also quite sensitive to the estimate provided above and naively choosing a smaller value of the stabilization parameter for such elements by replacing  $\text{meas}(\Omega_e^1)$  and  $\text{meas}(\Omega_e^2)$  by  $\text{meas}(\Omega_e)$ , for instance, upsets the positive-definiteness of the discrete system. In certain ways, this is analogous to discontinuous Galerkin methods where the choice of numerical flux has great bearing on the stability and consistency properties and consequently the convergence of the chosen method (Arnold et al., 2002).

Ideally, one would want a method where all the terms of the stiffness matrix,

including the classical and interfacial terms, grow at a comparable rate. Thus, for the sake of robustness, we aim to arrive at an estimate for the stabilization parameter which varies inversely with the size of the total element as opposed to that of a partial element. To do this, we leverage the freedom we have in choosing the weighting  $\gamma^m$ . Hence, if we choose  $\gamma^1 = \text{meas}(\Omega_e^1)/\text{meas}(\Omega_e)$  in (3.36) we obtain

$$C_I^2 \geq \frac{\text{meas}(\Gamma_*^e)}{(\text{meas}(\Omega_e))^2} (|\mathbf{D}^1|_{\text{meas}(\Omega_e^1)} + |\mathbf{D}^2|_{\text{meas}(\Omega_e^2)}). \quad (3.37)$$

Clearly,  $\alpha_e$  now varies inversely with element size and grows at the same rate as the classical finite element stiffness terms. We contend that this choice (3.37) is much more robust than existing alternatives.

Finally, we end by remarking that for higher order elements, owing to non-constant flux within an element an analytical estimate for the stabilization parameter is not immediately obvious. However, one could still qualitatively demonstrate the dependence of the stabilization parameter on the weights. For such elements, then, one can perform an additional element level eigenvalue calculation to arrive at the stabilization parameter as demonstrated in Embar et al. (2010).

## 3.4 Temporal Discretization

### 3.4.1 Stability analysis

For temporal integration of the semi-discrete equations, we employ the Newmark family of methods with particular focus on the central difference algorithm. The critical time step for the central difference algorithm is bounded from above by  $2/\omega_{max}$ , where,  $\omega_{max}$  is the maximum natural frequency, or the maximum eigenvalue given by the generalized eigenvalue problem

$$(\mathbf{K} - \omega^2 \mathbf{M})\mathbf{u} = \mathbf{0}. \quad (3.38)$$

*Dual approach:*

For the dual method, Carpenter et al. (1991) noticed that a purely explicit strategy returns an update scheme where the displacement at  $t_{n+1}$  is completely independent of traction at the interface at  $t_{n+1}$  and hence results in a singular system of equations. To counter this, they employed the implicit/explicit forward increment Lagrange multiplier algorithm to integrate the equations of motion (3.16) in time. The essential idea of the forward increment Lagrange multiplier algorithm is to build in the constraint from time step  $t_{n+1}$  into the time step  $t_n$ , and then integrate the resulting equations of motion using the central difference algorithm.

The temporal stability requirements now dictate that the stable time step for time integration is given by  $2/\omega_{max}^c$ , where  $\omega_{max}^c$  is the maximum frequency of the constrained system. However, Rayleigh's nested eigenvalue theorem (see Belytschko et al., 2000) states that the maximum eigenvalue of a constrained system is always bounded by the maximum eigenvalue of an unconstrained system. Therefore,  $\omega_{max}^c \leq \omega_{max}$ , where  $\omega_{max}$  is the maximum frequency for an unconstrained system. The eigenvalue problem associated with the constrained system (3.16) is given by

$$\begin{bmatrix} (\mathbf{K} - \omega^2\mathbf{M}) & \mathbf{G} \\ \mathbf{G}^T & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ \lambda \end{bmatrix} = \mathbf{0}. \quad (3.39)$$

Hence, for the dual method, the stability analysis undertaken in Menouillard et al. (2008) for unconstrained enriched elements is still valid and we employ the same lumping scheme as used there. A brief description of the said lumping scheme is provided below before conducting a stability analysis for the primal approaches.

*Primal approaches:*

For primal approaches, on the other hand, the additional contributions to the stiffness matrix due to the penalty and consistency expressions in the variational form alter



the bounds on the critical time step. A rather conservative yet standard way of obtaining the critical time step is to instead solve (3.38) at an element level. We can therefore concentrate on analyzing a single one-dimensional bar element with a material interface in order to gain a better understanding of the effect of the additional stiffness terms on the critical time step.

For illustration, let us consider a one-dimensional element having length  $l_e$  and a unit cross-sectional area with a material interface at a distance of  $\xi l_e$  from the left node, where  $\xi \in [0, 1]$  is the volume fraction of the cut element in  $\Omega^1$ . The overlapping bar elements and the truncated basis are as shown in Figure 3.3(a) and Figure 3.3(b). For simplicity, in the theoretical investigation, we assume that the element has same material properties (modulus of elasticity  $E$  and density  $\rho$ ) on both sides of the interface and perform the stability analysis only for the penalty method. However, the conclusions we draw out will also hold for Nitsche’s method.

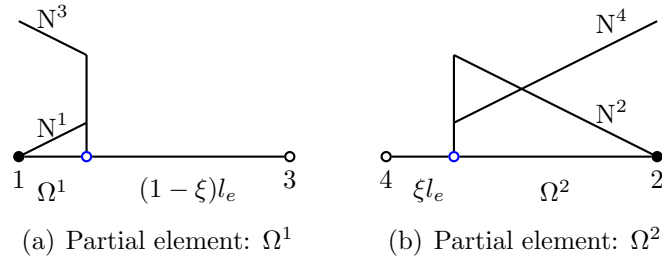


FIGURE 3.3 – Overlapping element formulation for a one-dimensional bar element. The black circles are the physical nodes corresponding to the background mesh and the hollow circles are the ghost nodes. The blue circle represents the embedded interface.

In order to proceed with the temporal stability analysis, a first step is to decide on a mass lumping procedure to diagonalize the mass matrix. As the discretization can be viewed as one that is obtained by superposing the fields from both sides of the interface, it is natural to think of a lumping procedure which lumps the mass in both the superposing elements in the ratio of their physical masses. This was the idea pursued in Menouillard et al. (2008). They showed that for a discontinuous

element such a procedure yields a stable time step equal to that of a standard finite element. Following the same approach, and adding the contribution from penalty terms, the stiffness and mass matrices can be written as:

$$\mathbf{K} = \begin{bmatrix} \xi \mathbf{K}^{\text{FEM}} + \mathbf{K}^{\text{Pen}} & -\mathbf{K}^{\text{Pen}} \\ -\mathbf{K}^{\text{Pen}} & (1 - \xi) \mathbf{K}^{\text{FEM}} + \mathbf{K}^{\text{Pen}} \end{bmatrix}; \quad \mathbf{M} = \begin{bmatrix} \xi \mathbf{M}^{\text{FEM}} & \mathbf{0} \\ \mathbf{0} & (1 - \xi) \mathbf{M}^{\text{FEM}} \end{bmatrix}. \quad (3.40)$$

where,  $\mathbf{K}^{\text{FEM}}$  and  $\mathbf{M}^{\text{FEM}}$  are the element stiffness and mass matrices for a classical bar element and using (3.25) and (3.24):

$$\mathbf{K}^{\text{Pen}} = \alpha \begin{bmatrix} (1 - \xi)^2 & (1 - \xi)\xi \\ \xi(1 - \xi) & \xi^2 \end{bmatrix}. \quad (3.41)$$

Now, clearly, the critical time step will depend on the location of the interface ( $\xi$ ) as well as the penalty parameter ( $\alpha$ ).

Firstly, in order to study the variation of the critical time step with the interface location within an element more clearly, we make a choice for the penalty parameter as  $\alpha = E/l_e$ . This choice ensures that all the terms contributing to the stiffness matrix grow at a comparable rate. We now solve the generalized eigenvalue problem at an element level by setting:

$$\det(\mathbf{M}^{-1}\mathbf{K} - \omega^2\mathbf{I}) = 0. \quad (3.42)$$

For the given stiffness and mass matrices, we get:

$$\omega^2 = \left\{ 0, \frac{2E}{l_e^2 \rho (1 - \xi)}, \frac{2E}{l_e^2 \rho \xi}, \frac{4E}{l_e^2 \rho} \right\}. \quad (3.43)$$

We can clearly see from equation (3.43) that as the interface approaches either node, i.e. as  $\xi \rightarrow 0$  or  $1$ , the maximum eigenvalue  $\omega_{\max} \rightarrow \infty$  and hence results in a null

critical time step. As a result, some of the standard mass lumping schemes developed for extended finite element methods fail to be robust when constraints are added to the interface (or crack surface).

### 3.4.2 Mass lumping approach based on kinetic energy conservation

An interpretation of the penalty method is to consider the penalty stiffness as infinitely stiff, massless springs at the interface. Hence, as the interface approaches a node, we are allowing the element mass to go to zero and at the same time causing a large increase in its stiffness. Therefore, we no longer allow the mass and stiffness matrices to scale similarly and end up reintroducing the problem of null critical time step. One approach to overcome this problem would be to assign part of the total element mass to the spring that constrains the kinematics at the interface and then use this mass to our advantage to prevent the element mass matrices of the two overlapping elements from becoming infinitesimal.

Using the above ideas, we aim to arrive at a lumped mass matrix, that preserves the equivalence between the kinetic energies of the discrete and continuous systems for rigid body modes as well as affine expansions. The kinetic energy expressions for the discrete and continuous systems can be given by:

$$T_{\text{disc}} = \frac{1}{2} \dot{\mathbf{d}}^T \mathbf{M} \dot{\mathbf{d}} \quad \text{and} \quad T_{\text{cont}} = \frac{1}{2} \int_{\Omega_e} \rho \dot{\mathbf{u}}^2 \, d\Omega, \quad (3.44)$$

where,  $\dot{\mathbf{d}}$  is the vector of nodal velocities and  $\mathbf{M}$  represents the lumped mass matrix with diagonal entries  $m_1, m_2, m_3$  and  $m_4$ .

*Affine expansion:* Let us first consider a case when the two sub-domains are trying to separate out from each other at the interface. This can be represented by an affine expansion of the bar about its center with the nodes moving away from the

center at a constant velocity. For this case, the discrete and exact kinetic energy can be written as:

$$T_{\text{disc}} = \frac{1}{2}(m_1 + m_3)(-v)^2 + \frac{1}{2}(m_2 + m_4)v^2 \quad \text{and} \quad T_{\text{cont}} = \frac{1}{2}\rho l_e v^2. \quad (3.45)$$

On assuming  $m_1 = m_2 = m_3 = m_4 = m_d$ , we can say that  $m_d = \frac{\rho l_e}{4}$ .

*Rigid body motion:* Now let us consider a case when the bar is exhibiting a rigid body motion at a constant velocity. The kinetic energy for the discrete and continuous systems is now given by:

$$T_{\text{disc}} = \frac{1}{2}(m_1 + m_2 + m_3 + m_4)v^2 \quad \text{and} \quad T_{\text{cont}} = \frac{1}{2}\rho l_e v^2. \quad (3.46)$$

Again on assuming that the diagonal entries are equal, we get  $m_d = \frac{\rho l_e}{4}$ . Therefore, the lumped mass matrix is now given by

$$\mathbf{M} = \frac{\rho l_e}{4} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (3.47)$$

In other words, we are now distributing the total element mass equally among all the four diagonal entries of the mass matrix regardless of the interface position, thereby preventing the difficulties associated with infinitesimal nodal masses.

On solving (3.42) with the proposed mass matrix, we get:

$$\omega^2 = \left\{ 0, \frac{6E - f_1 - f_2}{l_e^2 \rho}, \frac{6E - f_1 + f_2}{l_e^2 \rho}, \frac{4E}{l_e^2 \rho} \right\}, \quad (3.48)$$

where,  $f_1 = 8E\xi(1 - \xi)$  and  $f_2 = 2E(1 - 2\xi)\sqrt{(2\xi - 1)^2 + 8}$ . If we denote the critical time step for this case by  $\Delta t_{cr}$  and denote the critical time step of a standard finite

element by  $\Delta t_{cr}^{\text{FEM}}$ , from (3.48) we can say that for the proposed lumping scheme

$$\Delta t_{cr} \in \left( \frac{\Delta t_{cr}^{\text{FEM}}}{\sqrt{3}}, \Delta t_{cr}^{\text{FEM}} \right) \forall \xi. \quad (3.49)$$

The ideas presented above can easily be extended to two and three dimensions as well, as shown by the numerical performance of the lumping scheme in Figure 3.4. In Figures 3.4(a) and 3.4(b), we compare the performance of the two lumping procedure proposed here against the one proposed in Menouillard et al. (2008) for a one-dimensional bar element and a constant strain tetrahedral element respectively. In both cases, we normalize the critical time step with that of a standard finite element and plot it against the interfacial location.

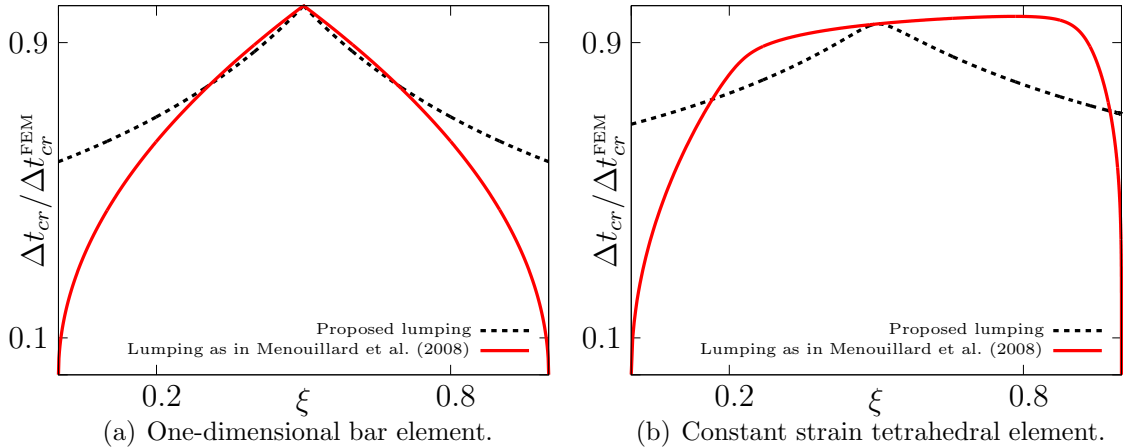


FIGURE 3.4 – Variation of normalized critical time step with the volume fraction of the cut element in  $\Omega^1$  for the two lumping schemes.

It is also useful to look at the critical time step, solely as a function of the penalty parameter. To this end, we choose  $\xi = 0.5$  and again solve the generalized eigenvalue problem given by (3.42) with the proposed mass matrix. Now we find that,

$$\omega^2 = \left\{ 0, \frac{4\alpha}{l_e \rho}, \frac{4E}{l_e^2 \rho}, \frac{4E}{l_e^2 \rho} \right\}. \quad (3.50)$$

It is easy to see from (3.50) that a high value of the penalty parameter has an adverse effect on the critical time step. From (3.36), on using the standard form of Nitsche’s method with  $\gamma^1 = \gamma^2 = 0.5$ , the stabilization parameter for a cut element as considered in Figure 3.3(a) and 3.3(b), is shown to vary inversely with the volume fraction of the element on the positive and negative side, *i.e.*  $\xi$  and  $(1 - \xi)$ . Therefore, for the cases when the interface is oriented such that  $\xi \rightarrow 0$  or  $1$ , the stabilization parameter tends to a large value. Now, clearly, from (3.50) this will severely restrict the critical time step and make explicit calculations prohibitively expensive. On the other hand, the suggested weighted stress approach (3.37) yields a stabilization parameter which remains bounded regardless of the orientation of the interface, causing only a nominal decrease in the stable time step.

### 3.5 Numerical Examples

In this section, we compare and contrast the performance of embedded methods against the classical finite element method on a series of wave-propagation test problems. Throughout the section and in the Figure legends, we denote the weighted version of Nitsche’s method by  $\gamma$ -Nitsche and the dual method by LMVV. For the dual method, we use the forward increment Lagrange multiplier algorithm to integrate the semi-discrete equations of motion in time while for penalty and weighted Nitsche’s method, we use the classical central difference algorithm in conjunction with the mass lumping proposed in Section 3.4. On employing the standard form of Nitsche’s method, we use the implicit/explicit mesh partitioning algorithms of Hughes and Liu (1978) and use the Newmark family of time integration schemes with the parameters  $\beta = 0.25$  and  $\gamma = 0.5$  for the implicit elements and the classical central difference algorithm for the explicit elements.

In addition for the penalty method, unless otherwise specified, we use a penalty parameter of  $E/h$ , where  $E$  is the Young’s modulus and  $h$  is the characteristic mesh

size. This choice allows the penalty terms in the stiffness matrix to grow at the same rate as the other terms allowing us to circumvent any ill-conditioning issues and resulting in a critical time step of the same order of magnitude as one obtained using dual and the consistent primal approaches. It must however be emphasized that such an approach cannot always be taken with the penalty method and the results presented here can be seen as the most favorable set returned by the penalty method.

Finally, as a reference for our studies, we solve all the examples using the classical finite element method with mesh lines explicitly fitted to the interface. We plot the time evolution of displacement and stress, at a given spatial location, and compare the results obtained from the embedded methods with those obtained using the classical finite element method. As a measure of constraint enforcement, we plot the error in the displacement and stress jump across the embedded surface against time for all three embedded methods. To evaluate the accuracy of the constraint enforcement, we employ the  $L_2$  error norm given by:

$$\mathcal{E} = \left( \int_{\Gamma_*} ([[\cdot]]^h - [[\cdot]]^{ex})^2 d\Gamma \right)^{\frac{1}{2}}, \quad (3.51)$$

where,  $[[\cdot]]^h$  denotes a jump in the quantity  $(\cdot)$  constructed using the finite element approximation and  $[[\cdot]]^{ex}$  denotes the exact value of the jump at the interface.

### 3.5.1 Impact loading

We first consider a model elastodynamics problem with an embedded material interface  $\Gamma_*$  such that:

$$\begin{aligned} \nabla \cdot \boldsymbol{\sigma}^1 &= \rho^1 \ddot{\mathbf{u}}^1 & \text{in } \Omega^1 \times (0, T) &= \{\mathbf{x} : \psi(\mathbf{x}) > 0\}, \\ \nabla \cdot \boldsymbol{\sigma}^2 &= \rho^2 \ddot{\mathbf{u}}^2 & \text{in } \Omega^2 \times (0, T) &= \{\mathbf{x} : \psi(\mathbf{x}) < 0\}, \\ \mathbf{u} &= 0 & \text{on } \Gamma_D \times (0, T) &= \{\mathbf{x} : z = 1\}, \\ \boldsymbol{\sigma} \cdot \mathbf{n} &= p(t) & \text{on } \Gamma_N \times (0, T) &= \{\mathbf{x} : z = 0\}, \\ \boldsymbol{\sigma}^1 \cdot \mathbf{n}^1 &= -\boldsymbol{\sigma}^2 \cdot \mathbf{n}^2 & \text{on } \Gamma_* \times (0, T) &= \{\mathbf{x} : \psi(\mathbf{x}) = 0\}, \\ \mathbf{u}^1 &= \mathbf{u}^2 & \text{on } \Gamma_* \times (0, T) &= \{\mathbf{x} : \psi(\mathbf{x}) = 0\}. \end{aligned} \quad (3.52)$$

*Planar surface*

As a first example, we consider an embedded planar interface given by  $\psi(\mathbf{x}) = z - 0.4856$ . We consider different material properties on each side of the interface such that in the sub-domain  $\Omega^1$ , the material is characterized by a Young’s modulus  $E^1$  of 200 GPa, Poisson’s ratio  $\nu^1$  of 0.3 and mass density  $\rho^1 = 7800 \text{ kg/m}^3$  whereas in the sub-domain  $\Omega^2$  we have a Young’s modulus  $E^2$  of 70 GPa, Poisson’s ratio  $\nu^2$  of 0.3 and mass density  $\rho^2 = 2700 \text{ kg/m}^3$ . We simulate the response of such a “segmented bar” to a sudden impact loading such that  $p(t) = \sigma_0 \mathcal{H}(t)$  where,  $\sigma_0 = 1$  MPa and  $\mathcal{H}$  is the Heaviside function. All embedded simulations are carried out on an unstructured constant strain tetrahedral mesh, with 8 divisions along each direction, for a total simulation time of  $T = 5 \text{ ms}$ .

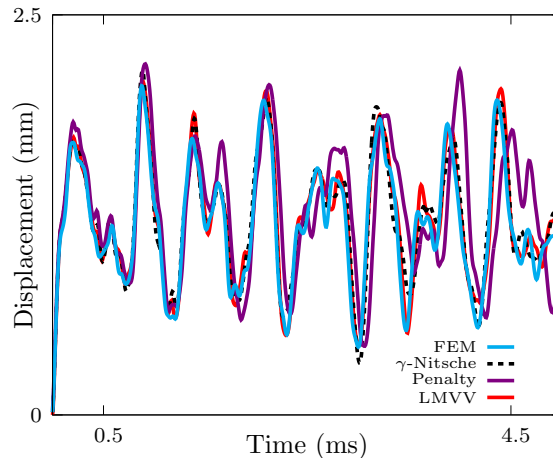


FIGURE 3.5 – Comparison of time evolution of displacement for conforming and embedded methods.

Figure 3.5 shows the evolution of displacement with time for all the three embedded methods as well as the classical finite element method at the spatial location  $(0,1,0)$ . From the plot, we can see that the curves compare well for all three methods. It is also evident from the plots that the variationally consistent methods have a closer agreement with the classical finite element method than the penalty method.



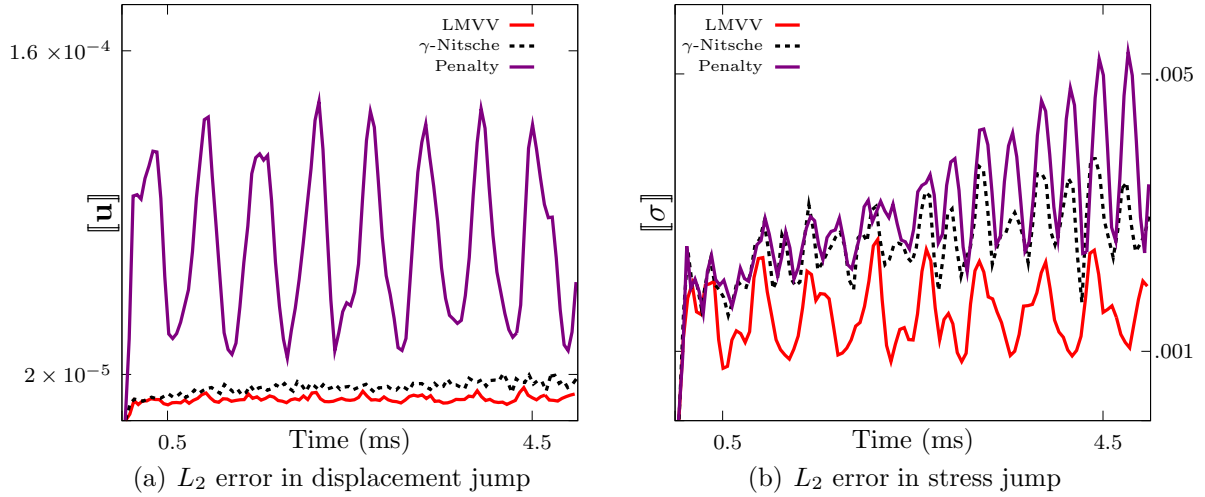


FIGURE 3.6 – Variation of  $L_2$  error in the constraint with time.

Figures 3.6(a) and 3.6(b) plot the evolution of  $L_2$  error in the jump constraints with time. Again, it is clear from the plots that for a comparable value of penalty parameter, the constraint is imposed much more accurately with the variationally consistent methods. The stable Lagrange multiplier formulation seems to enforce the constraint in stress jump with higher accuracy than Nitsche’s method.

It is possible to tune the penalty parameter to gain accuracy with the penalty method but apart from the ill-conditioning issues seen in quasi-static problems, this also adversely affects the critical time step. Also, since in most real problems, we have no *a priori* information on the solution, we can no longer afford the luxury of tuning the parameter. Additionally, as can be seen from Figure 3.7, a large value of the penalty parameter triggers non-physical oscillations in the traction field at the interface in the same way as was seen earlier for quasi-static problems (Sanders et al., 2009) as well as dynamic problems (Simone, 2004). It is interesting to note that even the standard form of Nitsche’s method seems to exhibit minor oscillations, and we suspect these again arise from the sensitivity of the stabilization parameter to the interface location. Though it is possible to smooth these out with the domain integral

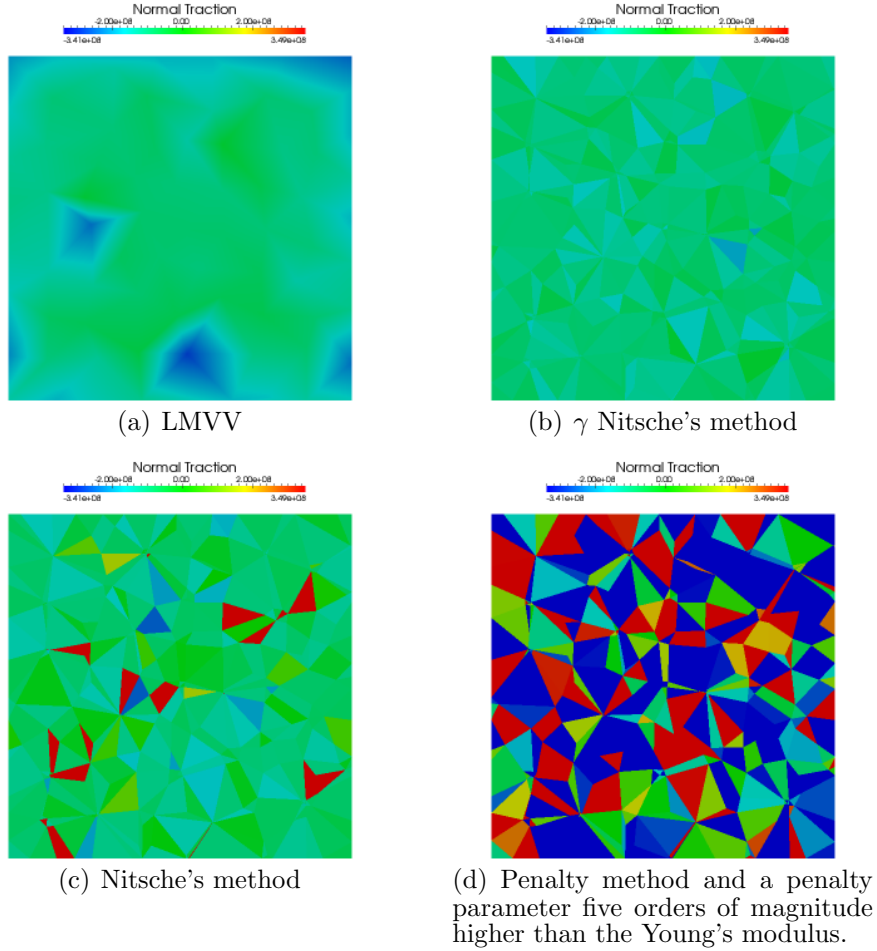


FIGURE 3.7 – Plot of normal traction at the embedded surface using all the three embedded approaches at a time after the wave first crosses the embedded surface.

formulation presented in Ji and Dolbow (2004), from Figure 3.7(b) the weighted form of Nitsche's method seems much less susceptible to these oscillations, further highlighting the robustness of the method.

### *Sensitivity analysis*

We modify the above model problem such that the domain has identical material properties, a Young's modulus  $E = 200$  GPa, a Poisson's ratio  $\nu = 0.3$  and a mass density of  $\rho = 7800 \text{ kg/m}^3$ , on both sides of the interface. We now study the sensitivity exhibited by an element level estimate of the critical time step to the

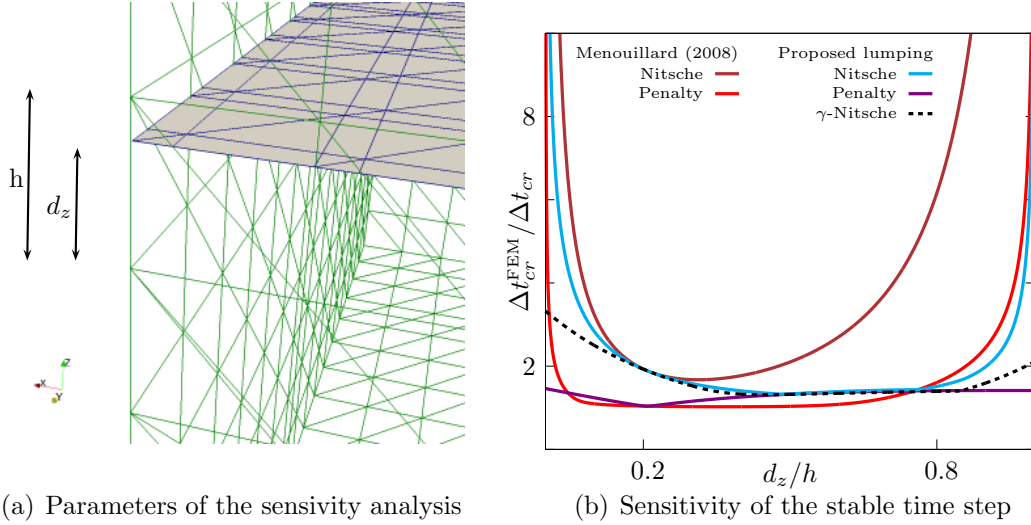


FIGURE 3.8 – Illustration of the sensitivity of the critical time step with the interface location exhibited by various primal embedded approaches.

interface location. For this purpose, we vary the interface location with respect to a given structured background mesh. An illustration of the parameters used in the sensitivity analysis is given in Figure 3.8(a).

We normalize the critical time step obtained from the various embedded approaches with that of classical finite element method and plot its variation as a function of the interface location in Figure 3.8(b). From the plot it is clear that the mass lumping proposed in Menouillard et al. (2008) breaks down as the interface approaches either internal boundary for all the primal approaches. Even with the proposed lumping technique, for the standard form of Nitsche’s method, the large stabilization terms increase the spectral radius of the stiffness matrix resulting in a prohibitively small time step. Clearly from the figure, the only two approaches that do not result in a substantial reduction in the time step are the penalty method and the weighted form of Nitsche’s method as suggested in Section 3.3, both used in conjunction with the proposed lumping technique.

*Popcorn shaped surface*

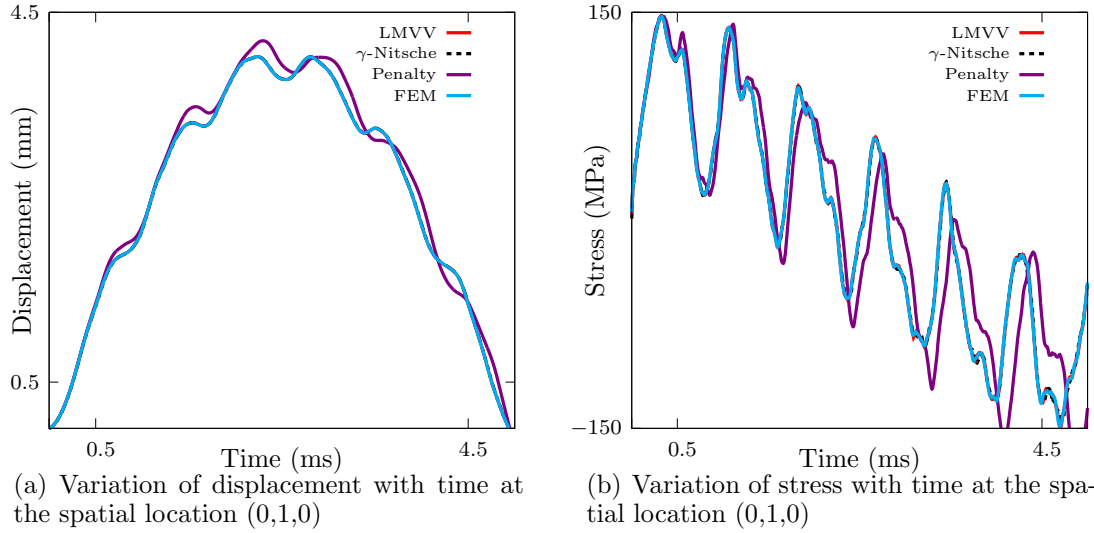


FIGURE 3.9 – Comparison of time evolution of displacement and stress for a sinusoidal impact load applied at the bottom surface obtained using classical and embedded approaches for the case with an embedded popcorn shaped surface.

We further illustrate the utility of the embedded approach on a domain with a popcorn shaped embedded surface defined by the implicit equations described previously in Chapter 2. To validate the results, we solve the same problem with the classical finite element method as well. On using the classical finite element method, to avoid the difficulties associated with explicitly gridding this complex surface, we consider identical material properties on both sides of the surface, given by  $E = 200$  GPa,  $\nu = 0.3$  and  $\rho = 7800$  kg/m<sup>3</sup>. To ensure that the embedded surface is discretized well enough, we choose a structured mesh with 24 divisions along each direction. The Neumann boundary is loaded by a time dependent sinusoidal traction such that  $p(t) = \sigma_0 \sin(\pi t/T)$  where  $\sigma_0 = 1$  MPa and we simulate the response for a total time of  $T = 5$  ms.

The variation of displacement and stress with time is plotted in Figures 3.9(a) and 3.9(b) respectively. We also plot a time snapshot of the displacement contour obtained using classical and embedded approaches in Figure 3.10. As before, excellent

agreement is observed between the classical and the embedded approaches and indicates that the overall numerical performance remains the same for both approaches.

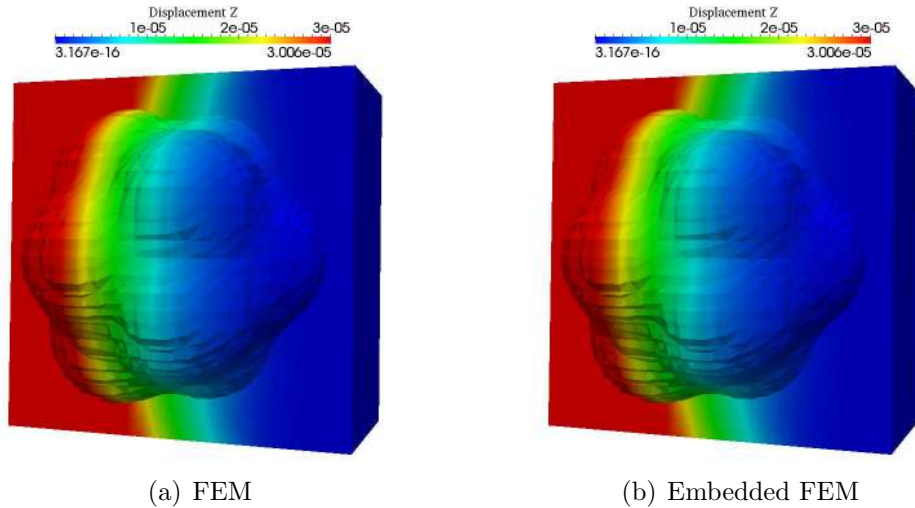


FIGURE 3.10 – Time snapshot of the displacement contour for classical and embedded approaches with an embedded popcorn shaped surface.

### *Material with microstructure*

As a final example, we investigate a wave propagation problem on a domain with complex microstructural inclusions in a similar vein to the one described in Moës et al. (2003). It can be appreciated that, for such problems, an embedded approach would be far more computationally efficient than an explicitly gridded approach, especially for multiscale problems.

Here, we consider a material filled with 32 spherical inclusions as shown in Figure 3.11. We distribute the centers of the spheres randomly within a unit cube while ensuring that no two spheres intersect. We assign a Young’s modulus of  $E^2 = 70$  GPa, Poisson’s ratio  $\nu^2 = 0.3$  and mass density  $\rho^2 = 2700kg/m^3$  to the spheres and a Young’s modulus of  $E^1 = 200$  GPa, Poisson’s ratio  $\nu^1 = 0.3$  and mass density  $\rho^1 = 7800kg/m^3$  to the matrix surrounding the spheres. We load the bottom surface

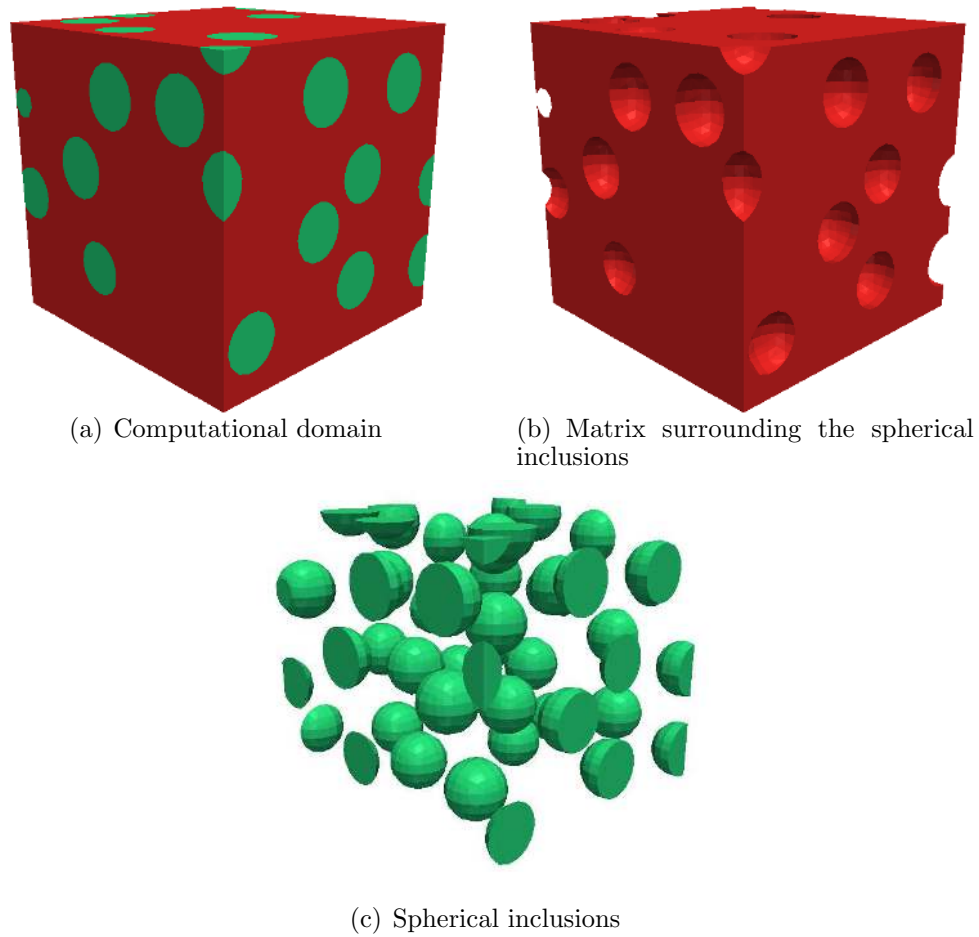


FIGURE 3.11 – Illustration of the microstructural geometry and computational domain with 32 randomly distributed spherical inclusions.

with a sinusoidal traction as described above for the popcorn shaped surface. We again simulate the response of the material for a total time of  $T = 5 \text{ ms}$ . Since it is evident from our earlier examples that the variationally consistent methods provide better accuracy, we conduct this simulation only with the two consistent approaches and compare our results with the classical FEM. The variation of displacement with time and stress with time is plotted in Figures 3.12(a)-(b) respectively. The time snapshot of the displacement contour at a comparable time in the simulation is also provided for the embedded as well as the classical FEM in Figure 3.13. As seen from Figures 3.12(a)-(b), the embedded methods provide identical results, we thus provide

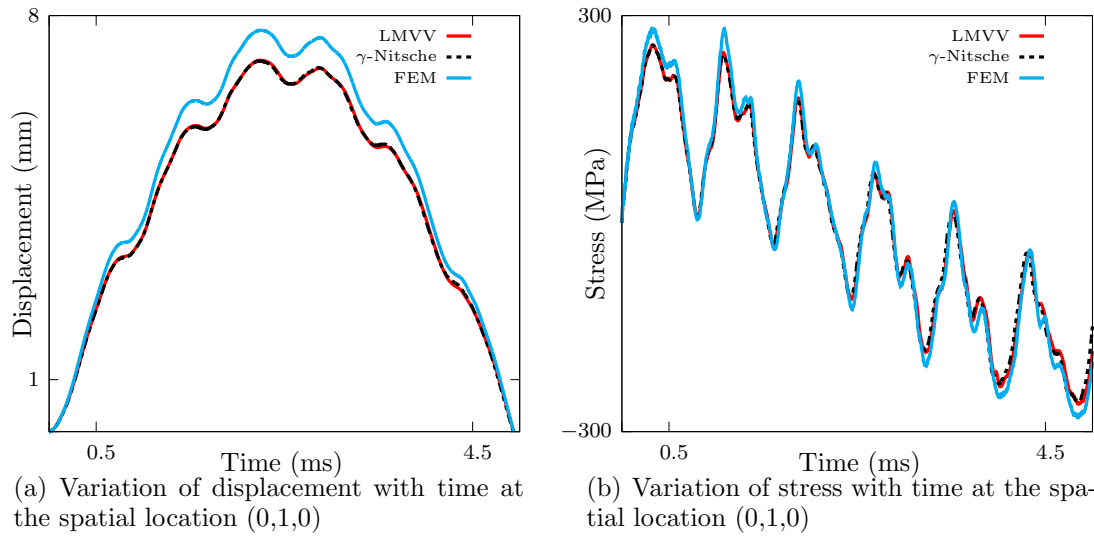


FIGURE 3.12 – Comparison of time evolution of displacement and stress for a sinusoidal impact load applied at the bottom surface obtained using classical and embedded approaches for a cubic domain filled with microstructural inclusions.

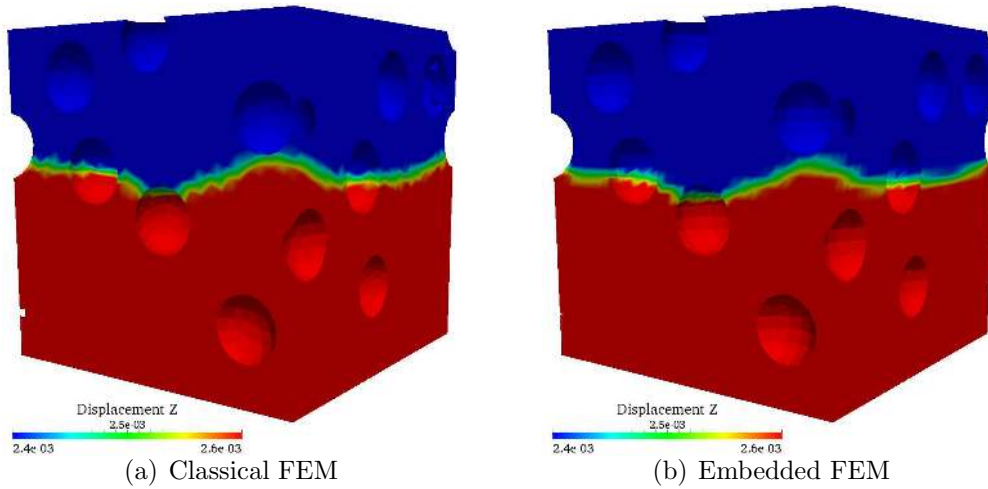


FIGURE 3.13 – Time snapshot of the displacement contour for classical and embedded approaches with an embedded popcorn shaped surface.

a single snapshot representing both the embedded methods. For better visualization, we only show the displacement contours on the matrix surrounding the spheres and have clipped out the spherical inclusions. From the plots, we see that the embedded methods agree fairly well with the classical finite element method.

The amplitude of the wave for embedded methods differs from the classical method by nearly 7%. However, we attribute the marginal difference in the results to a largely different discretization. For the embedded methods, we use a uniform discretization with 30 divisions along each axis resulting in a mesh with 186000 elements. For the classical finite element method, we generate a conforming mesh using the open source meshing software NETGEN. The discretization is therefore significantly different as the conforming mesh is optimized and refined more in the vicinity of the curved interfaces. The conforming mesh discretizes the domain into 77578 tetrahedrons.

It is also worth remarking that the CFL condition provides a stable time step of  $8.7 \times 10^{-7}$  s for the classical finite element method whereas for the embedded methods the stable time step is given as  $2.8 \times 10^{-6}$  s for LMVV and  $9.6 \times 10^{-7}$  s for the weighted Nitsche's method. In essence, the stable time step is not restricted any worse by the embedded methods than a corresponding fitted grid required for the classical FEM which can sometimes result in uncharacteristically small elements. This feature, coupled with the flexibility the embedded methods provide, make them a very attractive prospect for solving this class of problems.

### 3.6 Conclusion

In this Chapter, we investigated the performance of various embedded interface methods for explicit dynamics simulations. We highlighted the severe restrictions placed by the CFL stability conditions on the stable time step for the two primal methods on using the existing mass lumping schemes. As an alternative, we proposed a novel



mass lumping strategy based on conserving kinetic energy for rigid body modes and affine expansions. We showed that on using the proposed lumping scheme, the stable time step always remains bounded for the penalty method regardless of the orientation of the interface. We also highlighted the incompatibility of the standard form of Nitsche’s method with explicit time stepping schemes and proposed an alternative variational form.

We then compared the performance of the embedded methods with the classical finite element method on several benchmark wave-propagation examples. We observed excellent agreement with the classical finite element method for all our numerical examples. The advantages of variationally consistent methods observed in the quasi-static regimes - better constraint enforcement and greater stability in the traction profile - are still evident in the transient regimes.

However, for the stable Lagrange multiplier method, this robustness and accuracy comes at the price of incompatibility with purely explicit temporal integration. We also note that the stable Lagrange multiplier method involves considerable preprocessing to identify an inf-sup stable multiplier space and the assembly procedure itself is much more expensive. Moreover, for interfaces which evolve in time, the stable Lagrange multiplier method would require this preprocessing at every time step. Due to these considerations, we contend that the weighted Nitsche’s method stands out as the more efficient choice for embedded interface problems.

In addition to its favorable performance in explicit dynamics, we also noticed that the weighted Nitsche’s method yielded a much more stable traction profile compared to classical Nitsche’s method in Section 3.5.1. This raises important questions concerning the effect of weighting on the stability of interfacial quantities of interest. We also recall that in Chapter 2, we noticed a non-convergence of flux in the  $L_\infty$  norm and suspected this resulted from partially filled elements with arbitrarily small volume fractions. In this Chapter, we noticed that it is precisely those ele-

ments which result in the incompatibility of classical Nitsche's method with explicit dynamics. Similar problems were also reported by Burman and Hansbo (2012) and Becker et al. (2011) for fictitious domain methods and Sanders et al. (2012) for mesh mortaring approaches. It remains to be seen whether there is a common underlying cause for these seemingly different problems. We explore this issue further in the next Chapter.

# A Robust Nitsche's Method for Interface Problems

## 4.1 Introduction

In recent years, Nitsche's method, originally proposed in the early 1970s has seen a resurgence. This is primarily due to its efficiency in the treatment of interfacial phenomenon in computational mechanics. The central idea behind Nitsche's approach is to simply replace the Lagrange multipliers arising in a dual formulation through their physical representation, namely the normal flux at the interface. The flexibility of the approach has resulted in a wide range of applications of the method. To name a few, the method is used to enforce inter-element continuity in symmetric interior penalty discontinuous Galerkin methods (Arnold et al., 2002), mesh mortaring (Stenberg, 1998; Laursen et al., 2010; Sanders et al., 2012), enforcing transmission conditions in fictitious domain methods (Hansbo et al., 2003; Becker et al., 2003) and X-FEM type embedded interface methods (Hansbo and Hansbo, 2004; Dolbow and Franca, 2008; Dolbow and Harari, 2009).

While this classical form of Nitsche's method performed optimally in most situations, there were some anomalies. For example, in X-FEM methods, Mourad et al.

(2007), Dolbow and Franca (2008) and Dolbow and Harari (2009), reported high sensitivity of the normal flux when evaluated directly. In fact, in Chapter 2, we showed non-convergence of the normal flux in  $L_\infty$  norm when evaluated directly. In addition, Sanders et al. (2009) reported mild oscillations in the normal traction and heavy oscillatory behaviour in the tangential traction at the interface. Further, in Chapter 3 we noticed the incompatibility of the classical Nitsche formulation with explicit dynamics simulations.

Interestingly, similar issues were also reported with Nitsche based approaches in various other areas. For example, the mortar approaches of Laursen et al. (2010) and Sanders et al. (2012) showed a stress locking pattern when tying a soft material with a stiff one. In discontinuous Galerkin approximations, Lew and Negri (2011) artificially prevented the boundary from cutting arbitrarily small slices of elements. Burman and Hansbo (2012) added a ghost penalty term to improve the conditioning of the discrete system for domains with small sliver pieces. Severe ill-conditioning issues have also been discussed for these fictitious domain approaches by Becker et al. (2011).

In this Chapter, we propose a unified solution to some of those numerical issues. We trace these numerical issues to a common underlying cause - a non-optimal choice of numerical flux. As outlined in the seminal review by Arnold et al. (2002), depending on the choice of numerical flux, a host of discontinuous Galerkin approaches with varying mathematical properties can be derived. We concentrate on “jump” type constraints in particular and propose a modified numerical flux based on a weighting other than a simple arithmetic average. The numerical examples considered here all focus on embedded interfaces but the theory extends to other Nitsche based approaches as well.

The idea of using a weighted form has been tried under different contexts before. Notably, Zunino and co-workers (Zunino, 2009; Burman and Zunino, 2006; Ern et al.,

2009) proposed and analyzed a stiffness weighted interior penalty approach to model the case of vanishing diffusivity in advection-diffusion-reaction equations. The work of Cai et al. (2011) also analyzes a stiffness weighted interior penalty approach for heterogenous problems and establishes robust error estimators. Also in the context of mesh mortaring, Sanders et al. (2012) successfully used a stiffness weighted approach to alleviate stress locking problems exhibited by classical Nitsche’s method. More recently, Zunino et al. (2011) analyzed an unfitted Nitsche method and proposed an alternative that remains robust for the worst case among small cut elements and large heterogeneities.

The novelty of our approach lies in (a) establishing a clear relationship between the weights and the stabilization parameter, and (b) demonstrating the critical dependence of the interfacial quantities of interest on the stabilization parameter. We propose a weighting such that the bulk as well as the interfacial fields remain well behaved in the presence of (a) elements with arbitrarily small volume fractions, (b) large material heterogeneities, and (c) both large heterogeneities as well as arbitrarily small elements. In fact, through a functional relationship between weights and stabilization parameter, we prove that the proposed choice of weights is optimal.

We begin by defining the model problem and the variational form in the next section. In Section 4.3, we discuss the spatial discretization as well as provide a lower bound on the stabilization parameter. Here, we also propose an optimal choice for the weighting parameters by establishing a functional relationship between the stabilization and the weighting parameters of the method. In Section 4.4, we demonstrate the robustness of the proposed method against the classical Nitsche approach on several benchmark examples focusing particular attention on interfacial quantities of interest. Finally, in Section 4.5, we provide concluding remarks and possible extensions.

## 4.2 Model problem and variational formulation

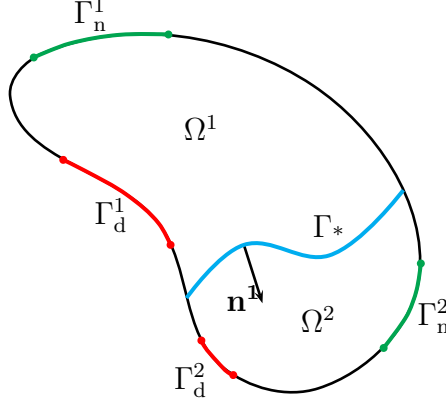


FIGURE 4.1 – Interface  $\Gamma_*$  partitions the bulk domains  $\Omega^1$  and  $\Omega^2$ . The Dirichlet boundaries  $\Gamma_d^1$ ,  $\Gamma_d^2$  and the Neumann boundaries  $\Gamma_n^1$  and  $\Gamma_n^2$  are shown. The complementary part of the boundary is traction free. The normal to the boundary of each domain,  $\mathbf{n}^m$ , points outwards as shown.

We begin by considering a particular Poisson's problem in the bulk domains  $\Omega^1$  and  $\Omega^2$ :

$$\begin{aligned} \nabla \cdot \kappa^m \nabla u^m &= -f^m && \text{in } \Omega^m, \\ u^m &= u_d^m && \text{on } \Gamma_d^m, \\ \kappa^m \nabla u^m \cdot \mathbf{n}^m &= 0 && \text{on } \Gamma_n^m, \end{aligned} \quad (4.1)$$

that are coupled together at the interface by the conditions:

$$\begin{aligned} \llbracket \kappa \nabla u \rrbracket \cdot \mathbf{n}^2 &= \bar{j} && \text{on } \Gamma_*, \\ \llbracket u \rrbracket &= \bar{i} && \text{on } \Gamma_*. \end{aligned} \quad (4.2)$$

where,  $\bar{i}$  and  $\bar{j}$  are sufficiently smooth functions on the interface. The definitions of  $\Omega^1$ ,  $\Omega^2$ ,  $\Gamma_d^m$  and  $\Gamma_n^m$  are as shown in Figure 4.1. The interface  $\Gamma_*$  acts as a partitioning boundary between the two bulk domains  $\Omega^1$  and  $\Omega^2$ . The normals  $\mathbf{n}^m$  are considered

as outward pointing from their corresponding domains and the shorthand notation  $\llbracket u \rrbracket$  corresponds to the jump,  $u^2 - u^1$ .

While the particular focus of our interest in this study is the imposition of the jump conditions given by (4.2), we develop an approach that is equally capable of handling both jump constraints given by (4.2) as well as stiff Neumann constraints. The motivation behind this generalization is to consider a possible extension of this method to frictional sliding problems where a stiff Neumann constraint is more likely to arise. To that end, we consider the following interfacial conditions:

$$\begin{aligned}\kappa^1 \nabla u^1 \cdot \mathbf{n}^1 &= \frac{1}{\epsilon} (\llbracket u \rrbracket - \bar{i}) + \gamma^2 \bar{j} && \text{on } \Gamma_*, \\ \kappa^2 \nabla u^2 \cdot \mathbf{n}^2 &= -\frac{1}{\epsilon} (\llbracket u \rrbracket - \bar{i}) + \gamma^1 \bar{j} && \text{on } \Gamma_*.\end{aligned}\tag{4.3}$$

Physically,  $1/\epsilon$  can be interpreted as the stiffness of cohesive springs arising while modeling a traction-separation law at the interface or the regularization parameter in the Coulomb frictional laws. As can be seen from equation (4.3), in the limiting case of  $\epsilon = 0$ , the Robin type constraints collapse into jump constraints given by equation (4.2). Here  $(\gamma^1, \gamma^2)$  are not problem specific parameters but are real numbers such that  $\gamma^1 + \gamma^2 = 1$ . A possible strategy for identifying these parameters will be discussed in Section 4.3.

#### 4.2.1 Weak form

We begin by defining the solution spaces,  $\mathbb{U} = \mathbb{U}^1 \times \mathbb{U}^2$  and weighting spaces,  $\mathbb{W} = \mathbb{W}^1 \times \mathbb{W}^2$  such that:

$$\begin{aligned}\mathbb{U}^m &= \{u \in H^1(\Omega^m), u = u_d^m \text{ on } \Gamma_d^m\}, \\ \mathbb{W}^m &= \{w \in H^1(\Omega^m), w = 0 \text{ on } \Gamma_d^m\}.\end{aligned}$$

Multiplying the governing equations in bulk by the weighting function, integrating by parts, applying the divergence theorem and summing the resulting equations from

both the bulk domains we have:

$$\sum_m \int_{\Omega^m} \nabla w^m \kappa^m \nabla u^m \, d\Omega - \sum_m \int_{\Gamma_*} w^m \kappa^m \nabla u^m \cdot \mathbf{n}^m \, d\Gamma = \sum_m \int_{\Omega^m} w^m f^m \, d\Omega. \quad (4.4)$$

A similar procedure for the boundary conditions on  $\Gamma_*$  yields:

$$\sum_m \epsilon \int_{\Gamma_*} w^m \kappa^m \nabla u^m \cdot \mathbf{n}^m \, d\Gamma = \int_{\Gamma_*} \llbracket w \rrbracket (\bar{\mathbf{i}} - \llbracket u \rrbracket) \, d\Gamma + \sum_m \epsilon \int_{\Gamma_*} (1 - \gamma^m) w^m \bar{\mathbf{j}} \, d\Gamma. \quad (4.5)$$

The presence of  $\epsilon$  on the highest order term lends a singularly perturbed character to the weighted residual form of the interfacial coupling conditions as given by equation (4.5). For  $\epsilon = 0$  or  $\epsilon \rightarrow 0$ , in such problems we tend to neglect the flux terms arising from the boundary conditions, when in fact they are of a comparable order to the rest of the terms in the equation. Often, in order to regain information on the highest order term in such singularly perturbed problems, there is a need to recast the equations through some rescaling. To that end, we divide equation (4.5) by  $\epsilon + \beta$ , where  $\beta$  is any positive number greater than zero, and obtain:

$$\begin{aligned} \sum_m \frac{\epsilon}{\epsilon + \beta} \int_{\Gamma_*} w^m \kappa^m \nabla u^m \cdot \mathbf{n}^m \, d\Gamma &= \frac{1}{\epsilon + \beta} \int_{\Gamma_*} \llbracket w \rrbracket (\bar{\mathbf{i}} - \llbracket u \rrbracket) \, d\Gamma \\ &+ \sum_m \frac{\epsilon}{\epsilon + \beta} \int_{\Gamma_*} (1 - \gamma^m) w^m \bar{\mathbf{j}} \, d\Gamma. \end{aligned} \quad (4.6)$$

Now, summing the equations (4.4) and (4.6) yields:

$$\begin{aligned} &\sum_m \int_{\Omega^m} \nabla w^m \kappa^m \nabla u^m \, d\Omega - \frac{\beta}{\epsilon + \beta} \sum_m \int_{\Gamma_*} w^m \kappa^m \nabla u^m \cdot \mathbf{n}^m \, d\Gamma + \frac{1}{\epsilon + \beta} \int_{\Gamma_*} \llbracket w \rrbracket \llbracket u \rrbracket \, d\Gamma \\ &= \sum_m \int_{\Omega^m} w^m f^m \, d\Omega + \frac{1}{\epsilon + \beta} \int_{\Gamma_*} \llbracket w \rrbracket \bar{\mathbf{i}} \, d\Gamma + \frac{\epsilon}{\epsilon + \beta} \int_{\Gamma_*} \bar{\mathbf{j}} \langle w \rangle_{1-\gamma} \, d\Gamma. \end{aligned} \quad (4.7)$$

From the interfacial constraints (6.2), the two following expressions can be derived:

$$\begin{aligned} \kappa^1 \nabla u^1 \cdot \mathbf{n}^1 + \kappa^2 \nabla u^2 \cdot \mathbf{n}^2 &= \bar{\mathbf{j}}, \\ \kappa^2 \nabla u^2 \cdot \mathbf{n}^2 &= \langle \kappa \nabla u \rangle_\gamma \cdot \mathbf{n}^2 + \gamma^1 \bar{\mathbf{j}}. \end{aligned} \quad (4.8)$$



Using these two relations, the boundary integral term of weighted fluxes in (4.7) can be re-written as follows:

$$\begin{aligned}
\sum_m \int_{\Gamma_*} w^m \kappa^m \nabla u^m \cdot \mathbf{n}^m \, d\Gamma &= \int_{\Gamma_*} w^1 \bar{j} \, d\Gamma + \int_{\Gamma_*} \llbracket w \rrbracket \kappa^2 \nabla u^2 \cdot \mathbf{n}^2 \, d\Gamma \\
&= \int_{\Gamma_*} (w^1 + \gamma^1 \llbracket w \rrbracket) \bar{j} \, d\Gamma + \int_{\Gamma_*} \llbracket w \rrbracket \langle \kappa \nabla u \rangle_\gamma \cdot \mathbf{n}^2 \, d\Gamma \\
&= \int_{\Gamma_*} \bar{j} \langle w \rangle_{1-\gamma} \, d\Gamma + \int_{\Gamma_*} \llbracket w \rrbracket \langle \kappa \nabla u \rangle \cdot \mathbf{n}^2 \, d\Gamma.
\end{aligned} \tag{4.9}$$

Substituting this last expression (4.9) in (4.7) leads to

$$\begin{aligned}
&\sum_m \int_{\Omega^m} \nabla w^m \kappa^m \nabla u^m \, d\Omega - \frac{\beta}{\epsilon + \beta} \int_{\Gamma_*} \llbracket w \rrbracket \langle \kappa \nabla u \rangle_\gamma \cdot \mathbf{n}^2 \, d\Gamma + \frac{1}{\epsilon + \beta} \int_{\Gamma_*} \llbracket w \rrbracket \llbracket u \rrbracket \, d\Gamma \\
&= \sum_m \int_{\Omega^m} w^m f^m \, d\Omega + \frac{1}{\epsilon + \beta} \int_{\Gamma_*} \llbracket w \rrbracket \bar{i} \, d\Gamma + \int_{\Gamma_*} \bar{j} \langle w \rangle_{1-\gamma} \, d\Gamma,
\end{aligned} \tag{4.10}$$

where, the shorthand notation  $\langle \cdot \rangle_\gamma = \gamma^2(\cdot)^2 + \gamma^1(\cdot)^1$ , refers to the weighted average of a quantity  $(\cdot)$  across the interface. Finally, on symmetrizing, the variational form can be stated as: Find  $u^h \in \mathbb{U}$  such that for all  $v^h \in \mathbb{W}$  such that:

$$\begin{aligned}
&\sum_m \int_{\Omega^m} \nabla w^m \kappa^m \nabla u^m \, d\Omega - \frac{\beta}{\epsilon + \beta} \int_{\Gamma_*} \llbracket w \rrbracket \langle \kappa \nabla u \rangle_\gamma \cdot \mathbf{n}^2 \, d\Gamma - \frac{\beta}{\epsilon + \beta} \int_{\Gamma_*} \llbracket u \rrbracket \langle \kappa \nabla w \rangle_\gamma \cdot \mathbf{n}^2 \, d\Gamma \\
&+ \frac{1}{\epsilon + \beta} \int_{\Gamma_*} \llbracket w \rrbracket \llbracket u \rrbracket \, d\Gamma - \frac{\epsilon\beta}{\epsilon + \beta} \int_{\Gamma_*} \langle \kappa \nabla u \rangle_\gamma \cdot \mathbf{n}^2 \langle \kappa \nabla w \rangle_\gamma \cdot \mathbf{n}^2 \, d\Gamma = \sum_m \int_{\Omega^m} w^m f^m \, d\Omega \\
&- \frac{\beta}{\epsilon + \beta} \int_{\Gamma_*} \bar{i} \langle \kappa \nabla w \rangle_\gamma \cdot \mathbf{n}^2 \, d\Gamma + \frac{1}{\epsilon + \beta} \int_{\Gamma_*} \llbracket w \rrbracket \bar{i} \, d\Gamma + \int_{\Gamma_*} \bar{j} \langle w \rangle_{1-\gamma} \, d\Gamma.
\end{aligned} \tag{4.11}$$

From (4.11), we can see that the unscaled equations, with  $\beta = 0$ , return the standard penalty like techniques for enforcing stiff Neumann and Dirichlet/jump constraints which are no longer well defined for  $\epsilon = 0$ . On the other hand, the

variational form for the rescaled equations is well-defined even for  $\epsilon = 0$ , such that:

$$\begin{aligned}
& \sum_m \int_{\Omega^m} \nabla w^m \kappa^m \nabla u^m \, d\Omega - \int_{\Gamma_*} \llbracket w \rrbracket \langle \kappa \nabla u \rangle_\gamma \cdot \mathbf{n}^2 \, d\Gamma - \int_{\Gamma_*} \llbracket u \rrbracket \langle \kappa \nabla w \rangle_\gamma \cdot \mathbf{n}^2 \, d\Gamma \\
& + \alpha \int_{\Gamma_*} \llbracket w \rrbracket \llbracket u \rrbracket \, d\Gamma = \sum_m \int_{\Omega^m} w^m f^m \, d\Omega - \int_{\Gamma_*} \bar{i} \langle \kappa \nabla w \rangle_\gamma \cdot \mathbf{n}^2 \, d\Gamma + \alpha \int_{\Gamma_*} \llbracket w \rrbracket \bar{i} \, d\Gamma \quad (4.12) \\
& + \int_{\Gamma_*} \bar{j} \langle w \rangle_{1-\gamma} \, d\Gamma,
\end{aligned}$$

where, we have redefined  $\alpha = 1/\beta$  as the stabilization parameter. In fact, if we now choose  $\gamma^1 = \gamma^2 = 0.5$ , we recover Nitsche's variational form as presented in Dolbow and Harari (2009) for enforcing jump constraints across an interface.

In the remainder of this Chapter, we concentrate on this special case of  $\epsilon = 0$  alone. At this point, we also remark on the similarity of the above variational form with that of a symmetric interior penalty discontinuous Galerkin approach. The only difference then lies in evaluating the boundary integrals which would arise at every internal boundary in a symmetric interior penalty approach as opposed to the case described here where they result only over the embedded surface.

The coercivity of Nitsche's variational form and consequently the performance of Nitsche's method depends critically on the stabilization parameter. Interestingly, in a discrete setting, estimates on the stabilization parameter are themselves dictated by the choice of weights,  $\gamma^1$  and  $\gamma^2$  for the weighted average operator in the variational form. We use numerical analysis to provide an optimal estimate for the stabilization parameter and also prescribe the best choice of weights for a robust numerical method in the next section.

### 4.3 Spatial discretization

We discretize the bulk domains  $\Omega^1$  and  $\Omega^2$  into a set of non-overlapping simplices. The interface  $\Gamma_*$  is allowed to be embedded in the domain in the sense that it is

allowed to cut through the elements. We construct a piecewise planar approximation to the interface and locate it through a zero iso-surface for a signed distance level set function in the domain. The interface is thus naturally discretized as the vertex set of intersection points between the zero iso-surface and the element edges. Any element with all its nodes having a positive level set value is considered to belong to  $\Omega^2$  while any element with all its nodes having a negative level set value is considered to belong to  $\Omega^1$ . For elements that are cut by the embedded surface, we create a duplicate set of nodes as proposed in Hansbo and Hansbo (2002). Essentially, we then have a discretized geometry that is defined separately on both the domains with an overlapping character in the vicinity of the embedded surface. To lend further clarity to the proposed formulation, we represent the overlapping element formulation for a cut tetrahedron with a triangular surface element in Figure 4.2. A similar element can of course be constructed when the embedded surface results in a quadrilateral surface element.

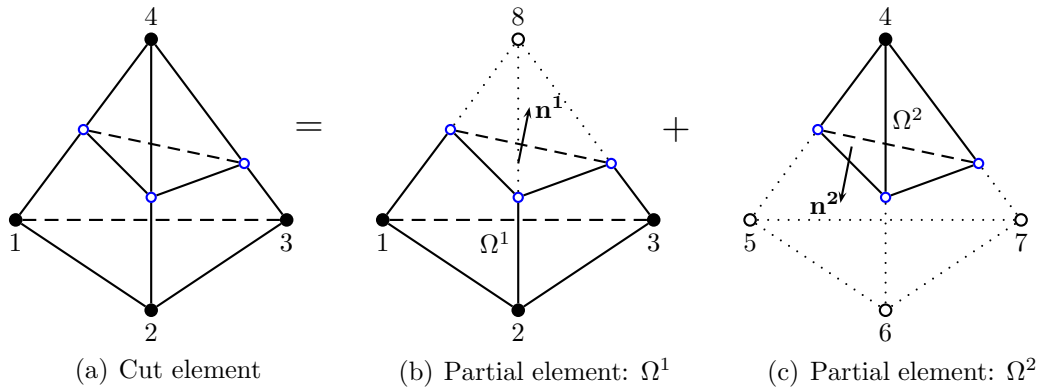


FIGURE 4.2 – Overlapping element formulation for a tetrahedral element cut by an embedded interface. The black circles are the physical nodes corresponding to the background mesh and the hollow circles are the ghost nodes. The blue circles represent the discretization of the embedded interface within an element. The discrete interface is constructed to be piecewise-planar throughout the mesh, and is based on a zero level-set. The discrete interfacial normal  $\mathbf{n}$  in each element is then given by that of the discrete interfacial plane.

The finite element discretization is now constructed on this overlapping domain again as prescribed in Hansbo and Hansbo (2002). The approximation and the

weighting functions are then given by:

$$u^h = \sum_m \sum_{i \in I^m} H^m N_i u_i, \quad w^h = \sum_m \sum_{i \in I^m} H^m N_i w_i, \quad (4.13)$$

where,  $I^m$  is the set of all nodes whose supports overlap the domain  $\Omega^m$  and  $H^m$  is the characteristic function given by

$$H^m(\mathbf{x}) = \begin{cases} 1 & \text{if } \mathbf{x} \in \Omega^m, \\ 0 & \text{otherwise.} \end{cases} \quad (4.14)$$

#### 4.3.1 Discrete equations

On introducing the discretization for the approximation and the weighting spaces as specified above into the variational form (4.12), it is straightforward to obtain the following linear algebraic system:

$$\begin{bmatrix} \mathbf{K}_b^1 + \mathbf{K}_n^1 + \mathbf{K}_s^1 & \mathbf{K}_c \\ \mathbf{K}_c^T & \mathbf{K}_b^2 + \mathbf{K}_n^2 + \mathbf{K}_s^2 \end{bmatrix} \begin{bmatrix} \mathbf{u}^1 \\ \mathbf{u}^2 \end{bmatrix} = \begin{bmatrix} \mathbf{f}_b^1 + \mathbf{f}_n^1 + \mathbf{f}_s^1 \\ \mathbf{f}_b^2 + \mathbf{f}_n^2 + \mathbf{f}_s^2 \end{bmatrix} \quad (4.15)$$

Comparing with the standard finite element discrete form, the equations above consist of the Nitsche contributions  $\mathbf{K}_n^m$ , contribution from the stabilization terms  $\mathbf{K}_s^m$  and the coupling terms  $\mathbf{K}_c$ . The bulk stiffness terms  $\mathbf{K}_b^m$  remain identical to the classical finite element stiffness expressions and are given by assembling the element level contributions, which are given as:

$$\mathbf{k}_b^m = \int_{\Omega_e^m} \mathbf{B}^T \kappa^m \mathbf{B} \, d\Omega_e \quad \text{for } m = 1, 2, \quad (4.16)$$

where the matrix  $\mathbf{B}$  contains the gradient of the shape functions. The contributions arising from the stabilization terms,  $\mathbf{K}_s^m$  is given by assembling:

$$\mathbf{k}_s^m = \alpha_e \int_{\Gamma_*^e} \mathbf{N}^T \mathbf{N} \, d\Gamma_e \quad \text{for } m = 1, 2. \quad (4.17)$$

The Nitsche contributions to the stiffness matrix are given by assembling:

$$\mathbf{k}_n^m = -\gamma_e^m \int_{\Gamma_*^e} \mathbf{N}^T \kappa^m (\mathbf{n}^m)^T \mathbf{B} \, d\Gamma_e - \gamma_e^m \int_{\Gamma_*^e} \mathbf{B}^T \mathbf{n}^m \kappa^m \mathbf{N} \, d\Gamma_e \quad \text{for } m = 1, 2. \quad (4.18)$$

Finally, the coupling terms arising in the discrete system are assembled from the following element level contributions:

$$\mathbf{k}_c = -\alpha_e \int_{\Gamma_*^e} \mathbf{N}^T \mathbf{N} \, d\Gamma_e - \gamma_e^2 \int_{\Gamma_*^e} \mathbf{N}^T \kappa^2 (\mathbf{n}^2)^T \mathbf{B} \, d\Gamma_e - \gamma_e^1 \int_{\Gamma_*^e} \mathbf{B}^T \mathbf{n}^1 \kappa^1 \mathbf{N} \, d\Gamma_e. \quad (4.19)$$

The stabilization parameter  $\alpha_e$  and the weighting parameters  $(\gamma_e^1, \gamma_e^2)$  are defined at an element level in Section 4.3.2.

The discrete forcing vector also has terms arising from bulk and interfacial quantities. The bulk forcing vector is given by assembling:

$$\mathbf{f}_b^m = \int_{\Omega_e^m} \mathbf{N}^T \mathbf{f}^m \, d\Omega_e \quad \text{for } m = 1, 2. \quad (4.20)$$

The Nitsche and the stabilization contributions to the forcing vector are obtained by assembling:

$$\begin{aligned} \mathbf{f}_n^1 &= \gamma_e^1 \int_{\Gamma_*^e} \bar{\mathbf{i}} \kappa^1 \mathbf{B}^T \mathbf{n}^1 \, d\Gamma_e + \gamma_e^2 \int_{\Gamma_*^e} \bar{\mathbf{j}} \mathbf{N}^T \, d\Gamma_e, & \mathbf{f}_s^1 &= - \int_{\Gamma_*^e} \alpha_e \bar{\mathbf{i}} \mathbf{N}^T \, d\Gamma_e, \\ \mathbf{f}_n^2 &= -\gamma_e^2 \int_{\Gamma_*^e} \bar{\mathbf{i}} \kappa^2 \mathbf{B}^T \mathbf{n}^2 \, d\Gamma_e + \gamma_e^1 \int_{\Gamma_*^e} \bar{\mathbf{j}} \mathbf{N}^T \, d\Gamma_e, & \mathbf{f}_s^2 &= \int_{\Gamma_*^e} \alpha_e \bar{\mathbf{i}} \mathbf{N}^T \, d\Gamma_e. \end{aligned} \quad (4.21)$$

#### 4.3.2 Estimate for the stabilization parameter

We follow Dolbow and Harari (2009) and provide a lower bound on the stabilization parameter that ensures a coercive bilinear form. To that end, we introduce the definitions of “energy” norm of a quantity over a domain and  $L_2$  norm of a quantity over a surface

$$|u^h|_{\Omega, \kappa}^2 = \sum_m \int_{\Omega^m} \nabla u^m \cdot \kappa^m \nabla u^m \, d\Omega, \quad |u^h|_{\Gamma_*}^2 = \int_{\Gamma_*} u^h u^h \, d\Gamma. \quad (4.22)$$

In the discrete setting, the variational form (4.12) can be written as:

$$a(w^h, u^h) = l(w^h), \quad (4.23)$$

where,  $a(w^h, u^h)$  and  $l(w^h)$  are given by:

$$\begin{aligned} a(w^h, u^h) &= \sum_m \int_{\Omega^m} \nabla w^{mh} \kappa^m \nabla u^{mh} \, d\Omega - \int_{\Gamma_*} \llbracket w^h \rrbracket \langle \kappa \nabla u^h \rangle_\gamma \cdot \mathbf{n}^2 \, d\Gamma \\ &\quad - \int_{\Gamma_*} \llbracket u^h \rrbracket \langle \kappa \nabla w^h \rangle_\gamma \cdot \mathbf{n}^2 \, d\Gamma + \alpha \int_{\Gamma_*} \llbracket w^h \rrbracket \llbracket u^h \rrbracket \, d\Gamma, \end{aligned} \quad (4.24)$$

$$\begin{aligned} l(w^h) &= \sum_m \int_{\Omega^m} w^{mh} f^m \, d\Omega - \int_{\Gamma_*} \bar{i} \langle \kappa \nabla w^h \rangle_\gamma \cdot \mathbf{n}^2 \, d\Gamma \\ &\quad + \alpha \int_{\Gamma_*} \llbracket w^h \rrbracket \bar{i} \, d\Gamma + \int_{\Gamma_*} \bar{j} \langle w^h \rangle_{1-\gamma} \, d\Gamma. \end{aligned} \quad (4.25)$$

From the definitions of the norms (4.22) and the bilinear form (4.24), it is easy to see that the discrete energy is given by

$$\begin{aligned} a(u^h, u^h) &= |u^h|_{\Omega, \kappa}^2 + \alpha |\llbracket u^h \rrbracket|_{\Gamma_*}^2 - 2 \int_{\Gamma_*} \llbracket u^h \rrbracket \langle \kappa \nabla u^h \rangle_\gamma \cdot \mathbf{n}^2 \, d\Gamma, \\ &\geq |u^h|_{\Omega, \kappa}^2 + \alpha |\llbracket u^h \rrbracket|_{\Gamma_*}^2 - 2 |\llbracket u^h \rrbracket|_{\Gamma_*} |\langle \kappa \nabla u^h \rangle_\gamma \cdot \mathbf{n}^2|_{\Gamma_*}, \\ &\geq (|u^h|_{\Omega, \kappa} - C_1 |\llbracket u^h \rrbracket|_{\Gamma_*})^2 + (\alpha - C_1^2) |\llbracket u^h \rrbracket|_{\Gamma_*}^2. \end{aligned} \quad (4.26)$$

The second line follows from Cauchy-Schwarz inequality. The last line follows from the generalized inverse estimate presented in Brezzi and Fortin (1991), which states the existence of a mesh-dependent configuration constant such that:

$$|\langle \kappa \nabla u^h \rangle_\gamma \cdot \mathbf{n}^2|_{\Gamma_*} \leq C_1 |u^h|_{\Omega, \kappa}. \quad (4.27)$$

We now use the generalized inverse estimate (4.27) to provide a lower bound on  $C_1^2$ . First, we insist on the coercivity elementwise which places a stronger constraint on the stabilization parameter than that necessitated by global coercivity

requirements. However, these element level calculations facilitate analytical estimates for constant strain triangular and tetrahedral elements resulting in a more efficient method.

For a constant strain triangle or a tetrahedron, on assuming that the material constant  $\kappa^m$  remains constant within an element we obtain:

$$|u^h|_{\Omega, \kappa}^2 = \sum_m \text{meas}(\Omega_e^m) \kappa^m |\nabla u^{mh}|^2. \quad (4.28)$$

Similarly for average flux at the interface we have:

$$\begin{aligned} |\langle \kappa \nabla u^h \rangle_\gamma \cdot \mathbf{n}^2|_{\Gamma_{*e}}^2 &= \text{meas}(\Gamma_{*e}) (\gamma_e^1 \kappa^1 |\nabla u^{1h} \cdot \mathbf{n}^2 + \gamma_e^2 \kappa^2 |\nabla u^{2h} \cdot \mathbf{n}^2|)^2, \\ &\leq \text{meas}(\Gamma_{*e}) (\gamma_e^1 \kappa^1 |\nabla u^{1h}| + \gamma_e^2 \kappa^2 |\nabla u^{2h}|)^2, \\ &\leq \text{meas}(\Gamma_{*e}) ((\gamma_e^1 \kappa^1 |\nabla u^{1h}|)^2 (1 + \delta) + (\gamma_e^2 \kappa^2 |\nabla u^{2h}|)^2 (1 + 1/\delta)). \end{aligned} \quad (4.29)$$

where the first line follows from the definition of average flux at the interface, second from the definition of a unit normal. The third line follows on using Young's inequality for any  $\delta > 0$ . Now, if we choose:

$$\begin{aligned} \delta &= \frac{\text{meas}(\Omega_e^1) \kappa^2 (\gamma_e^2)^2}{\text{meas}(\Omega_e^2) \kappa^1 (\gamma_e^1)^2}, \\ |\langle \kappa \nabla u^h \rangle_\gamma \cdot \mathbf{n}^2|_{\Gamma_{*e}}^2 &\leq \\ \text{meas}(\Gamma_{*e}) \sum_m \text{meas}(\Omega_e^m) \kappa^m |\nabla u^{mh}|^2 &\left( \frac{\kappa^2 (\gamma_e^2)^2}{\text{meas}(\Omega_e^2)} + \frac{\kappa^1 (\gamma_e^1)^2}{\text{meas}(\Omega_e^1)} \right). \end{aligned} \quad (4.30)$$

Then from (4.30), (4.28) and (4.27) the mesh dependent parameter  $C_I$  obeys the following relation:

$$C_I^2 \geq \text{meas}(\Gamma_{*e}) \left( \frac{\kappa^2 (\gamma_e^2)^2}{\text{meas}(\Omega_e^2)} + \frac{\kappa^1 (\gamma_e^1)^2}{\text{meas}(\Omega_e^1)} \right). \quad (4.31)$$

Now from (4.26), choosing  $\alpha_e \geq C_I^2$  makes the bilinear form coercive and consequently provides stability to the method. Based on the parametric studies conducted

in Embar et al. (2010), we choose  $\alpha_e = 2C_1^2$ , where  $C_1^2$  is obtained on considering the equality sign in (4.31).

From (4.31), we can see that the stabilization parameter depends directly on the weighting parameters  $\gamma_e^1$  and  $\gamma_e^2$ . From a consistency perspective, we are only constrained in our choice of weights such that they sum to unity. However, from a stability perspective we notice that an arbitrary choice would necessitate a large value for the stabilization parameter providing an unstable character to the method. For instance, if the weights are identically chosen as  $\gamma_e^1 = \gamma_e^2 = 0.5$  as is done in classical Nitsche's method, we notice that (a) as  $\text{meas}(\Omega_e^m) \rightarrow 0$  (b) as  $\kappa^m$  tends to a large value or (c) a combination of the above would result in an unusually large estimate for the element level stabilization parameter.

With classical Nitsche's method, this has often resulted in numerical issues which have surfaced in different contexts and have been reported in Laursen et al. (2010) as stress locking for mesh mortaring, in Chapter 3 as incompatibility with explicit dynamics, in Dolbow and Harari (2009), Mourad et al. (2007) and Sanders et al. (2009) through an uncharacteristic sub-optimal convergence in the interfacial field. Here, we propose a smarter choice for the weights to alleviate these numerical issues and arrive at a more robust form of Nitsche's method.

On using the condition  $\gamma_e^1 + \gamma_e^2 = 1$ , we can arrive at a functional relationship between the stabilization parameter and the interfacial weighting such that:

$$\alpha = 2 \times \text{meas}(\Gamma_{*e}) \left( \frac{\kappa^1(\gamma_e^1)^2}{\text{meas}(\Omega_e^1)} + \frac{\kappa^2(1 - \gamma_e^1)^2}{\text{meas}(\Omega_e^2)} \right). \quad (4.32)$$

Now, we can choose an interfacial weight that minimizes the stabilization parameter while ensuring coercivity. For an extrema, considering the stationary point of  $\alpha_e$  with respect to  $\gamma_e^1$ , we get:

$$\gamma_e^1 = \frac{\text{meas}(\Omega_e^1)/\kappa^1}{\text{meas}(\Omega_e^1)/\kappa^1 + \text{meas}(\Omega_e^2)/\kappa^2}, \quad (4.33)$$



It is also instructive to look at the second derivative of  $\alpha$  with respect to  $\gamma_e^1$ :

$$\frac{\partial^2 \alpha}{\partial (\gamma_e^1)^2} = 4 \times \text{meas}(\Gamma_{*e}) \left( \frac{\kappa^1}{\text{meas}(\Omega_e^1)} + \frac{\kappa^2}{\text{meas}(\Omega_e^2)} \right) \geq 0. \quad (4.34)$$

Therefore, with this particular choice of interfacial weighting given by equation (4.33), we indeed end up minimizing the stabilization parameter while ensuring coercivity of the bilinear form. On substituting this choice of weights back into (4.32), we get:

$$\alpha_e = 2 \times \frac{\text{meas}(\Gamma_{*e})}{\text{meas}(\Omega_e^1)/\kappa^1 + \text{meas}(\Omega_e^2)/\kappa^2} \quad (4.35)$$

Clearly, now even for the pathological cases encountered with classical Nitsche's method that we mentioned above, this choice of weights provides us with a well-defined estimate for the stabilization parameter that remains of the same order of magnitude as the classical finite element stiffness terms. We present a detailed numerical sensitivity study in the next section which further emphasizes this choice.

As an additional remark, we also note that though the above analysis was conducted particularly for cut elements, it generalizes without modifications for the case when an interface aligns with the background mesh surfaces as is the case with symmetric interior penalty discontinuous Galerkin methods. Therefore, even for those methods, while a simple arithmetic average would work well for homogenous materials and a purely stiffness weighted approach for only material heterogeneities, the proposed weighting would prove more beneficial in handling both large material and mesh anisotropies.

Finally, we conclude by remarking that for bilinear quadrilaterals and other higher order elements, since the flux within an element no longer remains a constant, we cannot provide analytical estimates for the stabilization parameter. However, a qualitative dependence between the weights and the stabilization parameter can still be

established in a similar way. For such elements an additional element level eigenvalue calculation could be performed to evaluate the stabilization parameter, as illustrated in Embar et al. (2010).

## 4.4 Numerical examples

In this section, we revisit several numerical examples studied previously in (Dolbow and Harari, 2009; Sanders et al., 2009; Laursen et al., 2010; Hautefeuille et al., 2012; Sanders et al., 2012; Dolbow and Franca, 2008), where classical Nitsche’s method was shown to perform poorly for certain pathological cases. We highlight the robustness of the proposed formulation over those very examples by contrasting its performance with the more conventional form prevalent in literature.

Throughout the section and in the figure legends, we denote the weighted form of Nitsche’s method by  $\gamma$  Nitsche and the standard form by Nitsche. The accuracy in bulk field is evaluated by means of the standard  $L_2$  error norm while the accuracy in the gradient is evaluated by means of the energy semi-norm. For the interfacial flux quantities, we also utilize  $L_\infty$  norm as a measure of accuracy.

### 4.4.1 Sensitivity study

As a first example, we examine the following model problem as considered by Hansbo and Hansbo (2002):

$$\begin{aligned}
 \nabla \cdot \kappa^m \nabla u^m &= -f^m && \text{in } \Omega^m, \\
 u &= 0 && \text{on } \Gamma_d = \{\mathbf{x} : z = 0 | 1\}, \\
 \kappa^m \nabla u^m \cdot \mathbf{n} &= 0 && \text{on } \Gamma_n = \{\mathbf{x} : x = 0 | 1; y = 0 | 1\}, \\
 \llbracket \kappa \nabla u \rrbracket \cdot \mathbf{n} &= 0 && \text{on } \Gamma_* = \{\mathbf{x} : \psi(\mathbf{x}) = 0\}, \\
 \llbracket u \rrbracket &= u^2(\mathbf{x}) - u^1(\mathbf{x}) && \text{on } \Gamma_* = \{\mathbf{x} : \psi(\mathbf{x}) = 0\}.
 \end{aligned} \tag{4.36}$$

Choosing  $f^1 = f^2 = 1$ , the equation admits an analytical solution given by:

$$u(\mathbf{x}) = \begin{cases} \frac{(3\kappa^1 + \kappa^2)z}{4\kappa^1(\kappa^1 + \kappa^2)} - \frac{z^2}{2\kappa^1} & \text{in } \Omega^1 = \{\mathbf{x} : \psi(\mathbf{x}) < 0\}, \\ \frac{\kappa^2 - \kappa^1 + (3\kappa^1 + \kappa^2)z}{4\kappa^2(\kappa^1 + \kappa^2)} - \frac{z^2}{2\kappa^2} & \text{in } \Omega^2 = \{\mathbf{x} : \psi(\mathbf{x}) > 0\}. \end{cases}$$

We now study the sensitivity of the interfacial flux with respect to: (a) the interface location within a cut element, and (b) contrast in material properties across the interface; over a structured three dimensional tetrahedral mesh with six divisions in each direction. In order to carefully examine the sensitivity with respect to the interface location, we move a planar interface within one layer of elements from one internal boundary to another as shown in Figure 4.3(a). For each of these interfacial locations, we vary the material parameters such that the ratio between  $\kappa^2$  and  $\kappa^1$  ranges from  $10^{-6}$  to  $10^6$ .

For an application such as modeling plasticity at the interface, it is important that the equivalent expression for flux obtained from Nitsche's method *i.e.*  $\alpha[[u]] - \langle \kappa \nabla u \rangle_\gamma \cdot \mathbf{n}$  is evaluated accurately at each gauss point on the interface. With such an application in mind, we plot the maximum error in flux at gauss points on the interface which is akin to looking at error in the  $L_\infty$  norm.

The error contour plots for weighted and classical Nitsche's method are shown in Figures 4.3(b)-(c). In the plot, the horizontal axis represents the normalized variation in interface location *i.e.*  $d_z/h$  while the vertical axis represents the log value of variation in contrast in material properties *viz.*  $\log(\kappa^2/\kappa^1)$ . It is clear from the plots that classical Nitsche's method is much more sensitive to both the degree of heterogeneity as well as the position of interface within a cut element. While the variation in error for the weighted form of Nitsche's method is relatively modest with error values ranging from 20-40% for almost the entire spectrum, classical Nitsche's method exhibits severe sensitivity with error values shooting up to 18000% for the worst

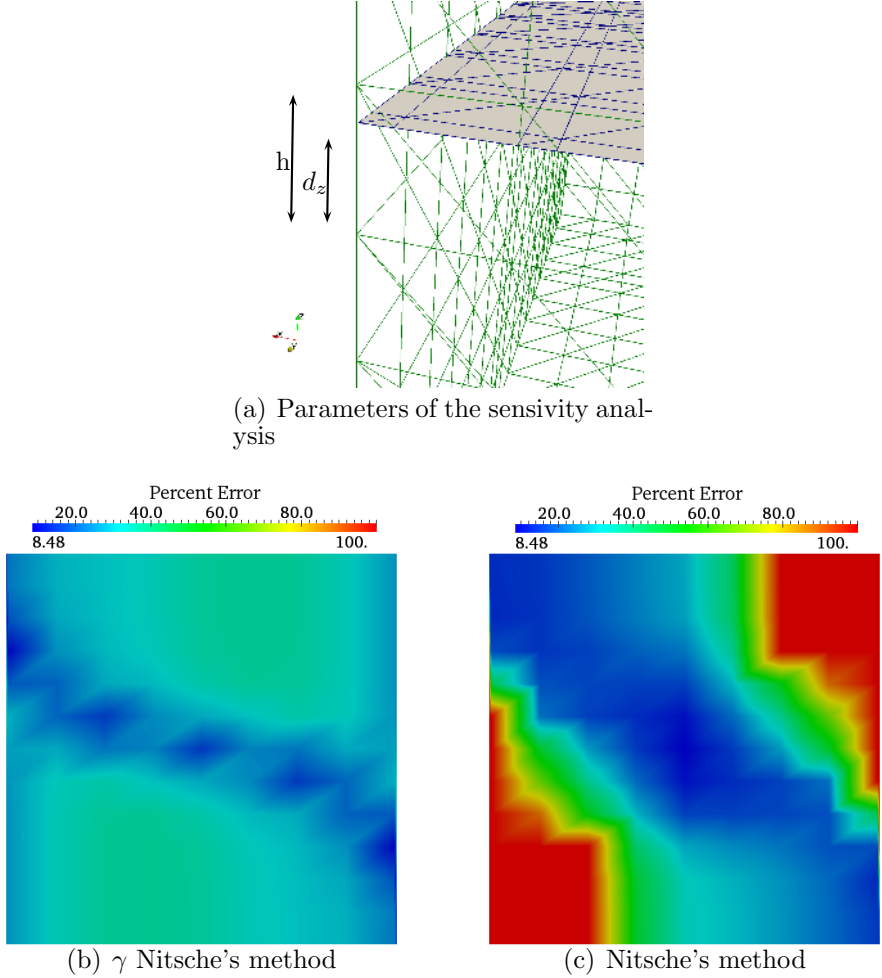


FIGURE 4.3 – Contour plot of the percentage error in  $L_\infty$  norm for equivalent flux at the interface. The variation in the interfacial location  $d_z/h$  is along the horizontal axis while the variation in contrast in material properties  $\log(\kappa^2/\kappa^1)$  is along the vertical axis. Percentage error is indicated by the color contour.

possible cases. The contour plot is scaled between 8-100 % for better visualization.

It is also interesting to note that this sensitivity plot for classical Nitsche’s method is not entirely symmetric. It seems that classical Nitsche’s method is worst affected when  $d_z/h \rightarrow 0$  and  $\kappa^2/\kappa^1 = 10^{-6}$  or its corresponding situation when  $d_z/h \rightarrow 1$  and  $\kappa^2/\kappa^1 = 10^6$ . To get a better understanding, we study the sensitivity with respect to one parameter at a time while holding the other fixed. We first plot the maximum error by varying only the interface location for a high contrast in material properties

in Figure 4.4(a) and for identical material properties in Figure 4.4(b). We then vary the contrast in material properties while holding the interface location fixed such that in Figure 4.5(a) the interface is approaching a lower internal boundary and in Figure 4.5(b) the interface is approaching an upper internal boundary.

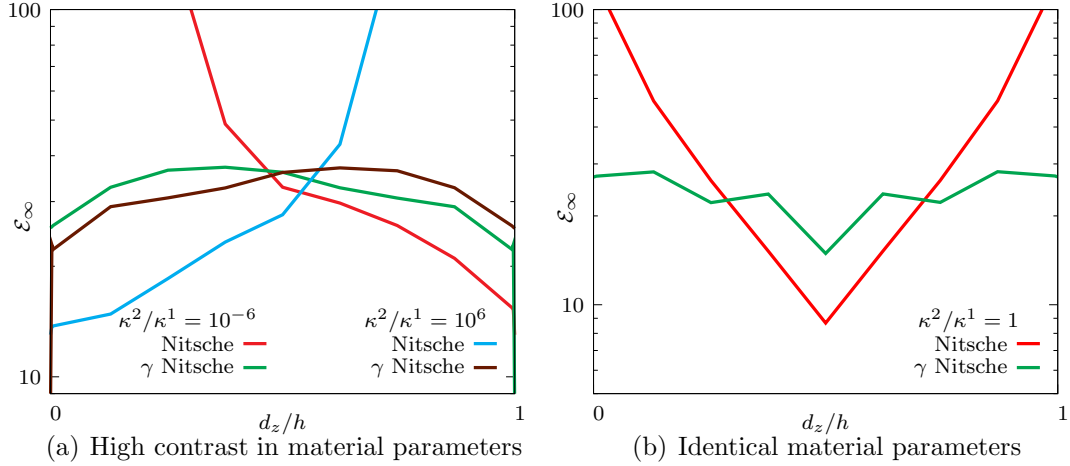


FIGURE 4.4 – Sensitivity in interfacial flux error with interfacial location. Percentage error in the interfacial flux in  $L_\infty$  norm is plotted along the vertical axis and the normalized interfacial location is plotted along the horizontal axis for (a) high degree of heterogeneity (b) homogeneous material.

From Figure 4.4(b), it is clear that even for a homogeneous material, as the interface approaches either internal boundary, for classical Nitsche’s method, interfacial flux is erroneous to the point that it is barely usable without some post-treatment as reported earlier by (Mourad et al., 2007; Dolbow and Franca, 2008; Hautefeuille et al., 2012). On the other hand, weighted Nitsche’s method remains well behaved. The only difference between classical and weighted approaches for the aforementioned case is the averaging of the flux that arises in the consistency terms in the variational form. This averaging as it turns out is the key to the numerical stability of the method.

Averaging biased towards either side mandates a large value of the stabilization parameter which causes the method to revert back to unstable behaviour as exhibited by penalty function approaches with a large penalty parameter. Essentially, a large

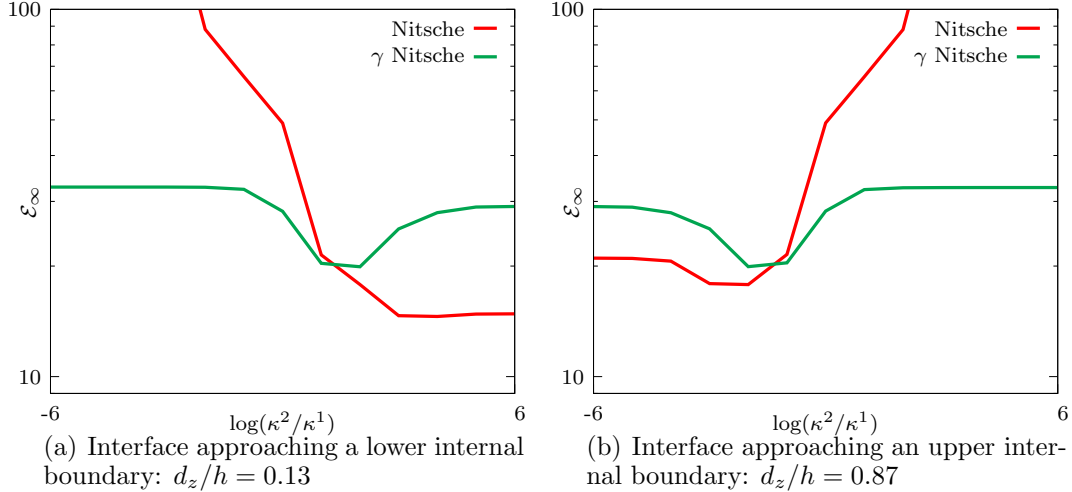


FIGURE 4.5 – Sensitivity in interfacial flux error with degree of heterogeneity. Percentage error in the interfacial flux in  $L_\infty$  norm is plotted along the vertical axis and the log ratio of material parameters is plotted along the horizontal axis for (a) an interface approaching a lower internal boundary (b) an interface approaching an upper internal boundary.

value for the stabilization parameter nullifies the advantages of a weakly continuous approach and enforces the constraints a little too strongly. This further explains the asymmetry in the plots in Figures 4.4(a), 4.5(a) and 4.5(b). If we concentrate on the case when  $\kappa^2/\kappa^1 = 10^6$  and  $d_z/h \rightarrow 1$ , for classical Nitsche’s method, we notice a spike in the error. For that case, a simple averaging is already overestimating the flux from  $\Omega^2$  and when the material parameter is large we end up amplifying that affect. On the other hand, when  $d_z/h \rightarrow 0$ , a simple average is underestimating the flux contribution from  $\Omega^2$  and a high value of material parameter counterbalances that to some extent resulting in better performance of classical Nitsche’s method.

This suggests to us that as long as we can ensure an “unbiased” averaging in the flux for the consistency terms arising in the variational form, we retain a well behaved numerical method. Fortunately, the essence of the above qualitative discussion is captured by the expression for stabilization parameter (4.32) which leads us to the choice of the weights for the averaging as specified in (4.33).

#### 4.4.2 Pure bending

We next consider a rectangular beam of length  $L$  and height  $H$  subjected to pure bending conditions as considered by Laursen et al. (2010) and Sanders et al. (2009, 2012) and demonstrate the robustness of the proposed approach.

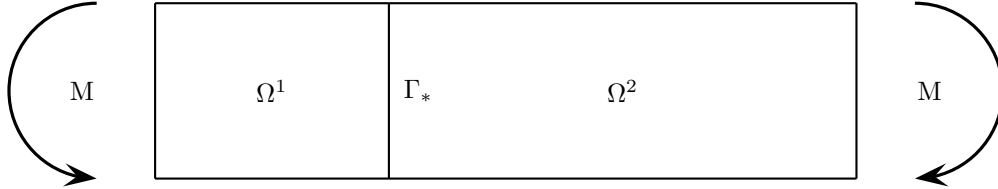


FIGURE 4.6 – Geometry and loading for the beam bending problem.

On considering idealized pure bending with no shear and centering the computational domain around  $(0,0)$ , one can arrive at the analytical expressions for the displacement field given by:

$$u_x^m(\mathbf{x}) = \left\{ \frac{2pxy}{E^m H} \text{ in } \Omega^m \right. , \quad u_y^m(\mathbf{x}) = \left\{ \frac{-p(x^2 - \nu y^2)}{E^m H} \text{ in } \Omega^m \right. . \quad (4.37)$$

where  $p$  is the maximum value of the distributed traction applied at the boundaries to produce the bending moment  $M$ . For the purpose of this numerical experiment, we choose  $p = 1$ . In addition, we consider a straight vertical interface such that  $\psi(\mathbf{x}) = x - x_*$ , partitions the domain  $\Omega$  into component sub-domains  $\Omega^1 = \{\mathbf{x} : \psi(\mathbf{x}) < 0\}$  and  $\Omega^2 = \{\mathbf{x} : \psi(\mathbf{x}) > 0\}$ . Displacements and tractions are assumed to be continuous across the interface.

#### *Stress locking*

We first demonstrate the stress locking phenomenon as observed in Laursen et al. (2010); Sanders et al. (2012). The phenomenon is best highlighted when there is a high contrast in stiffnesses on either side of an embedded interface chosen as  $\psi(\mathbf{x}) =$

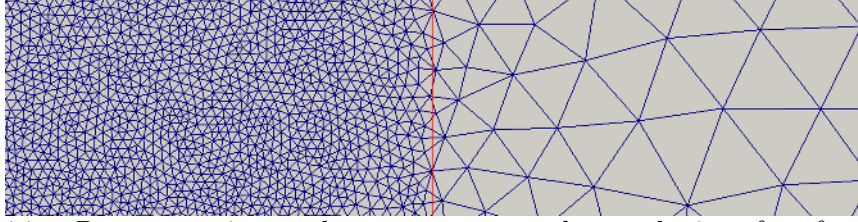


FIGURE 4.7 – Representative mesh geometry zoomed near the interface for the stress locking problem.

$x$ , as well as a corresponding scaling in the mesh densities. Figure 4.7 shows a zoom of a representative mesh in the vicinity of the interface. Here, we consider the material properties such that the Young’s moduli  $E^1 = 10^9$  and  $E^2 = 10^3$  while the Poisson’s ratios are identically considered as  $\nu^1 = \nu^2 = 0$ . The loading is prescribed by constraining the displacement field to the exact solution given by equation (4.37) on the Dirichlet boundary  $\Gamma_d = \{\mathbf{x} : x = -L/2|L/2\}$ . Clearly from Figure 4.8, we can see that classical Nitsche’s method results in wildly oscillating behaviour in the bending stress and renders the results unusable when the bending stresses are desired by an analyst.

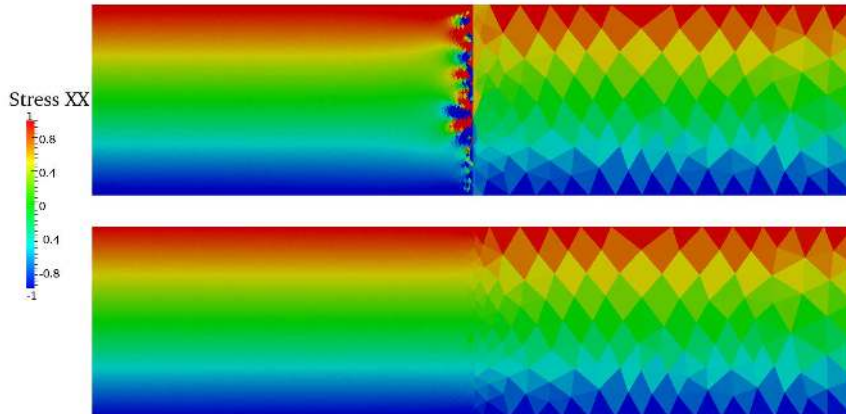


FIGURE 4.8 – Bending stresses top: for classical Nitsche’s method and bottom: for  $\gamma$  Nitsche’s method.

Furthermore, we conduct a spatial convergence study on five sequentially refined unstructured meshes. The mesh on the stiffer side is approximately four times as refined as the mesh on the softer side. Also, the meshes are refined such that this



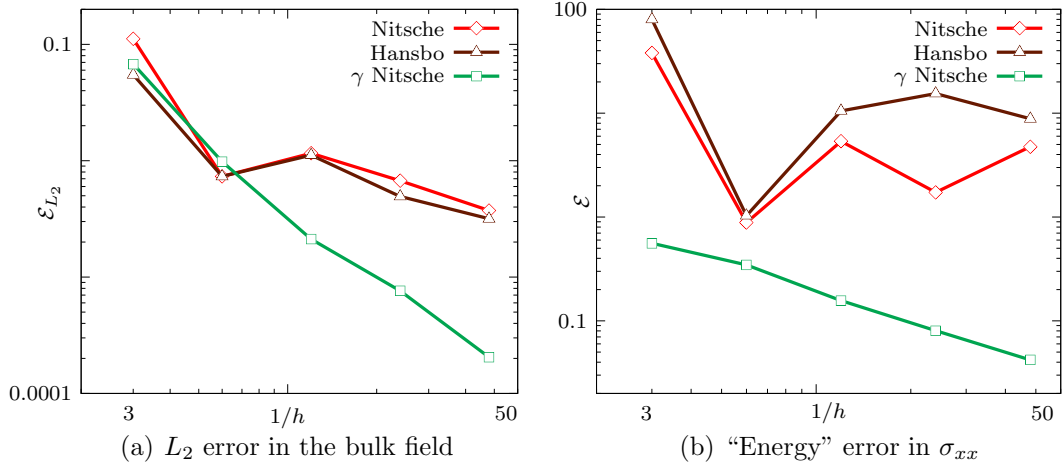


FIGURE 4.9 – Convergence study for the beam bending problem solved on a rectangular beam with a straight vertical interface.

contrast in mesh densities is always maintained the same for each refinement. Figure 4.9(a) plots the convergence of bulk field in  $L_2$  norm while Figure 4.9(b) plots the convergence of the bending stress  $\sigma_{xx}$  in the energy semi-norm. From Figure 4.9(b), it is clear that classical Nitsche’s method fails to converge which corroborates the result seen in Figure 4.8. It is interesting to note that an area-weighted Nitsche’s method as proposed in Chapter 3 also fails to converge. This is not surprising however as equation (4.32) suggests that an area-weighted approach can still lead to a large stabilization parameter for certain configurations of cut elements if there is a high degree of heterogeneity. The weighting proposed here on the other hand is expected to behave better by always keeping the stabilization terms in check as demonstrated by the optimal convergence behaviour of the proposed approach. It is also interesting to note from Figure 4.9(a) that classical as well as the area-weighted approaches yield sub-optimal convergence behaviour in  $L_2$  norm in the bulk field as well.

### *Interfacial traction*

In order to successfully model more complex interfacial behaviour like frictional sliding, it is essential to have an accurate representation of normal as well as tangential traction at the interface. With that in mind, we now examine the interfacial traction for the aforementioned beam bending problem.

We again consider a rectangular beam subjected to pure bending as above but with length  $L = 16$  and height  $H = 4$  and first conduct computations on a structured triangular mesh for a non-pathological case, such that  $d_x/h = 0.5$  for classical Nitsche's method. The definition of  $d_x$  is as shown in the inset of Figure 4.10(a). We consider 21 divisions along both  $x$  and  $y$  directions and consider the interface to be located exactly at the center of the domain, such that  $\psi(\mathbf{x}) = x$ . The material properties are identically prescribed as  $E^1 = E^2 = 10^3$  and  $\nu^1 = \nu^2 = 0$ . The exact solution is still given by equation (4.37) and is again prescribed as a Dirichlet boundary condition on the surfaces  $\Gamma_d = \{\mathbf{x} : x = -L/2 | L/2\}$ . The normal and interfacial traction profiles are plotted in Figures 4.10(a)-(b) respectively for classical Nitsche's method, weighted Nitsche's method as well as for penalty method. The penalty parameter used for penalty method scales as  $E/h$ ,  $h$  being the characteristic length of the mesh.

Similar plots were provided by Sanders et al. (2009), where they noticed that normal traction profile showed very mild oscillations while the tangential traction oscillated wildly for classical Nitsche's method. They rightly attributed this to the poor performance of a constant strain triangular element. However, we contend here that while a constant strain triangular element contributes in part, some of this unstable behaviour can also be attributed to the numerical instability inherent in the classical Nitsche formulation as evident from the relatively smoother profiles produced by the penalty method for the chosen penalty parameter. It is therefore

interesting to note from 4.10(a)- (b), the traction profile obtained from the weighted formulation is much smoother in both the normal and tangential directions. Tangential traction still exhibits oscillatory pattern but the amplitude of these oscillations is much lower than classical Nitsche’s method.

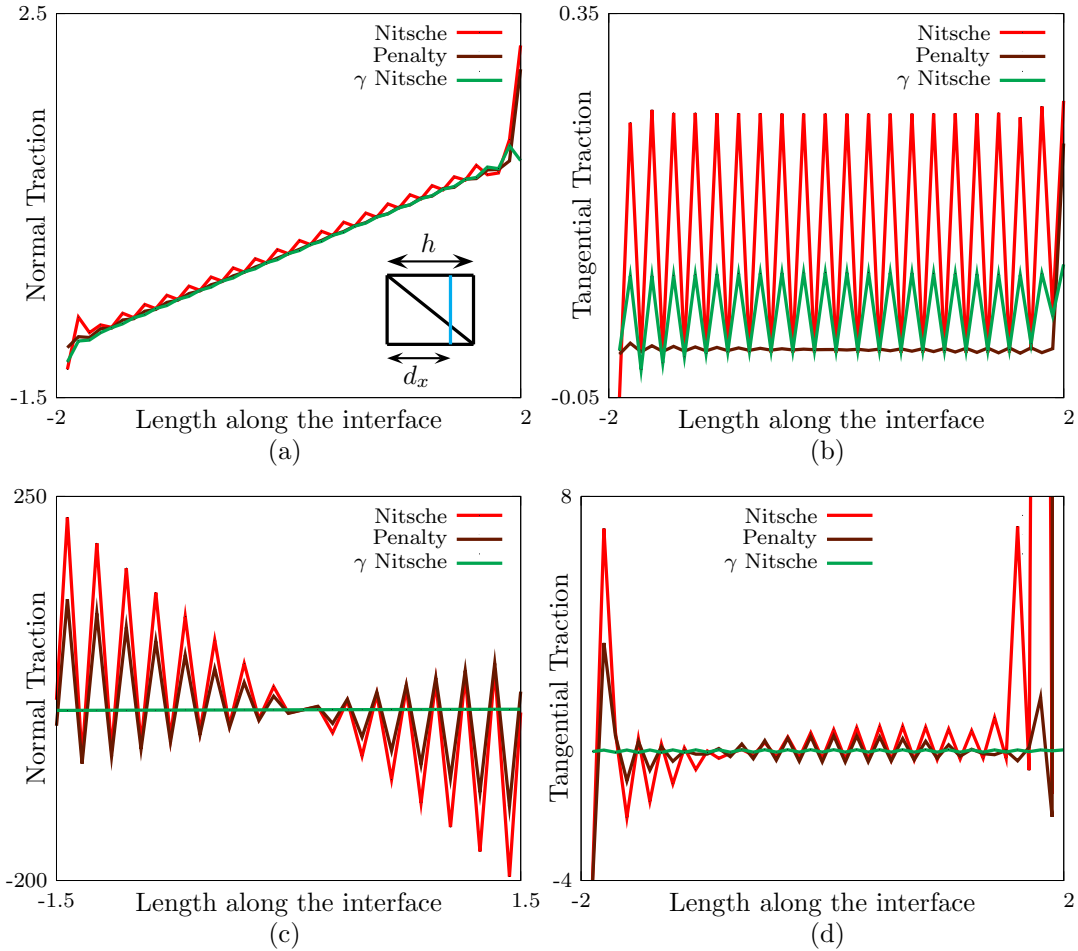


FIGURE 4.10 – Plot of normal and tangential traction at the interface for beam bending problem. Contrast in material properties as well as distance between the closest internal boundary and the interface is varied from identical material properties ( $E^1 = E^2 = 10^3$ ) on both sides and  $d_x = h/2$  for top-left and right to  $E^1 = 10^9$ ,  $E^2 = 10^3$  and  $d_x = .01$  for bottom-left and right. The inset in (a) shows the definition of parameters  $d_x$  and  $h$ .

Since the weighted form exhibits advantages even in a case that is supposedly non-pathological for classical Nitsche’s method, we now perform the same analysis for a pathological case with  $d_x = .01$ . We also consider a highly heterogeneous

case with  $E^1 = 10^9$  and  $E^2 = 10^3$ . Poisson's ratio is identically considered zero in both domains. Clearly from Figures 4.10(c)-(d), one can appreciate that the traction obtained from classical Nitsche's method is as bad as that obtained from an unstable Lagrange multiplier implementation.

Penalty method also seems to revert back to unstable behaviour but we must mention that the penalty parameter used here scales as  $\max(E^1, E^2)/h$ . It is likely that a smaller parameter yields better performance but the problems in a penalty function approach are twofold - (a) we see a loss in optimal convergence behaviour in both bulk and interfacial fields and (b) the accuracy of the results in both the bulk and interfacial fields are highly dependent on the penalty parameter and this choice is completely at an analyst's discretion leading to a lack of robustness.

We also compare the interfacial tractions obtained for the beam bending problem with two other choices of weighting in the Nitsche approach: (a) the area-weighted approach as proposed in Chapter 3, and (b) the stiffness-weighted approach as proposed in references (Zunino, 2009; Burman and Zunino, 2006; Ern et al., 2009). We again consider a highly heterogeneous case with  $E^1 = 10^3$  and  $E^2 = 10^9$ , while the Poisson's ratio is identically zero. Again, it is clear from the plots (see Figures 4.11) that the approach presented in this work leads to a much more stable traction profile than one would obtain from choosing either a simple area-weighted approach or a pure stiffness weighted approach.

It is also worth remarking that one might improve on the results obtained on using only the approach suggested in (Hansbo and Hansbo, 2002) or the one suggested in (Zunino, 2009; Burman and Zunino, 2006; Ern et al., 2009) throughout, by switching between them on an element by element basis as suggested by Zunino *et al.* (Zunino et al., 2011). However, as one can appreciate from the plots shown in Figure 4.11 such a strategy might not be sufficient in the face of both small cut elements and large heterogeneities.

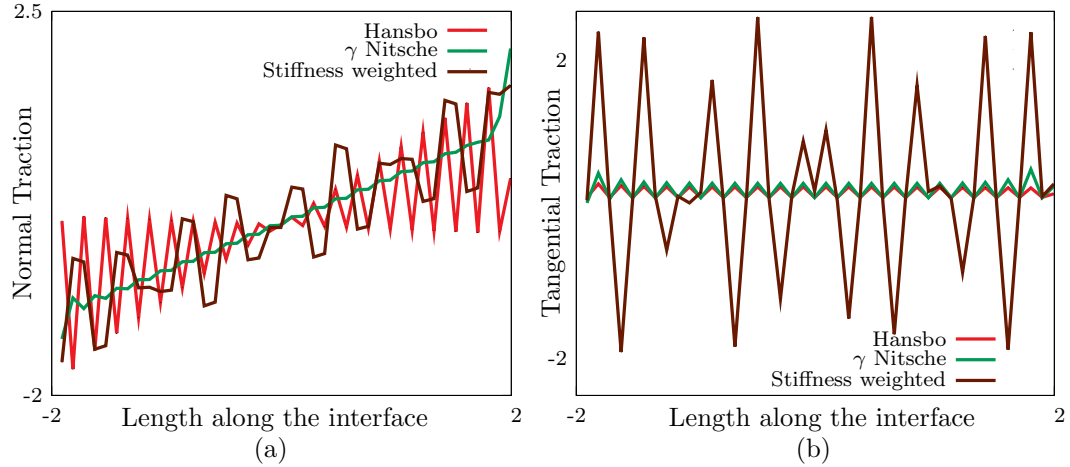


FIGURE 4.11 – Comparison of analytical surface tractions obtained using the proposed approach with those obtained using other choices of weights. The material parameters are  $E^1 = 10^3$ ,  $E^2 = 10^9$  and  $d_x = .05$ .

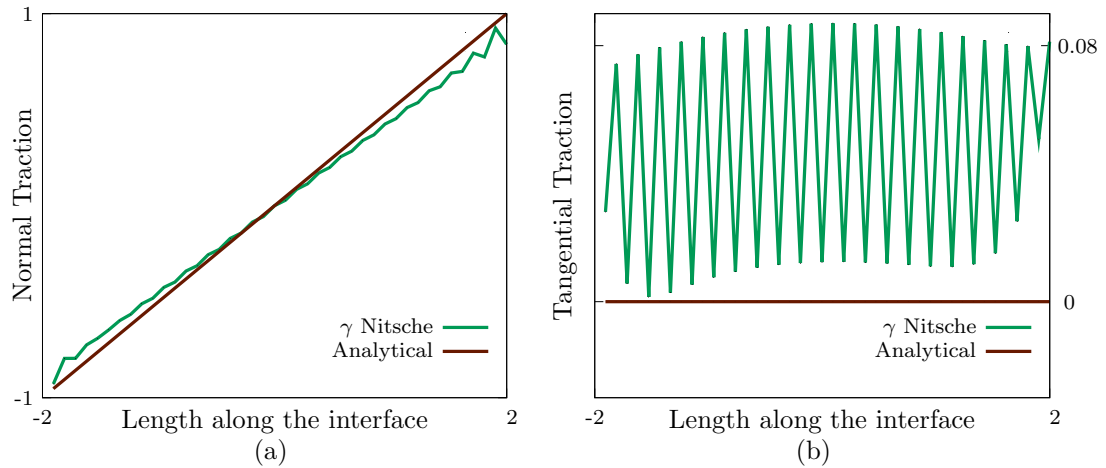


FIGURE 4.12 – Comparison of analytical surface tractions with those obtained using  $\gamma$  Nitsche's method for the pathological case:  $E^1 = 10^9$ ,  $E^2 = 10^3$  and  $d_x = .01$ .

Finally, we also compare the normal and tangential tractions obtained from the weighted formulation with the analytical solutions in Figures 4.12(a)-(b). We can clearly see that the proposed weighted formulation remains well behaved even for a severely pathological case as described here. We again notice some oscillatory behaviour in the tangential field however. At the same time, we also remark that the amplitude of oscillations remains approximately the same as that seen for the case

when  $d_x/h = 0.5$  and identical material properties indicating minimal sensitivity as opposed to classical Nitsche's method.

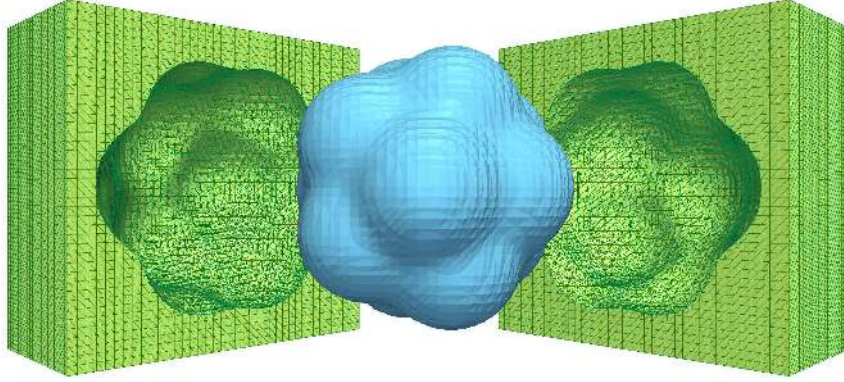


FIGURE 4.13 – Computational domain with  $\Omega^2$  shown in green and  $\Omega^1$  shown in blue for the logarithmic field problem with a popcorn shaped embedded surface.

Though this situation is designed particularly to highlight a worst case scenario for classical Nitsche's method, while modeling heterogeneities with generalized/extended finite element methods or with discontinuous Galerkin methods; one often encounters such situations and it is highly inconvenient to use tolerancing schemes to get rid of these pathological cases. It is therefore encouraging that the weighted formulation remains well behaved and allows for a method without the need for any tolerancing schemes.

#### 4.4.3 Popcorn interface

As a final example, we revisit the logarithmic field problem with a popcorn shaped surface considered in Chapter 2 and reformulate the Dirichlet problem studied there as a jump problem given below.

$$\begin{aligned}
 \Delta u &= -f & \text{in } \Omega &= \Omega^1 \cup \Omega^2, \\
 u &= \log r & \text{on } \Gamma_d &= \{\mathbf{x} : x = 0|1; y = 0|1; z = 0|1\}, \\
 \llbracket \nabla u \rrbracket \cdot \mathbf{n} &= \nabla u^2 \cdot \mathbf{n}^2 & \text{on } \Gamma_* &= \{\mathbf{x} : \psi(\mathbf{x}) = 0\}, \\
 \llbracket u \rrbracket &= u^2(\mathbf{x}) - u^1(\mathbf{x}) & \text{on } \Gamma_* &= \{\mathbf{x} : \psi(\mathbf{x}) = 0\}.
 \end{aligned}$$

Choosing  $f = -1/r^2$  in  $\Omega^2 = \{\mathbf{x} : \psi(\mathbf{x}) > 0\}$  and  $f = 0$  in  $\Omega^1 = \{\mathbf{x} : \psi(\mathbf{x}) < 0\}$ , it is easy to see that the exact solution is given by a constant field  $u^1 = 1$  in  $\Omega^1$  and a logarithmic field  $u^2 = \log r$  in  $\Omega^2$  where  $r$  is a radial coordinate defined with respect to the center of the field  $\mathbf{x}_c = (0.5, 0.5, 0.5)$ . The computational domain is shown in Figure 4.13.

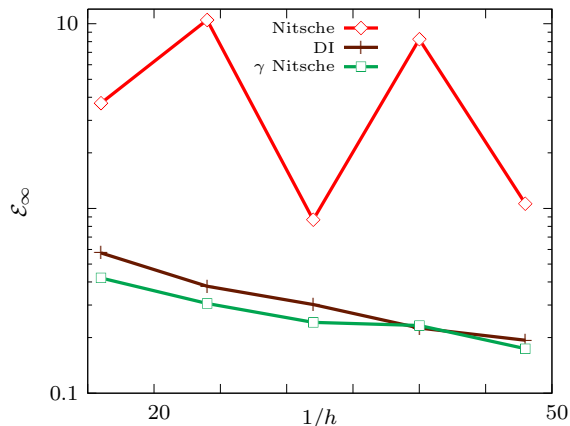


FIGURE 4.14 – Convergence of flux in  $L_\infty$  norm for the logarithmic field problem with a popcorn shaped embedded surface. In the figure legend, we denote the error in flux obtained from the domain integral formulation by DI.

We plot the maximum error in the average flux at the interface i.e.  $\langle \nabla u \rangle_\gamma \cdot \mathbf{n}^2$ , evaluated at each Gauss point in Figure 4.14. In Chapter 2, it was shown that on directly calculating the normal flux from the gradient of the field, classical Nitsche’s method fails to converge in the  $L_\infty$  norm. In contrast, from Figure 4.14 we can see that the weighted formulation proposed here converges with optimal rates. In fact, for this particular problem, in the  $L_\infty$  norm, the weighted formulation seems to have better accuracy than even the domain integral postprocessing technique to recover flux as proposed by Ji and Dolbow (2004).

## 4.5 Conclusion

In this work, we demonstrated the lack of robustness exhibited by the classical Nitsche formulation for a certain class of embedded interface problems. Indeed,

the bulk field remains reasonably well behaved even in these pathological cases and it is therefore essential to look at the behaviour of the interfacial field to comment on the robustness of the method. We applied numerical analysis to highlight the possible cause of this lack of robustness and proposed an alternate variational form based on a smarter choice for the weights in the Nitsche consistency terms.

We then demonstrated through several numerical examples the robustness of the proposed variational form and contrasted it against the standard form prevalent in literature. We report quadratic rates of convergence in the bulk field and linear rates of convergence in the interfacial fields. We also report minimal sensitivity with respect to the interface location as well as degree of heterogeneity on using the proposed form. Even though our simulations are restricted to embedded interface problems, we contend that the results hold for Nitsche based fictitious domain methods and discontinuous Galerkin methods such as the symmetric interior penalty approaches where there is a huge contrast in material properties and/or when there is a large gradient in the mesh density. We also contend that for applications where the interfacial field is of critical importance, such as evolving interface problems and frictional contact, it is imperative to use the weighted form of Nitsche's method.

Going forward, the robust traction recovery of the weighted Nitsche approach is encouraging for modeling the evolution of tangential traction through an inelastic dissipative mechanism. In the next Chapter, we will extend the proposed method to model such non-linear behavior at the interface.



# A Nitsche Stabilized Embedded Finite Element Method for Frictional Sliding Problems

## 5.1 Introduction

Robust enforcement of nonlinear constitutive laws (such as frictional contact) on embedded interfaces has remained a challenge. Key issues include the development of stable methodologies that preserve the accurate recovery of interfacial tractions. In this Chapter, we extend the weighted Nitsche's method, proposed in Chapter 4, to enforce frictional constraints at the interface, and contend that it is the most accurate, efficient, and robust method for this class of problems.

The first work to treat frictional contact on embedded interfaces was performed by Dolbow et al. (2001), in the context of the X-FEM. In that work, the Large Time INcrement (LATIN) method of Ladevéze (1999) was employed to enforce frictional contact constraints on crack surfaces. The LATIN method is essentially an augmented Lagrange multiplier method that has been widely used for domain decomposition problems. It possesses the advantages of a symmetric system matrix, and does not require the calculation of a global tangent operator. For a recent exten-

sion of the method to three-dimensional problems, see Gravouil et al. (2011). Since the work of Dolbow et al. (2001), both dual and primal methods have been pursued for frictional contact constraints on embedded interfaces.

The challenge with Lagrange multiplier methods for this class of problems concerns the construction of a stable Lagrange multiplier space. Stability is crucial in this case to avoid artificial oscillations in the interfacial tractions. Such a stable space was constructed by Kim et al. (2007) through a discretization of the surface mesh independent from the underlying bulk mesh. However, the extension of their approach to three-dimensional problems is not immediately obvious. Béchet et al. (2009) constructed a nonlocal Lagrange multiplier field based on the underlying mesh (see Hautefeuille et al. (2012) for an extension of this approach to 3D). For an application of the method of Béchet et al. (2009) to large sliding contact, see Nistor et al. (2009).

Classical penalty methods have also been explored for this class of problems, see for example, the work of Liu and Borja (2008), and Khoei and Nikbakht (2007) for small deformation elastostatics. They noted the superior rates of convergence for the penalty method compared to the LATIN method. The approach has been extended to problems with bulk plasticity (Khoei et al., 2006; Liu and Borja, 2009) and large sliding contact (Khoei and Mousavi, 2010; Liu and Borja, 2010a). It should be noted that these approaches still represent a regularization of a discrete formulation that is unstable, and involve a free penalty parameter.

More recently, Liu and Borja (2010b) proposed a stabilized formulation for low-order elements, with the option of either Lagrange multipliers or penalty regularization to enforce the interfacial constraints. Unfortunately, the method still retains a free stabilization parameter as well as a free penalty parameter for the regularized case. Coon et al. (2011) proposed the use of classical Nitsche's method to model frictional contact on embedded interfaces with the X-FEM. In that work, however,

the Nitsche stabilization parameter was left free, and a smoothing technique was developed for calculating the frictional tractions. Classical Nitsche’s method was also proposed for frictionless contact with standard finite element approximations by Wriggers and Zavarise (2008).

In this Chapter, we advocate the use of weighted Nitsche’s method, proposed in Chapter 4, for modeling frictional contact on embedded interfaces. There are many advantages to the weighted Nitsche’s approach over the aforementioned methods. First and foremost, it is a primal method that does not introduce additional degrees of freedom at the embedded interface. Secondly, upon linearization the method provides fast convergence in the displacement field. Thirdly, as described in Chapter 4, with the aid of numerical analysis and low-order elements, the method does not contain any free parameters. Finally, in comparison to the standard Nitsche method, our approach yields much more accurate approximations to the interfacial traction fields. This accuracy is essential for robustly capturing the evolution of the interfacial slip.

This Chapter is outlined as follows. In the next Section, we introduce the model problem and the associated variational formulation. In Section 5.3, the discretized forms for the embedded finite element method with weighted Nitsche stabilization are presented. Numerical examples demonstrating the accuracy and efficacy of the approach are then presented in Section 5.4. Finally, the last Section provides a summary, some concluding remarks and possible extensions.

## 5.2 Model problem and variational formulation

### 5.2.1 *Governing equations for linear elastostatics and sliding contact*

We begin by considering two domains  $\Omega^1$  and  $\Omega^2$  separated from each other by an interface  $\Gamma_*$  (see Figure 5.1). In this work, we will confine attention to two-dimensional problems, although the extension to three-dimensional problems is conceptually iden-

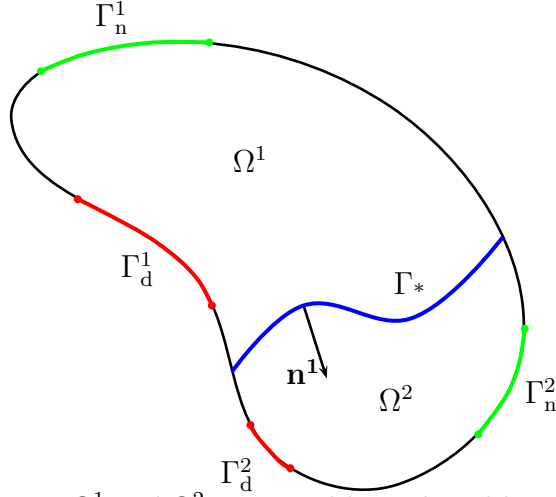


FIGURE 5.1 – Domains  $\Omega^1$  and  $\Omega^2$  separated by a shared boundary  $\Gamma_*$ . The Dirichlet boundaries  $\Gamma_d^1$ ,  $\Gamma_d^2$  and the Neumann boundaries  $\Gamma_n^1$  and  $\Gamma_n^2$  are as shown. The complementary part of the boundary is traction free. The normal to the boundary of each sub-domain,  $\mathbf{n}^m$ , points outwards from the domain  $\Omega^m$  as shown.

tical. Force balance is assumed to hold in each domain and across the interface. The governing equations in each of the domains are given, in indicial notation, by:

$$\begin{aligned} \sigma_{ij,j}^m &= 0 & \text{in } \Omega^m, \\ u_i^m &= u_d^m & \text{on } \Gamma_d^m, \\ \sigma_{ij}^m n_j^m &= h_i^m & \text{on } \Gamma_n^m, \end{aligned} \quad (5.1)$$

where  $\sigma_{ij}^m$  and  $u_i^m$  denote the components of the stress and displacement fields in domain  $\Omega^m$ , respectively, and  $n_j^m$  the components of the unit outward normal. The displacement is fixed to the prescribed field  $u_d^m$  on the Dirichlet portion of the boundary, and  $h_i^m$  denotes the prescribed traction on the Neumann portion of the boundary.

Introducing the traction field on the embedded interface by projecting the stress from each domain,

$$t_i^m = \sigma_{ij}^m n_j^m \quad \text{on } \Gamma_*. \quad (5.2)$$

Force balance on the interface is given by

$$t_i^1 + t_i^2 = 0 \quad \text{on } \Gamma_*. \quad (5.3)$$

We assume a linear elastic response for the constitutive relationship in the bulk domain:

$$\sigma_{ij}^m = C_{ijkl}^m u_{(k,l)}^m \quad \text{in } \Omega^m, \quad (5.4)$$

where  $C_{ijkl}^m$  denotes the fourth-order elasticity tensor, and  $u_{(k,l)}^m$  is the symmetric gradient of the displacement field.

Regarding the interfacial kinematics and constitutive laws, given (5.3), we find it convenient to use a single interfacial traction field, defined by

$$t_i \equiv -t_i^1 = t_i^2 \quad \text{on } \Gamma_*, \quad (5.5)$$

and express things in the  $\mathbf{n}^1 - \boldsymbol{\tau}^1$  plane of the interface. In particular, consider the following decompositions of interfacial tractions and displacements into normal and tangential components

$$t_i = t_N n_i^1 + t_\tau \tau_i^1, \quad u_i^m = u_N^m n_i^1 + u_\tau^m \tau_i^1, \quad \text{on } \Gamma_*. \quad (5.6)$$

We assume that sliding conditions hold on the interface, such that there exists no gap in the normal component of the displacement field, *i.e.*

$$u_N^1 = u_N^2 \quad \text{on } \Gamma_*, \quad (5.7)$$

while the tangential component of the displacement field is allowed to be discontinuous, and is strongly tied to the tangential traction on the interface. In particular, we assume that the tangential traction evolves through a given inelastic dissipative law at the interface. We demonstrate our proposed method by considering two interfacial laws: (a) perfect plasticity and (b) Coulombian frictional behavior. However, the proposed framework can also be used to model more complex interfacial behavior. As there is a strong analogy between the governing equations for both Coulombian friction and elastic-perfectly plastic behavior, we list the governing equations for both these cases in a common framework following Simo and Hughes (1998) below.

We use  $[[u_j]] = u_j^2 - u_j^1$  to denote the jump in the displacement field across the interface. We first additively decompose the tangential displacement into an “elastic” or recoverable part and a non-recoverable plastic part, such that:

$$[[u_\tau]] = [[u_\tau^{el}]] + [[u_\tau^{pl}]]. \quad (5.8)$$

The flow rule describing sliding and the yield (or slip) function are written in rate form, and given by:

$$[[\dot{u}_\tau^{pl}]] = \dot{\beta} \frac{t_\tau}{||t_\tau||}, \quad (5.9)$$

$$\phi(t_\tau) = ||t_\tau|| - t_Y, \quad (5.10)$$

where  $\dot{\beta}$  represents the slip rate and  $t_Y$  represents the yield traction. For a perfectly plastic interface, this yield traction assumes a constant value and is a material characteristic property. In the case of Coulombian behavior this traction depends on the normal traction such that  $t_Y = \mu t_N$ , where  $\mu$  represents a coefficient of friction between the two materials. Finally, the Kuhn-Tucker complementarity conditions and the consistency equation are given by:

$$\begin{aligned} \dot{\beta} \geq 0, \quad \phi(t_\tau) \leq 0, \quad \dot{\beta}\phi = 0, \\ \dot{\beta}\dot{\phi} = 0 \text{ (if } \phi = 0). \end{aligned} \quad (5.11)$$

The first line in equation (5.11) specifies the requirements on the admissibility of the traction field. It also states that any slip only occurs on the yield surface.

### 5.2.2 Variational form

The variational statement can now be derived by multiplying the governing equations in each domain by a corresponding weighting function and summing them together.

On application of the divergence theorem, we have:

$$\sum_m \int_{\Omega^m} w_{(i,j)}^m \sigma_{ij}^m \, d\Omega - \sum_m \int_{\Gamma_*} w_i^m t_i^m \, d\Gamma = \sum_m \int_{\Gamma_m} w_i^m h_i^m \, d\Gamma. \quad (5.12)$$

Upon using the traction continuity (5.3) and the definition of the interfacial traction (5.5), we obtain the following weak form: Find  $u_i \in \mathcal{U}_i$  such that

$$\sum_m \int_{\Omega^m} w_{(i,j)}^m \sigma_{ij}^m \, d\Omega - \int_{\Gamma_*} \llbracket w_i \rrbracket t_i \, d\Gamma = \sum_m \int_{\Gamma_m} w_i^m h_i^m \, d\Gamma \quad \forall w_i \in \mathcal{V}_i, \quad (5.13)$$

where  $\llbracket w \rrbracket = w^2 - w^1$  represents the interfacial jump in the variations. Here,  $\mathcal{U}_i$  and  $\mathcal{V}_i$  represent spaces of sufficiently smooth functions for the displacements and variations, respectively.

## 5.3 Algorithmic treatment and discretization

### 5.3.1 Weighted Nitsche's method for frictional sliding

We focus attention on the contact virtual work. The second integral in (5.13) can be rewritten as a sum of normal and tangential contributions, *viz.*

$$\int_{\Gamma_*} \llbracket w_i \rrbracket t_i \, d\Gamma = \int_{\Gamma_*} (\llbracket w_N \rrbracket t_N + \llbracket w_\tau \rrbracket t_\tau) \, d\Gamma. \quad (5.14)$$

Methods for the treatment of contact are largely differentiated by the algorithmic treatment of the interfacial traction (Laursen, 2003; Wriggers, 2006). Along these lines, we begin by defining a weighted interfacial pressure and shear:

$$p_\gamma = n_i^1 \langle \sigma_{ij} \rangle_\gamma n_j^1, \quad f_\gamma = \tau_i^1 \langle \sigma_{ij} \rangle_\gamma n_j^1, \quad (5.15)$$

where  $\langle \sigma_{ij} \rangle_\gamma = \gamma^1 \sigma_{ij}^1 n_j^1 + \gamma^2 \sigma_{ij}^2 n_j^1$ , represents a weighted average of stress across the interface. Here, the weights  $\gamma^1 > 0$  and  $\gamma^2 > 0$  are only required to satisfy  $\gamma^1 + \gamma^2 = 1$ .

In the weighted Nitsche’s method for frictional sliding, we replace the normal traction in (5.14) by

$$t_N = p_\gamma - \alpha_N \llbracket u_N \rrbracket, \quad (5.16)$$

where  $\alpha_N > 0$  is a stabilization parameter. This expression is analogous to what appears in augmented Lagrangian treatments of contact, where (for frictional sliding) one would find the normal traction replaced by

$$t_N = \lambda_N - \epsilon_N \llbracket u_N \rrbracket, \quad (5.17)$$

with  $\lambda_N$  the Lagrange multiplier and  $\epsilon_N$  the penalty parameter. The primary difference between Nitsche’s method for contact and augmented Lagrangian treatments is that the former does not introduce an additional field. Further, the parameter  $\alpha_N$  doesn’t serve so much to enforce the non-interpenetrability constraint as it does to ensure the coercivity of the bilinear form. See Wriggers and Zavarise (2008) for additional discussion comparing Nitsche’s method to penalty and augmented Lagrangian methods.

Similar to equation (5.16), we now introduce the weighted Nitsche regularization for the tangential traction:

$$t_\tau = f_\gamma - \alpha_\tau (\llbracket u_\tau \rrbracket - \llbracket u_\tau^{pl} \rrbracket). \quad (5.18)$$

This equation describes a linear relationship between the “elastic” part of the tangential slip and the tangential tractions. The basic difference between the two equations (5.16) and (5.18) stems from the fact that there is no “inelastic” or non-recoverable component in the equations describing normal tractions. Here, the stabilization parameter  $\alpha_\tau$  serves as the slope of the regularized traction-displacement relationship in the tangential direction. This expression, too, is analogous to what appears in augmented Lagrangian treatments of frictional contact (see, e.g. Laursen, 2003).



Substitution of (5.16) and (5.18) into (5.14) and (5.13) results in a weak form that is non-symmetric. A symmetric weak form can be recovered by adding in symmetric analogs to (5.16), (5.18), and (5.14). As examples, see the weak forms developed by Wriggers and Zavarise (2008) for frictionless contact and that of Coon et al. (2011) for frictional contact. As discussed in Simo and Laursen (1992) and Liu and Borja (2008), for Coulombian friction models, the consistent tangent matrix is naturally non-symmetric due to the coupling between the normal and tangential directions. Accordingly, we leave out such contributions to the bilinear form to arrive at a variationally consistent but non-symmetric method. The precise definitions of the weights and the stabilization parameters that arise in the formulation is detailed at an element level in Section 5.3.4.

### 5.3.2 Spatial discretization

We triangulate the bulk domains  $\Omega^1$  and  $\Omega^2$  into a set of non-overlapping linear triangles. Further, we consider the interface  $\Gamma_*$  to be ‘embedded’ in the domain; *i.e.* we represent the interface independently of the bulk discretization so that it cuts through elements. We follow Hansbo and Hansbo (2002) and construct the discretization in the vicinity of embedded interface through overlapping elements also described previously in Chapters 3 and 4. We also follow Chapters 3 and 4 and locate the embedded interface implicitly through a level-set function. Figure 5.2 shows a schematic of the followed approach.

The finite element discretization is now constructed on these overlapping domains as described in Hansbo and Hansbo (2002). The approximation and the weighting functions are then given by:

$$\mathbf{u}^h = \sum_m \sum_{i \in I^m} H^m \mathbf{N}_i \mathbf{u}_i, \quad \mathbf{w}^h = \sum_m \sum_{i \in I^m} H^m \mathbf{N}_i \mathbf{w}_i, \quad (5.19)$$

where,  $I^m$  is the set of all nodes whose supports overlap the domain  $\Omega^m$  and  $H^m$  is

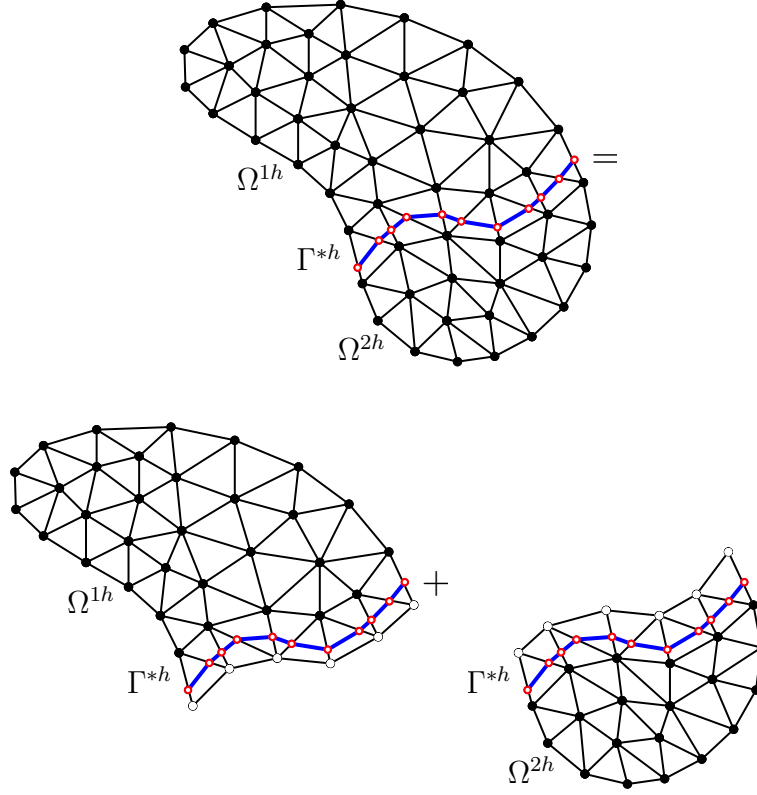


FIGURE 5.2 – Spatial discretization for the embedded interface formulation. The black circles are the physical nodes corresponding to the background mesh and the hollow circles are the phantom nodes. The red hollow circles represent the discretization of the embedded interface within an element. The discrete interface is constructed to be piecewise-planar throughout the mesh, and is based on a zero iso-surface of the level-set function.

the characteristic function given by

$$H^m(\mathbf{x}) = \begin{cases} 1 & \text{if } \mathbf{x} \in \Omega^m, \\ 0 & \text{otherwise.} \end{cases} \quad (5.20)$$

### 5.3.3 Discrete equations

On introducing the discretization for the approximation and the weighting spaces as specified above into the variational form (5.13), it is straightforward to obtain the following discrete statement of equilibrium in the residual form:

$$\mathbf{R}(\mathbf{u}) = \mathbf{F}_{\text{ext}} - (\mathbf{F}_{\text{int}}^b + \mathbf{F}_{\text{int}}^c) = \mathbf{0}. \quad (5.21)$$

---

**Algorithm 4** Updating interfacial tractions for an element intersected by an embedded interface at the (n+1)-th load step and k-th iteration

---

**for all** gauss-points  $gp$  **do**

  Compute jump in the displacements:  $\llbracket \mathbf{u} \rrbracket_{n+1}^{(k)}$

  Compute trial traction:  $\mathbf{t}_{n+1}^{\text{trial}, (k)} = \langle \boldsymbol{\sigma}_{n+1}^{(k)} \rangle_{\gamma} \mathbf{n}^1 - \alpha^e \mathbf{I} \llbracket \mathbf{u} \rrbracket_{n+1}^{(k)}$

  Compute trial yield function:  $\phi_{n+1}^{\text{trial}, (k)} = \phi(t_{\tau, n+1}^{\text{trial}, (k)}) = \|t_{\tau, n+1}^{\text{trial}, (k)}\| - t_Y$

**if** yield condition is not violated:  $\phi_{n+1}^{\text{trial}, (k)} \leq 0$  **then**

    Trial state is true state:  $\mathbf{t}_{n+1}^{(k)} = \mathbf{t}_{n+1}^{\text{trial}, (k)}$

**else**

    Normal direction - no yielding:  $t_{N, n+1}^{(k)} = t_{N, n+1}^{\text{trial}, (k)}$

    Tangential direction - return mapping algorithm:

$$\Delta\beta = \frac{\phi_{n+1}^{\text{trial}, (k)}}{\alpha^e}$$

$$t_{\tau, n+1}^{(k)} = t_{\tau, n+1}^{\text{trial}, (k)} - \alpha^e \Delta\beta \frac{t_{\tau, n+1}^{\text{trial}, (k)}}{\|t_{\tau, n+1}^{\text{trial}, (k)}\|}$$

$$\llbracket u \rrbracket_{\tau, n+1}^{pl, (k)} = \llbracket u \rrbracket_{\tau, n}^{pl} + \Delta\beta \frac{t_{\tau, n+1}^{\text{trial}, (k)}}{\|t_{\tau, n+1}^{\text{trial}, (k)}\|}$$

    Update  $\alpha^e$

**end if**  
**end for**

---

In the absence of body forces,  $\mathbf{F}_{\text{ext}}$  is obtained by locally assembling the element contributions from traction boundary conditions on the Neumann boundary:

$$\mathbf{f}_{\text{ext}}^m = \int_{\Gamma_{n^e}^m} \mathbf{N}^T \mathbf{h}^m \, d\Gamma_e \quad \text{for } m = 1, 2. \quad (5.22)$$

The internal force vector has contributions from both bulk material and contact tractions. The bulk contribution to the internal force vector,  $\mathbf{F}_{\text{int}}^b$ , is obtained by assembling:

$$\mathbf{f}_{\text{int}}^{b, m} = \int_{\Omega_e^m} (\mathbf{B}^T \mathbf{D}^m \mathbf{B}) \mathbf{u}_e^m \, d\Omega_e \quad \text{for } m = 1, 2, \quad (5.23)$$

where the matrix  $\mathbf{B}$  contains the shape function derivatives, the matrix  $\mathbf{D}^m$  denotes the discrete counterpart of the elasticity tensor and  $\mathbf{u}_e^m$  is the local vector of nodal unknowns. Finally, the contact contribution to the internal force vector is obtained

by assembling:

$$\mathbf{f}_{\text{int}}^c = \int_{\Gamma_{*e}} \mathbf{N}^T \mathbf{t}(\llbracket \mathbf{u} \rrbracket) d\Gamma_e, \quad (5.24)$$

where  $\mathbf{t}(\llbracket \mathbf{u} \rrbracket)$  is the contact traction. Clearly, the resulting problem is nonlinear as the tangential tractions have a dependence on tangential slip.

We apply the external load incrementally and solve the nonlinear set of equations (5.21) at each load step using the Newton-Raphson iterative scheme. We linearize the internal force vector,  $\mathbf{F}_{\text{int}, n+1}^{\text{b}, (k)} + \mathbf{F}_{\text{int}, n+1}^{\text{c}, (k)}$ , about the current state, defined by  $\mathbf{u}_{n+1}^{(k)}$ , using a first-order Taylor series expansion to obtain:

$$\mathbf{K}_{n+1}^{(k)} \Delta \mathbf{u}_{n+1}^{(k+1)} = \mathbf{F}_{\text{ext}, n+1} - (\mathbf{F}_{\text{int}, n+1}^{\text{b}, (k)} + \mathbf{F}_{\text{int}, n+1}^{\text{c}, (k)}), \quad (5.25)$$

We solve for the incremental nodal displacement,  $\Delta \mathbf{u}_{n+1}^{(k+1)}$ , at the k-th iteration. For notational ease, subsequently, we omit the superscript, k, and the subscript, n+1, denoting the iteration and load counters respectively. The tangent matrix, at the (n+1)-th load step and the k-th iteration is now denoted by  $\mathbf{K}$ , such that:

$$\mathbf{K} = \frac{\partial(\mathbf{F}_{\text{int}}^{\text{b}} + \mathbf{F}_{\text{int}}^{\text{c}})}{\partial \mathbf{u}}. \quad (5.26)$$

On substituting the global internal force vector assembled using (5.23) and (5.24) in (5.26), it is easy to obtain the following block structure for the tangent matrix:

$$\mathbf{K} = \begin{bmatrix} \mathbf{K}^{\text{b},1} + \mathbf{K}_{\text{d}}^{\text{c},1} & \mathbf{K}_{\text{od}}^{\text{c},1} \\ \mathbf{K}_{\text{od}}^{\text{c},2} & \mathbf{K}^{\text{b},2} + \mathbf{K}_{\text{d}}^{\text{c},2} \end{bmatrix}, \quad (5.27)$$

where the matrices  $\mathbf{K}^{\text{b},m}$  arise from the linearization of the bulk contribution to the internal force vector,  $\mathbf{F}_{\text{int}}^{\text{b}}$ . These can be assembled from the local matrices:

$$\mathbf{k}^{\text{b},m} = \int_{\Omega_e^m} \mathbf{B}^T \mathbf{D}^m \mathbf{B} d\Omega_e \quad \text{for } m = 1, 2. \quad (5.28)$$

In addition, we also have contributions to the leading diagonal blocks and off-diagonal blocks of the tangent matrix *viz.*  $\mathbf{K}_d^{c,m}$  and  $\mathbf{K}_{od}^{c,m}$  respectively. These contributions arise from the linearization of the contact contribution to the internal force vector,  $\mathbf{F}_{int}^c$ .

Clearly these contributions have a direct dependence on the interfacial tractions. These tractions are updated by applying the classical return mapping algorithm over the incremental form of the rate equations (5.8)–(5.11) which govern the evolution for tangential tractions. Algorithm 4 describes the traction update procedure within a standard finite element loop. While the interface is still under perfect contact, the matrices can be assembled from the following local matrices:

$$\begin{aligned}
\mathbf{k}_d^{c,m} &= \int_{\Gamma_{*e}} \mathbf{N}^T \boldsymbol{\alpha}_e \mathbf{N} \, d\Gamma_e - \gamma_e^m \int_{\Gamma_{*e}} \mathbf{N}^T (\mathbf{n}^m)^T \mathbf{D}^m \mathbf{B} \, d\Gamma_e \quad \text{for } m = 1, 2, \\
\mathbf{k}_{od}^{c,1} &= - \int_{\Gamma_{*e}} \mathbf{N}^T \boldsymbol{\alpha}_e \mathbf{N} \, d\Gamma_e + \gamma_e^2 \int_{\Gamma_{*e}} \mathbf{N}^T (\mathbf{n}^2)^T \mathbf{D}^2 \mathbf{B} \, d\Gamma_e, \\
\mathbf{k}_{od}^{c,2} &= - \int_{\Gamma_{*e}} \mathbf{N}^T \boldsymbol{\alpha}_e \mathbf{N} \, d\Gamma_e + \gamma_e^1 \int_{\Gamma_{*e}} \mathbf{N}^T (\mathbf{n}^1)^T \mathbf{D}^1 \mathbf{B} \, d\Gamma_e.
\end{aligned} \tag{5.29}$$

When in perfect contact, the stabilization parameters in the normal and tangential directions are chosen identically such that  $\boldsymbol{\alpha}_e = \alpha \mathbf{I}$ , where the stabilization parameter  $\alpha$  and the interfacial weights  $\gamma_e^m$  are evaluated locally as detailed in Section 5.3.4.

At the onset of sliding, the linearization of the normal traction remains the same as given by (5.29), however, the linearization of the tangential traction depends on the particular form of the interfacial response. In the case of a perfectly plastic interface, we have a constant value of tangential traction and hence no contribution to the tangent matrix. We enforce this in the  $\mathbf{n} - \boldsymbol{\tau}$  plane by the following:

$$\boldsymbol{\alpha}_e = \begin{bmatrix} \alpha & 0 \\ 0 & 0 \end{bmatrix} ; \quad \frac{\partial f_\gamma}{\partial \mathbf{u}} = \mathbf{0}, \tag{5.30}$$

where  $f_\gamma$  is given by (5.15) and  $\partial f_\gamma / \partial \mathbf{u}$  is a row vector in the matrix  $(\mathbf{n}^m)^\top \mathbf{D}^m \mathbf{B}$  that contributes towards the tangential traction. When the interface exhibits a Coulombian frictional response, the tangential traction is given as a characteristic coefficient of friction,  $\mu$ , times the normal traction. This can be enforced in the  $\mathbf{n} - \boldsymbol{\tau}$  plane by the following:

$$\boldsymbol{\alpha}_e = \begin{bmatrix} \alpha & 0 \\ \mu\alpha & 0 \end{bmatrix} ; \quad \frac{\partial f_\gamma}{\partial \mathbf{u}} = \mu \frac{\partial p_\gamma}{\partial \mathbf{u}}, \quad (5.31)$$

where  $p_\gamma$  is also given by (5.15) and  $\partial p_\gamma / \partial \mathbf{u}$  represents a row vector in the matrix  $(\mathbf{n}^m)^\top \mathbf{D}^m \mathbf{B}$  that contributes towards the normal traction.

As a final remark, we mention that the stick-slip response is identified at the gauss-points of integration for each interfacial segment. Consequently, we can also have interfacial elements with one gauss point exhibiting a sliding response while the other exhibits a stick-state. Since we are summing up the contributions from each gauss point to the local tangent matrix, this behavior does not warrant any additional consideration.

#### 5.3.4 Estimate for the stabilization parameter and interfacial weights

We now provide precise definitions to the weighting and stabilization parameters, locally, following Chapter 4. Before the onset of sliding, the stabilization parameter in both normal and tangential directions is identically chosen as:

$$\alpha = 2 \times \text{meas}(\Gamma_{*e}) \left( \frac{|\mathbf{D}^1|(\gamma_e^1)^2}{\text{meas}(\Omega_e^1)} + \frac{|\mathbf{D}^2|(\gamma_e^2)^2}{\text{meas}(\Omega_e^2)} \right). \quad (5.32)$$

This choice ensures that the bilinear form remains coercive and thus ensures stability. The analysis required to arrive at the above expression is detailed previously in Chapter 3 for vector problems and Chapter 4 for scalar problems. Further, as described in Chapter 4, we choose an interfacial weights that minimize the stabilization

parameter while ensuring coercivity:

$$\gamma_e^m = \frac{\text{meas}(\Omega_e^m)/|\mathbf{D}^m|}{\text{meas}(\Omega_e^1)/|\mathbf{D}^1| + \text{meas}(\Omega_e^2)/|\mathbf{D}^2|}, \quad m = 1, 2 \quad (5.33)$$

Finally, it is also worth highlighting some of the differences that arise in the nonlinear setting as opposed to the linear case where the aforementioned analysis was conducted. Firstly, the bilinear form we are advocating is non-symmetric in contrast to the symmetric bilinear form of linear problems. We could still use the stabilization parameter obtained from analyzing symmetric problems because the symmetrization term has a negative contribution to the bilinear form. From a coercivity standpoint, omitting this term only makes the parameter more conservative.

Secondly, the frictional contact constraint we are enforcing here is, in the tangential direction, akin to enforcing a stiff Neumann constraint before the onset of sliding and a Neumann constraint thereafter. In the normal direction, on the other hand, we have a stiff Neumann constraint throughout. As discussed in Chapter 4, the stability issues in these enriched approaches arise only while imposing stiff Neumann or Dirichlet type constraints. A standard Neumann constraint can be handled weakly just as in classical finite element methods. Therefore, while the normal direction is handled throughout using the proposed stabilized approach, in the tangential direction, the stabilized approach is only used to handle perfectly-tied kinematics.

## 5.4 Numerical examples

In this section, we consider several benchmark examples to validate the performance of the proposed approach. In the figure legends, the weighted form of Nitsche's method is referred as  $\gamma$ -Nitsche. We provide plots for both the bulk fields as well as the contact tractions and compare them with existing studies. In particular, we compare and contrast the performance of the weighted Nitsche's method with the

classical Nitsche’s method (with weights identically chosen as 0.5) and the penalty method to illustrate the advantages of the proposed method.

To load the structure, we follow a displacement based incremental loading scheme. In particular, we prescribe an initial value for the displacement increment such that we reach the full load in 25 steps. However, if we fail to converge at any step, we reduce the increment in half until we see convergence. Once this reduction is done, if convergence occurs in less than 4 iterations for more than four subsequent load steps, we double the increment until we reach the initial value. We follow a similar procedure while unloading as well.

#### 5.4.1 *Perfectly plastic interface*

In the following examples, we consider the interface to follow perfect plasticity.

##### *Compressive loading of a composite beam*

As a first example, we consider a composite beam in a state of plane stress and loaded in pure compression. We consider a straight vertical interface  $\Gamma_* = \{\mathbf{x} : \psi(\mathbf{x}) = 0\}$ ; where  $\psi(\mathbf{x})$  is given by the signed-distance function  $\mathbf{x} - \mathbf{x}_*$ . This interface partitions the domain  $\Omega$  into component sub-domains  $\Omega^1 = \{\mathbf{x} : \psi(\mathbf{x}) < 0\}$  and  $\Omega^2 = \{\mathbf{x} : \psi(\mathbf{x}) > 0\}$  as shown in Figure 5.3(a). We consider identical Young’s moduli on both sides of the interface such that  $E^1 = E^2 = 1000$  units. However, we only allow the material in  $\Omega^1$  to behave under Poisson’s effect; in particular, we choose  $\nu^1 = 0.3$  and  $\nu^2 = 0$ . This choice naturally leads us to a problem where we expect sliding at the interface. We characterize this sliding behavior by considering perfect plasticity at the interface with a yield traction  $t_Y$ . By varying the value for this yield traction  $t_Y$ , we can simulate different material response - ranging from frictionless sliding to perfect contact at the interface. The above problem set-up is motivated by the contact patch test described in Crisfield (2000) and is an extension of the set-up



described there to frictional sliding problems. This problem has also been studied previously with classical Nitsche and penalty approaches by Mayr (2010).

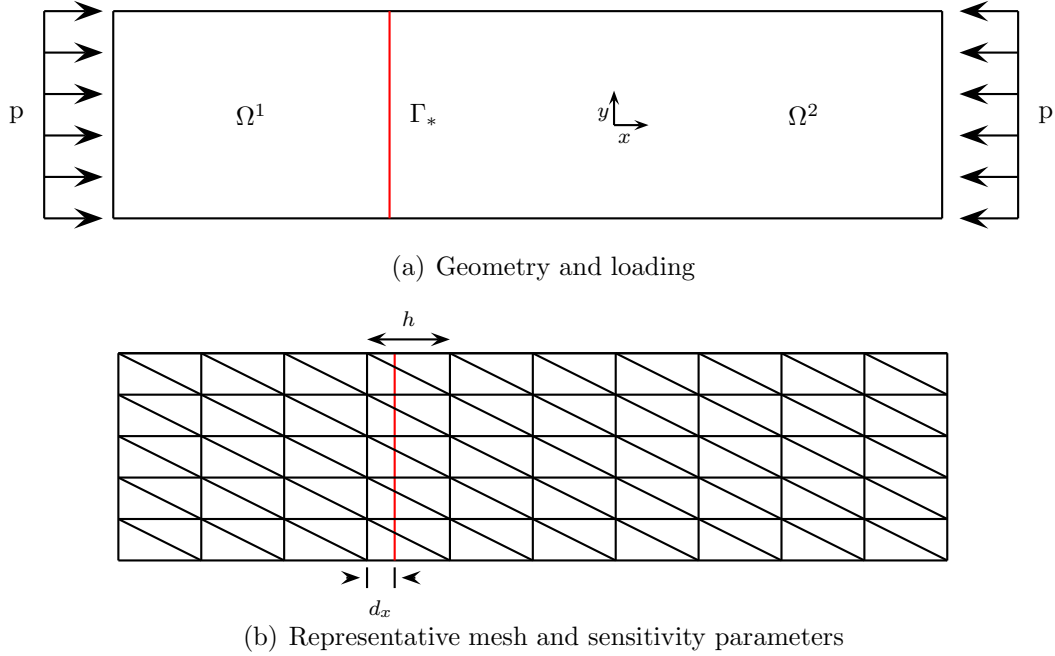


FIGURE 5.3 – A beam with a material interface loaded in pure compression: (a) illustration of geometry and loading conditions (b) representative finite element mesh and definition of the sensitivity parameters.

We consider the beam to be  $L = 16$  units long and  $H = 4$  units high. On assuming a uniaxial state of stress  $\sigma_{xx} = -25$  units for the beam, it is easy to construct an analytical displacement field in the component sub-domains:

$$u_x(\mathbf{x}) = \left( (x_* + L/2)\epsilon_{xx}^1 + (L/2 - x_*)\epsilon_{xx}^2 \right) \frac{x}{L} \quad \text{in } \Omega^1 \cup \Omega^2, \quad (5.34)$$

$$u_y^m(\mathbf{x}) = y\epsilon_{yy}^m \quad \text{in } \Omega^m, \quad (5.35)$$

where the strains  $\epsilon_{xx}^m$  and  $\epsilon_{yy}^m$  are given as:

$$\epsilon_{xx}^m = \frac{\sigma_{xx}}{E^m} \quad \text{in } \Omega^m, \quad \epsilon_{yy}^m(\mathbf{x}) = -\frac{\nu^m}{E^m}\sigma_{xx} \quad \text{in } \Omega^m \quad (5.36)$$

These displacement values are now applied as Dirichlet boundary conditions on the boundaries  $\Gamma_d = \{\mathbf{x} : x = -L/2|L/2\}$ . The remaining surfaces are traction-free.

We now consider a structured triangular mesh with 72 divisions in the x-direction and 18 divisions in the y-direction. Since embedded methods are notorious with respect to interface configurations which result in arbitrarily small volume fractions, we deliberately choose a position of the interface that corresponds to a value of  $d_x/h = 0.8\%$ . The definitions of  $d_x$  and  $h$  are shown in Figure 5.3(b). In particular, for the first study, we fix the position of the interface such that  $x_* = -2.4462$ . Finally, as a convergence criterion for the Newton-Raphson algorithm, we specify a tolerance of  $1 \times 10^{-12}$  in the energy norm.

In Figures 5.4(a)-(f), we plot the y-component of displacement on returning back to the completely unloaded state for different values of the interfacial yield traction  $t_Y$ . This represents the residual plastic strain in the considered problem. In Figures 5.4(a) and 5.4(f), which represent the kinematically linear cases *viz.* frictionless sliding at the interface and a perfectly-tied interface respectively, we find the expected result of no residual strain. For the rest of the cases, *i.e.* Figures 5.4(b)-(e), we again find the expected trend of decreasing residual strain with increasing values of the interfacial yield traction.

On using the weighted form of Nitsche's method, 25 displacement increments was enough to obtain convergence in all the cases except when the yield traction  $t_Y = 0.5$ . For this case, we required 32 displacement increments. In contrast, classical Nitsche's method required much smaller increments and close to 180 load steps to reach the maximum prescribed load while satisfying the specified tolerance for convergence. This is however not a surprising result considering the poor performance of classical Nitsche's method, with respect to interfacial tractions, when the interface is close to either internal boundary as illustrated in several previous studies (Mourad et al., 2007; Dolbow and Harari, 2009; Sanders et al., 2009; Zunino et al., 2011). This also motivates our next study where we intend to study the sensitivity of the convergence behavior of Newton-Raphson algorithm with the interface position.

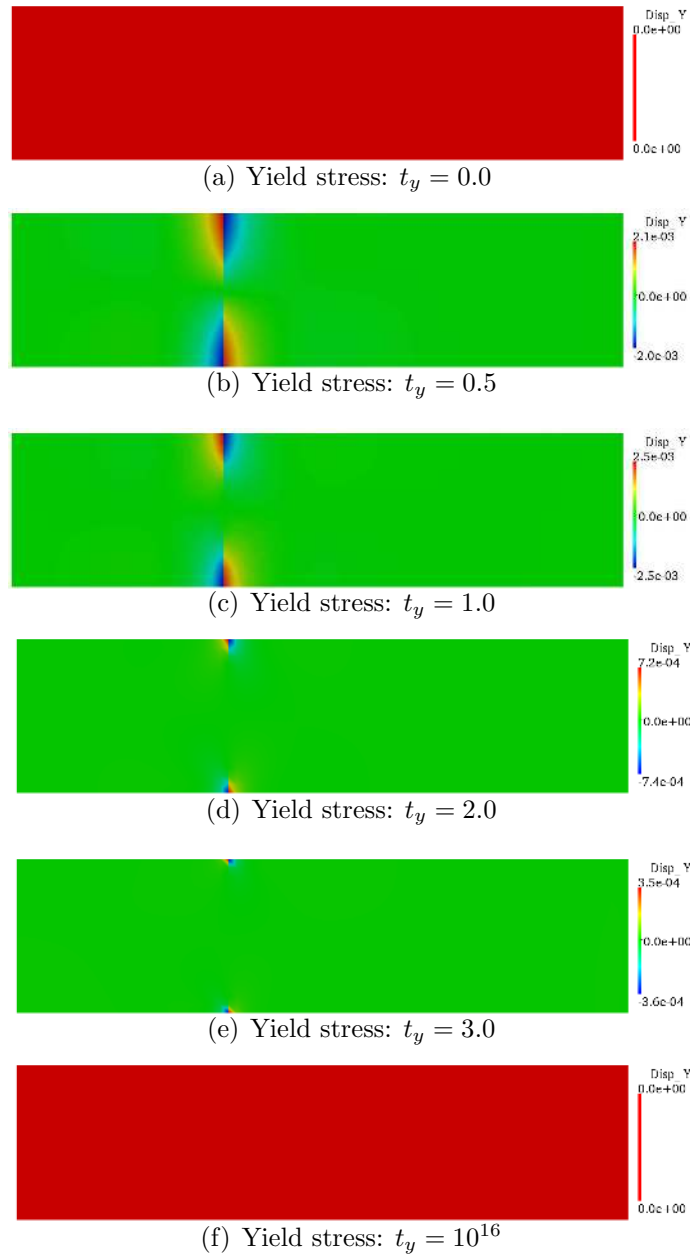


FIGURE 5.4 – Residual strain in the fully unloaded state with yield stress at the interface presented in ascending order. Case (a) represents a frictionless sliding problem while case (f) represents a perfectly tied interface. All the other subcases allow for frictional sliding.

**Table 5.1** – Newton-Raphson convergence behavior of weighted Nitsche’s method and Nitsche’s method with respect to position of interface within an element.

Normalized interface position $d_x/h$ (in %)	$\gamma$ -Nitsche’s method		Nitsche’s method	
	$n_{\text{load}}$	$n_{\text{iter}}$	$n_{\text{load}}$	$n_{\text{iter}}$
0.1	25	6	DNC*	-
0.2	30	5	DNC	-
0.4	30	5	DNC	-
0.8	32	5	DNC	-
5	27	5	DNC	-
10	26	5	25	9
50	25	6	25	6
90	25	5	25	10
95	28	5	34	6
99.2	32	5	175	6
99.6	25	5	337	7
99.8	25	5	DNC	-
99.9	25	5	DNC	-

\* DNC: Did not converge in 500 load steps.

*Sensitivity study*

We now replicate the problem set-up described in Section 5.4.1 and conduct a sensitivity study with respect to the interface position. Since we would get the largest yield zone when the yield traction is the smallest, we conduct this study for the case when  $t_Y = 0.5$ . We compare the performance of the proposed weighted form with classical Nitsche’s method as we move the interface from one internal boundary to another. The results of this sensitivity study are shown in Table 5.1. The first column in Table 5.1 contains the normalized interfacial position  $d_x/h$ . The second column shows the convergence behavior of weighted Nitsche’s method while the third column describes that of classical Nitsche’s method. We describe the convergence behavior in terms of two parameters: (a) minimum number of load steps required to reach the full load while converging to the specified tolerance in the energy norm (denoted by  $n_{\text{load}}$  in the Table) and (b) of all the converged load steps, the maximum

number of iterations required at any load step (denoted by  $n_{\text{iter}}$  in the Table). If the method fails to converge in 500 load steps, we state non-convergence (denoted by DNC = did not converge in the Table).

From Table 5.1, we notice that as the interface approaches either internal boundary, the convergence profile of the classical Nitsche’s method reduces drastically. In fact even as the measure of interface position  $d_x/h$  approaches 10% on either side, the rate of Newton-Raphson convergence falls as indicated by the larger number of iterations required for convergence. As illustrated in several previous studies (Mourad et al., 2007; Dolbow and Harari, 2009; Sanders et al., 2009; Zunino et al., 2011), as the interface approaches either internal boundary and yields arbitrarily small elements, the classical Nitsche’s method results in a deteriorating traction profile. High errors in traction can yield to false prediction of yielding at the interface. Additionally, these arbitrarily small element configurations also yield a particularly large value for the stabilization parameter. Since the stabilization parameter acts as an initial elastic stiffness in the return-mapping algorithm and therefore establishes an initial guess for the Newton-Raphson algorithm, it is not surprising that a large value leads the algorithm astray. The weighted Nitsche’s method, on the other hand, exhibits a much more uniform convergence pattern and shows minimal sensitivity to the interface position. As shown previously, this is a direct consequence of the much more robust traction recovery that characterizes this method.

#### 5.4.2 *Coulombian frictional law*

In the following examples, we consider the interface to follow Coulomb’s frictional law.

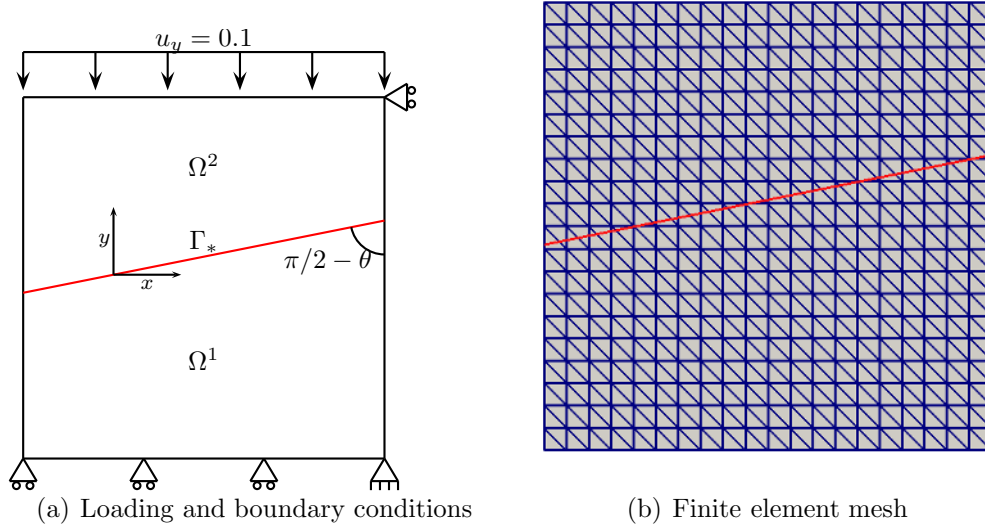


FIGURE 5.5 – A unit square domain with an inclined material interface that exhibits Coulombian frictional behavior (a) illustration of geometry and loading conditions - the interface is inclined such that  $\tan \theta = 0.2$  (b) finite element mesh used in computations.

### *Compressive loading of a plate with an inclined interface*

The first example we consider with a Coulombian frictional model involves a plate with an inclined material interface loaded in compression. The material interface  $\Gamma_* = \{\mathbf{x} : \psi(\mathbf{x}) = 0\}$  acts as an internal boundary to the domains  $\Omega^1$  and  $\Omega^2$ . The level set is given by the equation of an inclined plane  $\psi(\mathbf{x}) = y - 0.2x - 0.4586$ . The slope of the line defines the inclination of the interface and is equal to  $\tan \theta = 0.2$ . We consider identical material properties on both sides of the interface, *i.e.*  $E^1 = E^2 = 1000$  units and  $\nu^1 = \nu^2 = 0.3$ . The Coulomb's frictional coefficient characterizing the interface is chosen as  $\mu = 0.19$  and  $\mu = 0.21$  for two separate computations.

This problem has been studied previously by Dolbow et al. (2001) and Kim et al. (2007). It serves as a good benchmark for frictional sliding problems because of its close analogy with a classical problem in mechanics *viz.* a rigid block resting on a rough inclined plane. Analogous to that problem, we can predict slipping behavior when the frictional coefficient  $\mu < \tan \theta$  and a stick state otherwise. Therefore, since for the given interface  $\tan \theta = 0.2$ , we expect a sliding response when  $\mu = 0.19$  and

**Table 5.2** – Newton-Raphson convergence behavior of weighted Nitsche’s method for the problem where a plate with an inclined interface loaded in compression.

Iteration number	Energy norm
1	1.00e+00
2	9.41e-06
3	5.35e-32

a stick state otherwise.

We conduct the simulations on a unit square domain and use a structured triangular mesh with 20 divisions in each direction. The finite element mesh used in computations is shown in Figure 5.5(b). Also, we perform the computations using weighted Nitsche’s method. The loading is applied on the top surface  $\Gamma_d = \{\mathbf{x} : y = 1\}$  as a Dirichlet boundary condition such that  $u_y = -0.1$ . The nodes on the bottom surface are constrained from moving in y-direction. The problem set-up and the boundary conditions are shown in Figure 5.5(a). In addition, we apply the entire loading in one load step.

We plot the resulting response of the structure in Figures 5.6(a)-(b). Consistent with expectations, when the frictional coefficient between the materials is less than the tangent of the inclination of the surface, we recover slipping behavior while when it is larger, we see a stick state. For the stick response, the problem is linear and again consistent with expectations the method converged in two Newton-Raphson iterations. In contrast, when slipping is expected, the method took three iterations to converge. The convergence profile for the slipping case is given by Table 5.2.

*Sliding of an elastic block on a rigid surface*

As a next example, we investigate the performance of the method by simulating an elastic block sliding on a rigid surface. This is a standard benchmark study used for frictional contact problems and was first examined by Oden and Pires (1984)

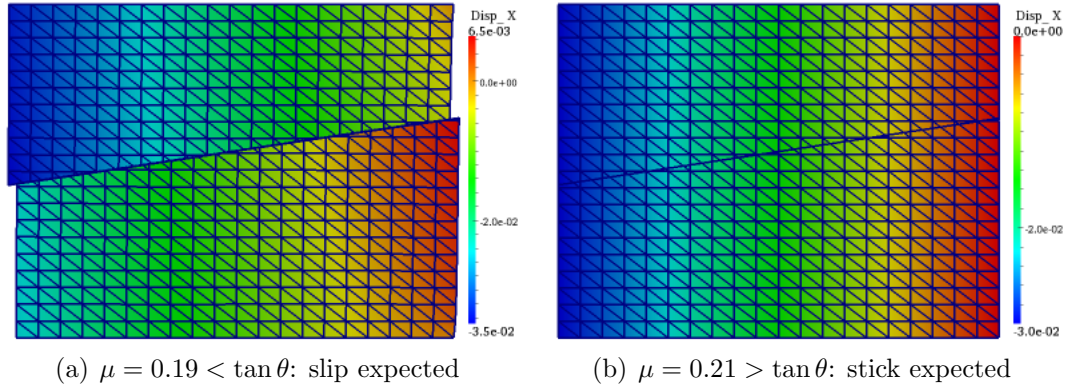


FIGURE 5.6 – Horizontal displacement contours and deformed geometry for compressive loading of a plate with an inclined material interface. The inclination of the interface is such that  $\tan \theta = 0.2$ . Slip is predicted (left) when the frictional coefficient  $\mu < \tan \theta$  while stick is predicted (right) when  $\mu > \tan \theta$ . The deformation is scaled by a factor of 2.

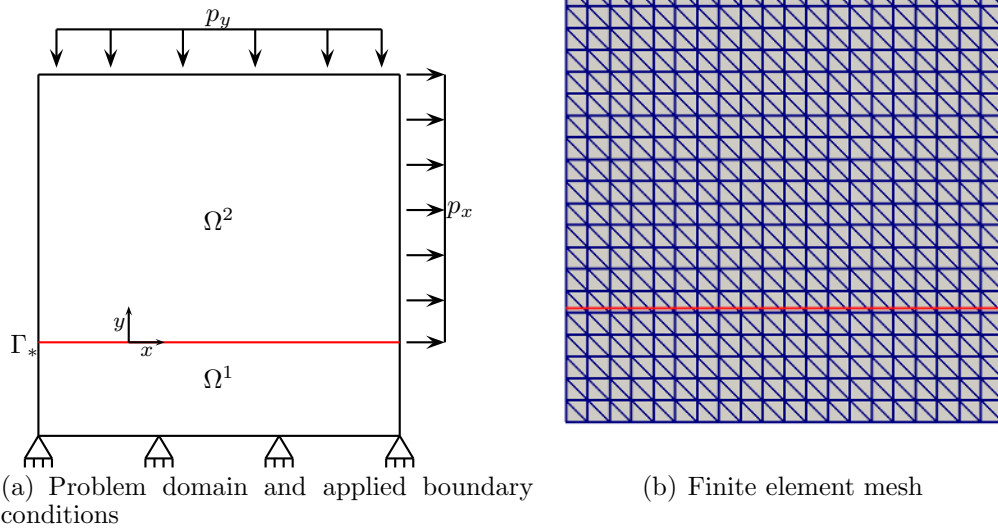


FIGURE 5.7 – Elastic block sliding on a rigid surface: (a) illustration of geometry and loading conditions: the applied traction boundary conditions are such that  $p_y = 200$  units and  $p_x = 60$  units. (b) Discretized problem domain used in computations.



and then by Wriggers et al. (1990) and Simo and Laursen (1992) in subsequent studies. This example serves as a more quantitative benchmark as we can compare the contact tractions with the existing studies. The problem set-up is described in Figure 5.7(a). We consider a square domain  $\Omega$  spanning  $(-2, 2) \times (-2, 2)$ . The material interface  $\Gamma_* = \{\mathbf{x} : \psi(\mathbf{x}) = 0\}$  partitions the domain into component sub-domains  $\Omega^1$  and  $\Omega^2$ . Here we consider a straight material interface  $\psi(\mathbf{x}) = y + 0.9644$ . We consider the material in the sub-domain  $\Omega^2$  to be elastic with a Young's modulus of  $E^2 = 1000$  units and Poisson's ratio of  $\nu^2 = 0.3$ . The material in sub-domain  $\Omega^1$  is considered rigid. In order to simulate rigid response, we choose  $E^1 = 10^{12}$  units and  $\nu^1 = 0$ . Consistent with earlier studies, the Coulomb frictional coefficient between the materials is chosen as  $\mu = 0.5$ . The applied tractions are as shown in Figure 5.7(a); on the top surface a compressive loading of  $p_y = 200$  units is applied. To remain consistent with earlier studies, the compressive traction is applied such that the first and last layer of elements from both the left and right boundaries are allowed to remain traction free. Also the extreme right boundary is pulled with a force of  $p_x = 60$  units. Additionally, in order to replicate the response of a rigid material, the bottom surface is constrained in both directions.

The computations are conducted on a structured triangular mesh with 20 divisions in each direction. The finite element mesh is shown in Figure 5.7(b). We remark that the chosen value of the level set results in an interfacial position that is not generally ideal for embedded methods. In particular, it is deliberately chosen to result in arbitrarily small volume fractions. Though the problem set-up can be considered rather idealized, in an embedded interface context it provides a convenient platform to examine the performance of the proposed approach in the presence of both arbitrarily small elements and large material heterogeneities. It is well known that embedded methods have problems when they encounter either of these cases independently (Zunino et al., 2011; Sanders et al., 2012; Dolbow and Harari, 2009;

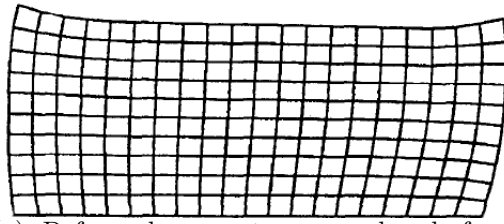
**Table 5.3** – Newton-Raphson convergence behavior of embedded methods for the sliding block problem.

Iteration number	Energy norm		
	$\gamma$ -Nitsche's method	Nitsche's method	Penalty method
1	1.00e+00	1.00e+00	1.00e+00
2	1.18e-04	1.29e+11	5.78e-02
3	9.25e-05	3.89e+15	2.23e-03
4	9.89e-09	7.57e+18	1.50e-05
5	5.53e-13	1.03e+24	3.31e-08
6	-	1.29e+27	4.36e-11
7	-	1.48e+30	4.50e-14
8	-	1.60e+33	-
9	-	1.06e+36	-
10	-	1.46e+40	-

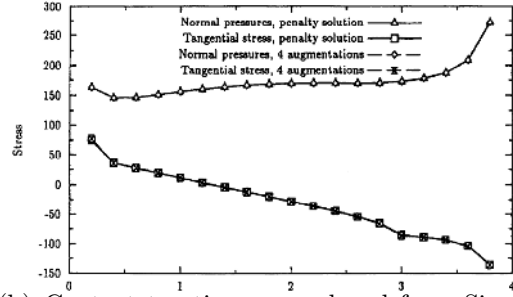
Mourad et al., 2007) so examining them in the presence of both provides a stiff challenge for the method. Here, we illustrate the robustness of the weighted Nitsche approach by contrasting it with both penalty and classical Nitsche's approaches.

In Table 5.3, we show the convergence profile of the Newton-Raphson method in the energy norm for all three embedded methods. Classical Nitsche's method fails to converge to the specified tolerance of  $1 \times 10^{-12}$  and in fact diverges. The penalty method on the other hand is highly sensitive to the choice of the penalty parameter. The right combination of the normal parameter  $\alpha_N$  and  $\alpha_\tau$  which results in convergence is hard to choose. The common practice of choosing a value that is orders of magnitude higher than the larger Elastic modulus leads to divergence. On the other hand, if the parameter is chosen closer to the softer Elastic modulus, the constraint is not satisfied well and we again fail to converge. Through some trial and error, however, the following choices resulted in convergence  $\alpha_N = 1 \times 10^9$  and  $\alpha_\tau = 8 \times 10^5$ . The convergence profile for this specified choice is shown in the fourth column of Table 5.3. The rate of convergence is slightly slower for the penalty

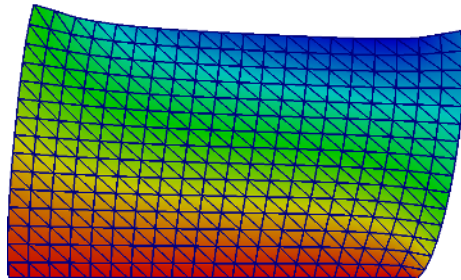
method and it takes seven iterations to reach the specified tolerance as opposed to five by the weighted Nitsche’s method. Additionally, as a word of caution, we also mention that a slightly different choice in the tangential direction,  $\alpha_\tau = 5 \times 10^5$ , led to divergence indicating a very high sensitivity to the penalty parameter.



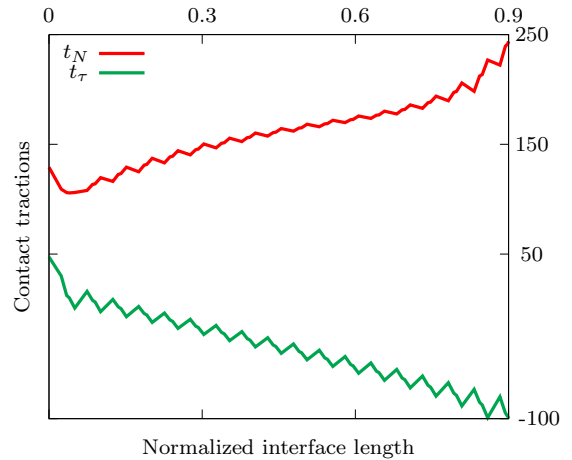
(a) Deformed geometry reproduced from Simo and Laursen (1992).



(b) Contact tractions reproduced from Simo and Laursen (1992).



(c) Deformed geometry using weighted Nitsche’s method. The plotted displacement contours are the y-components of the displacement.



(d) Contact tractions using weighted Nitsche’s method.

FIGURE 5.8 – Deformed geometry and contact tractions for the elastic block sliding on a rigid surface.

The deformed shape and the y-displacement contours obtained using the weighted Nitsche’s method are shown in Figure 5.8(c) and seem to be in good agreement with

earlier studies (see Figure 5.8(a)). We also compare the contact tractions for the weighted Nitsche’s method and the penalty method for the above specified values of penalty parameters. From Figure 5.8(d), we find that the weighted Nitsche’s method yields a traction profile that is in close agreement with existing studies (Figure 5.8(b)). The tractions do not give an exact quantitative match, but the differences are small. We can attribute this slight mismatch to various differences in the problem set-up. Firstly, we are trying to replicate the behavior of a rigid block using a very high elastic modulus. Secondly, we are also using constant strain triangular elements as opposed to earlier studies which used quadrilateral elements. In addition, here as well as in the existing studies, the traction profile is plotted at gauss points which are bound to be different owing to a different discretization.

We also notice a slightly oscillatory behavior in the traction profile in both normal and tangential fields. As discussed earlier, the considered problem represents a highly pathological case given the presence of both large heterogeneities and arbitrarily small elements. Under those circumstances, these minor oscillations do not represent a significant drawback. In fact as demonstrated by the convergence profile of the Newton-Raphson algorithm, these oscillations are too small to have any considerable impact on the convergence behavior and accuracy. Furthermore, as discussed in Sanders et al. (2009), constant strain triangular elements are also prone to shear-locking and the use of a higher-order element (bilinear quads, for instance) might significantly improve the performance.

The advantages of the proposed approach are further illustrated when we look at the traction profile recovered by the penalty method (see Figure 5.9). The traction field in both normal as well as tangential directions oscillates wildly rendering it unusable. The unstable traction profile explains why it is hard to choose a combination of parameters  $\alpha_N$  and  $\alpha_\tau$  that leads to convergence. In conclusion, the shortcomings of the penalty method, from an analyst’s perspective, are twofold - not only is it

hard to choose the penalty parameters ( $\alpha_N$  and  $\alpha_\tau$ ) that would lead to convergence, but also that the choice that yields convergence is not guaranteed to yield a stable traction profile.

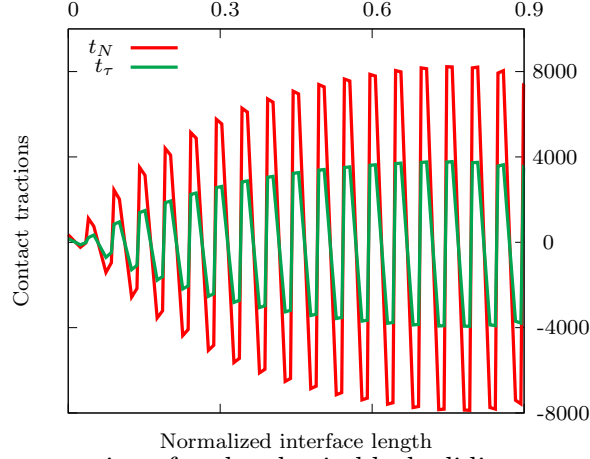


FIGURE 5.9 – Contact tractions for the elastic block sliding on a rigid surface using penalty method with penalty parameters  $\alpha_N = 1 \times 10^9$  and  $\alpha_\tau = 8 \times 10^5$ .

### *Mesh convergence study*

As a next example, we conduct the mesh convergence study performed in Liu and Borja (2008) to better understand the numerical convergence of the method as the mesh is refined. As described in Liu and Borja (2008), we consider a unit square domain  $\Omega = (0, 1) \times (0, 1)$  with a material interface  $\Gamma_* = \{\mathbf{x} : \psi(\mathbf{x}) = 0\}$ . The level set of the interface is given as  $\psi(\mathbf{x}) = y - 0.5$ . The interface divides the domain  $\Omega$  into component sub-domains  $\Omega^1$  and  $\Omega^2$ . The material properties in both  $\Omega^1$  and  $\Omega^2$  are considered identical such that  $E^1 = E^2 = 10$  GPa and  $\nu^1 = \nu^2 = 0.3$ . Further, the frictional coefficient for the interface is given as  $\mu = 0.1$ . The bottom surface is constrained to move in both directions while a fixed displacement of  $u_x = 0.05$  (m) and  $u_y = 0.09x - 0.01$  (m) is applied on the top surface. The geometry and boundary conditions are described in Figure 5.10. We now consider four finite element discretizations with 222, 1250, 5202 and 20402 triangular elements respectively.

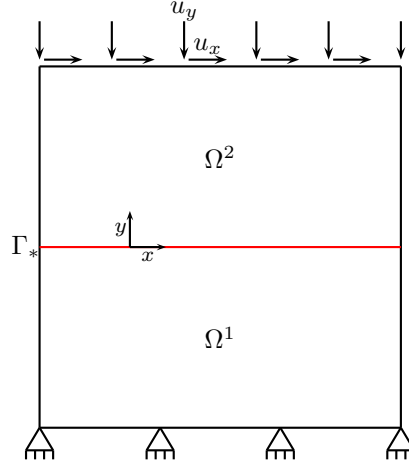


FIGURE 5.10 – A unit square domain loaded in compression and shear at the top surface and constrained at the bottom surface. Frictional sliding is allowed at the material interface  $\Gamma_*$ .

We first examine the convergence of normal tractions and tangential slip as the mesh is refined. Figure 5.11(a) plots the normal traction profile for all the four meshes using weighted Nitsche’s method. The results are in close agreement to those presented in Liu and Borja (2008). As noticed by them, the normal tractions are nearly insensitive to mesh refinement. We do notice slight oscillations in the normal traction profile for the coarse mesh. However, unlike their study, we have not smoothed the obtained tractions through a domain integral type postprocessing technique. We plot the tractions obtained at the gauss points through direct evaluation. The tangential slips reported in our study (Figure 5.12(a)) are slightly smaller than those reported by Liu and Borja (2008).

To investigate this further, since they used the penalty method in their study, we plot the normal tractions and tangential slips on using the penalty method as well in Figure 5.11(b) and 5.12(b). However, we first performed a parameter study on the penalty parameter to remove any penalty sensitivity in the results. For this parameter study we concentrated on the convergence of tangential slip and considered the results to be free of penalty sensitivity when the tangential slip was

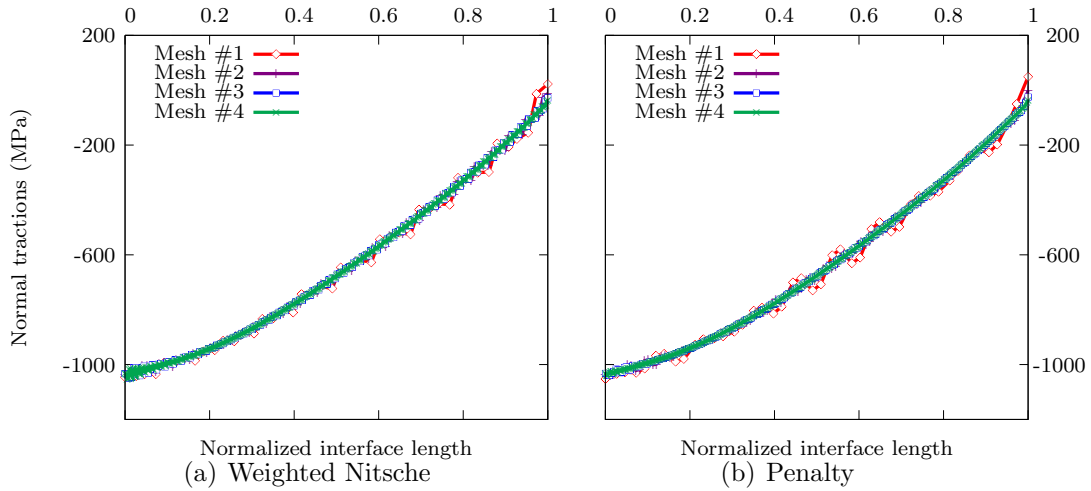


FIGURE 5.11 – Convergence of normal tractions with mesh refinement for weighted Nitsche’s method (left) and penalty method (right). For the penalty method, the parameter used is  $\alpha_N = \alpha_\tau = 5 \times 10^{-12}$  and was obtained from a parameter study.

insensitive to the increase in penalty parameter. This gave us a penalty parameter of  $\alpha_N = \alpha_\tau = 5 \times 10^{12}$  for all four meshes. A smaller parameter yields a larger slip which is in closer agreement with Liu and Borja (2008), however this serves to further illustrate why it is important to eliminate free-parameters within a method. As seen from Figures 5.11(a)-(b), and Figures 5.12(a)-(b), on removing the penalty sensitivity for the penalty method, the results are almost identical for both methods.

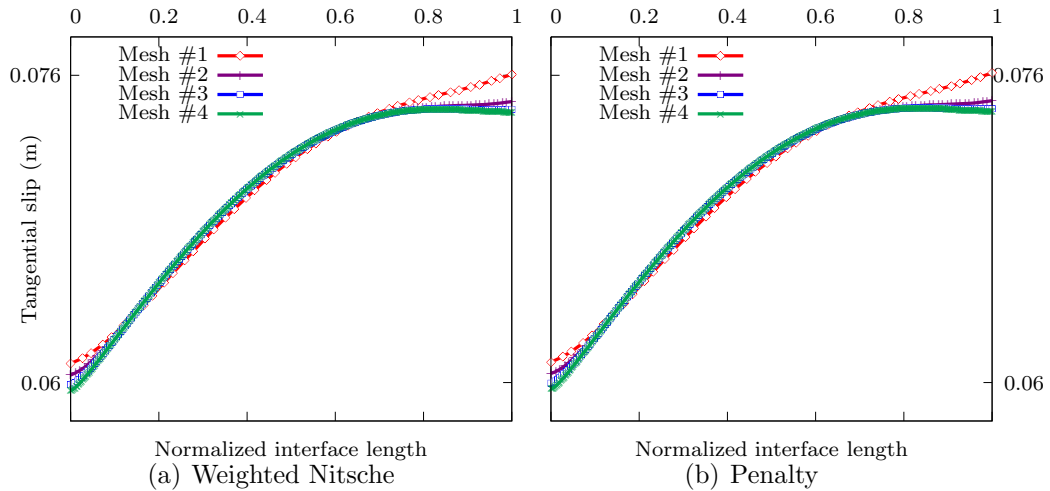


FIGURE 5.12 – Convergence of tangential slip with mesh refinement for weighted Nitsche’s method (left) and penalty method (right). For the penalty method, the parameter used is  $\alpha_N = \alpha_\tau = 5 \times 10^{-12}$  and was obtained from a parameter study.

The advantages of a variationally consistent approach however come forth on more closely inspecting the non-interpenetration constraint. To do so, we plot the non-interpenetration constraint at the interface for the weighted Nitsche’s method as well as the penalty method in Figure 5.13(a). For the penalty method, we first use a parameter that we obtained from the sensitivity study *i.e.*  $\alpha_N = \alpha_\tau = 5 \times 10^{12}$ . It is easy to see that this value of the penalty parameter results in a much larger interpenetration than weighted Nitsche’s method. However, since it is possible to tune the penalty parameter, we choose a value which enforces the non-interpenetration constraint with nearly identical accuracy to Nitsche’s method *viz.*  $\alpha_N = \alpha_\tau = 1 \times 10^{16}$ . Now, we plot the normal tractions for both these choices of penalty parameters as well as weighted Nitsche’s method in Figure 5.13(b). The traction profile for the weighted Nitsche’s method overlaps with that of the penalty method with a parameter of  $\alpha_N = \alpha_\tau = 5 \times 10^{12}$  so that the curves are nearly indistinguishable. However, the traction profile for the penalty method with  $\alpha_N = \alpha_\tau = 1 \times 10^{16}$  results in highly oscillatory behavior. From these results, it is easy to appreciate that while we can tune the penalty parameter to better enforce the constraints, it results in a deteriorating traction profile. Unsurprisingly, as is often seen with mixed methods, the stronger the constraint is enforced, the larger the oscillations. A smart choice for the method parameters in the weighted Nitsche approach, as prescribed here, strikes the right balance between obtaining an accurate enforcement of constraints while not compromising the stability of the interfacial fields.

#### *Crack face contact*

As a final example, we examine frictional sliding across a crack surface. This problem was first investigated by Dolbow et al. (2001) and then by Liu and Borja (2008). The problem involves a pre-cracked square elastic domain subjected to compressive loading. The crack faces are allowed to slide across each other following Coulomb’s



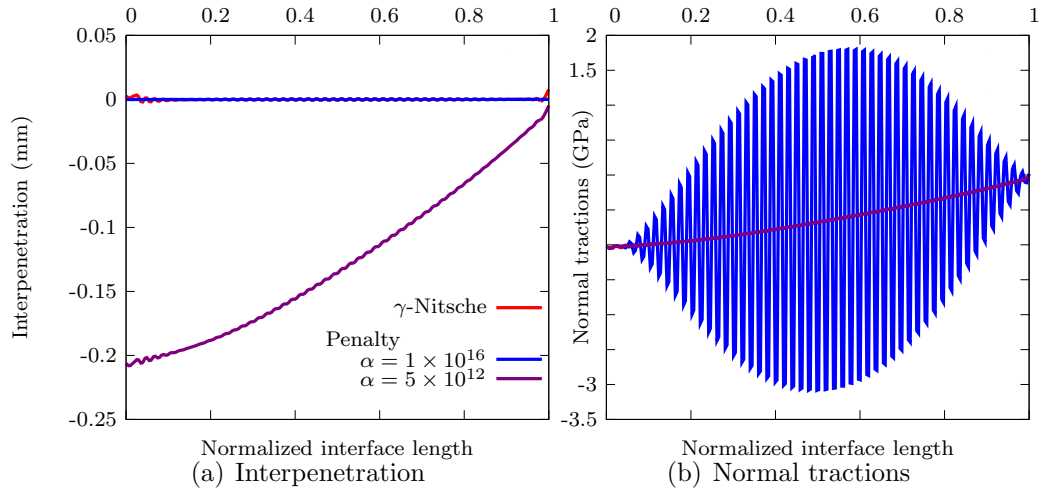


FIGURE 5.13 – Comparison of the relative performance of the weighted Nitsche and penalty methods in enforcing the non-interpenetration constraint (left) and recovery of normal tractions (right). The results for the penalty method are plotted for two parameters  $\alpha_N = \alpha_\tau = 5 \times 10^{12}$  (in violet) and  $\alpha_N = \alpha_\tau = 1 \times 10^{16}$  (in blue). Traction profile for weighted Nitsche (in red) and penalty method with  $\alpha_N = \alpha_\tau = 5 \times 10^{12}$  (in violet) are nearly indistinguishable.

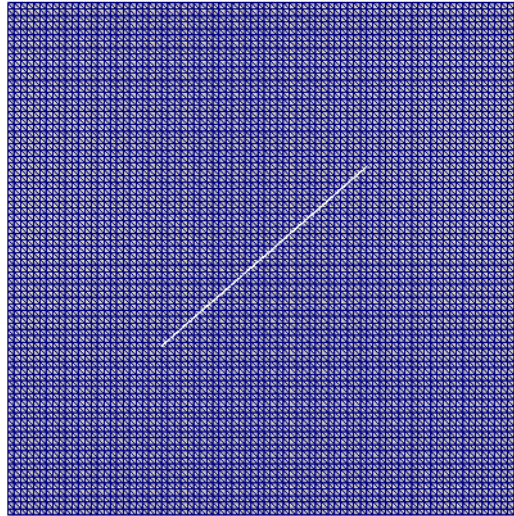
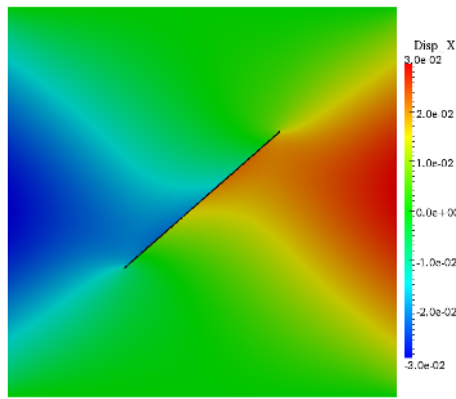
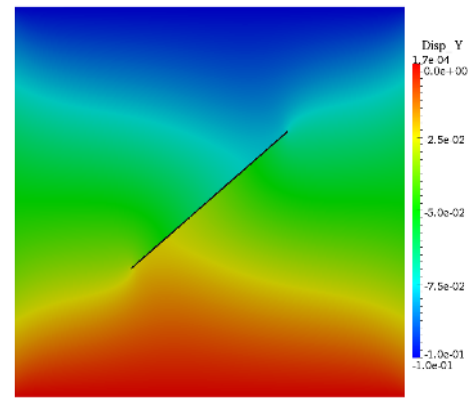


FIGURE 5.14 – Finite element mesh for the problem simulating crack face frictional contact. The domain is a unit square  $(0, 1) \times (0, 1)$  with an inclined crack extending from the coordinates  $(0.3, 0.33)$  to  $(0.7, 0.68)$ .

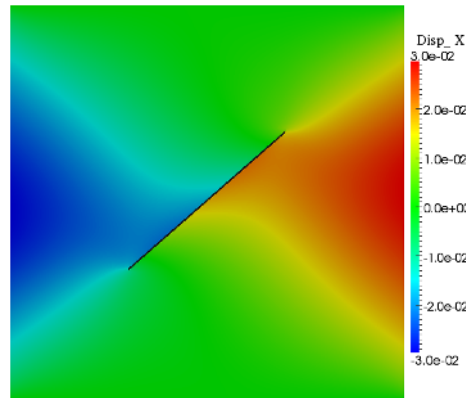
law but the crack tips are not allowed to advance. Here, we consider a unit square domain with a crack extending from the coordinates  $(0.3, 0.33)$  to  $(0.7, 0.68)$ . The material parameters are chosen as  $E = 10$  GPa and  $\nu = 0.3$ . The Coulomb's frictional coefficient on the crack surface is assumed to be  $\mu = 0.1$ . The crack is loaded compressively by applying a uniform displacement of  $u_y = 0.1$  on the top surface. The bottom surface is considered to be fixed. The computations are performed on an  $80 \times 80$  mesh as shown in Figure 5.14. Finally, the tolerance for Newton-Raphson iterations is set to  $1 \times 10^{-12}$  in the energy norm.



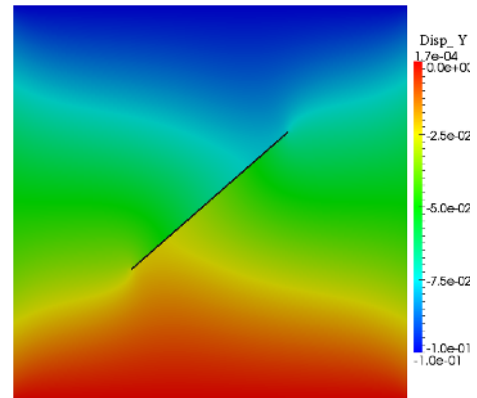
(a) x-displacement on using weighted Nitsche's method



(b) y-displacement on using weighted Nitsche's method



(c) y-displacement on using Penalty method with  $\alpha_N = \alpha_\tau = 1.0 \times 10^{13}$



(d) y-displacement on using Penalty method with  $\alpha_N = \alpha_\tau = 1.0 \times 10^{13}$

FIGURE 5.15 – x and y displacement contours obtained using weighted Nitsche's method (top) and Penalty method (bottom) for the crack face frictional contact problem.

The horizontal and vertical displacement contours obtained using weighted Nitsche's

method are plotted in Figures 5.15(a)-(b). For comparison, we also plot the displacement contours using the penalty method with the same penalty parameters as Liu and Borja (2008), *i.e.*  $\alpha_N = \alpha_\tau = 1.0 \times 10^{13}$ . The plots are again in excellent agreement with earlier studies. The proposed method converged in three Newton-Raphson iterations as opposed to the LATIN, iterative strategy employed earlier in Dolbow et al. (2001) which required in excess of 100 iterations to converge. Additionally, unlike the LATIN method used by Dolbow et al. (2001) and the penalty method of Liu and Borja (2008) we do not have any tunable regularization parameters in our method which could affect the accuracy of our constraint enforcement as well as the rate of Newton-Raphson convergence.

These results are very encouraging and in conjunction with our earlier work in Chapter 3, make the proposed approach a desirable candidate in tackling high-fidelity quasi-static and dynamic crack propagation studies (see, for example, the work of Combescure and co-workers Combescure et al. (2005); Elguedj et al. (2007); Grégoire et al. (2007, 2009)).

## 5.5 Conclusion

We proposed a Nitsche stabilized approach to model frictional contact problems. The proposed formulation extends Nitsche's method to handle non-linear constraints and allows us to incorporate frictional behavior at the interface. The frictional problems described here are naturally non-symmetric and therefore we propose a non-symmetric variational form. As with linear problems, the method parameter arising in the variational formulation is provided by analytical estimates, lending the formulation additional robustness. The frictional constraint is incorporated directly in the variational form as a Neumann condition. This also allows us to incorporate various forms of frictional constraints in the same framework, as demonstrated in the numerical studies where we incorporate both perfectly plastic interfaces and

Coulombian frictional models.

We performed several numerical studies to demonstrate the advantages of the proposed approach for frictional sliding problems over the more commonly used penalty methods. In these studies, we also highlighted the lack of robustness of the more classical form of Nitsche's method. The advantages of the proposed method over a penalty approach stem from its variational consistency. This allows the method to enforce the constraints much more accurately while using a small stabilization parameter - akin to an augmented Lagrange multiplier method with small penalties. However, this is a primal approach and therefore neither adds additional degrees of freedom nor requires an outer augmentation loop. Unlike penalty methods, where accuracy and convergence both strongly depend on the regularization parameter, the proposed method is free of tunable parameters. Furthermore, as seen from the numerical examples, the proposed choice for the method parameter naturally strikes the right balance between enforcing the non-interpenetrability constraint and recovering the interfacial tractions - a balance hard to achieve with traditional mixed methods.

Going forward, in the next Chapter, we extend the proposed approach to handle multiple locally intersecting interfaces.

# A Nitsche Stabilized Embedded Finite Element Method for Junctions

## 6.1 Introduction

In this thesis, so far, while we have considered complex geometrical shapes embedded within a bulk mesh, they have all represented a single interface – a boundary separating two domains. In this Chapter, we relax this assumption and consider problems with multiple intersecting interfaces within a region. In particular, we extend the weighted Nitsche’s approach to consider perfectly-tied and perfectly plastic intersecting interfaces. These intersections could arise either inside a computational domain - where two internal interfaces intersect; or on the boundary of the computational domain - where an internal interface intersects with the external boundary. We propose a variational treatment of both the interfacial kinematics and the external Dirichlet constraints within Nitsche’s framework. In light of these intersections, the previously provided definitions of the method parameters might no longer be sufficient. We will now need to account for these intersections explicitly in the local coercivity analysis to ensure the method parameters are chosen correctly.

The motivation for investigating these intersecting interfaces comes from some of the applications (in computational micromechanics and geo-sciences) discussed in Chapter 1. We have also discussed some of the computational challenges associated with traditional finite element methods for such problems there. Most of these challenges arise because interfaces cannot be represented independently from the underlying finite element mesh. Not surprisingly then, there have been many studies investigating the use of enriched approximations (such as X-FEM) which can accomplish this decoupling for these topologically complex problems.

Moës et al. (2003) first proposed the use of X-FEM for the homogenization of periodic basic cells in two-phase composites. Since then, the combined X-FEM/level-set approach has been extended to model interfacial failure in composites (see Hettich and Ramm, 2006; Hettich et al., 2008), for obtaining homogenized properties for heterogeneous structures (see Legrain et al., 2011), and for obtaining effective elastic moduli in nanocomposites (Yvonnet et al., 2008). For geo-mechanical problems, X-FEM has been employed to model faulting (see the work of Liu and Borja, 2009, 2010a), earthquake ruptures (Coon et al., 2011) and subsurface flow (Huang et al., 2011).

There have also been several studies investigating the performance of X-FEM in the presence of junctions. Daux et al. (2000) first studied branched discontinuities in the X-FEM. To treat these, they proposed using a Heaviside function for every intersecting interface and also an additional junction function. Sukumar et al. (2003) modeled brittle fracture in polycrystalline materials using an X-FEM approach. Further, Simone et al. (2006) proposed a G-FEM approach for modeling grain boundary sliding in polycrystalline materials. Robbins and Voth (2007) used X-FEM with Heaviside enrichments to model the shock response of polycrystalline materials. For problems with intersecting material interfaces, Aragón et al. (2010) presented novel enrichment functions that preserve optimal convergence rates for

quadratic approximations. More recently, Shabir et al. (2011) investigated the role of cohesive properties on intergranular crack propagation in brittle polycrystalline materials using the G-FEM approach of Simone et al. (2006).

However, to our knowledge, development of numerical approaches that could robustly enforce stiff kinematic constraints, such as perfect and sliding contact, over intersecting embedded interfaces has received little attention. The traditionally used approaches suffer from shortcomings well-known in mixed finite element methods, *viz.* an unstable traction field. Only recently, Siavelis et al. (2012) presented a stable Lagrange multiplier approach to enforce large sliding constraints on branched discontinuities. This method relies on the vital vertex algorithm of Béchet et al. (2009) to construct an *inf-sup* stable space for the Lagrange multipliers.

By contrast, we extend the weighted Nitsche approach, presented in Chapter 5, to model small-deformation frictional sliding on intersecting embedded interfaces. We use this method in conjunction with the overlapping element formulation of Hansbo and Hansbo (2004). As discussed earlier, the method is advantageous because it does not introduce an additional multiplier field and is parameter free. Also, the proposed approach provides a natural way to treat elements with multiple discontinuities without giving rise to linear dependencies as opposed to the method of Simone et al. (2006) and is much easier to implement compared to the junction function approach of Daux et al. (2000) and Siavelis et al. (2012).

The Chapter is organized as follows. We begin by defining the model problem and the variational form in the next section. In Section 6.3, we discuss the spatial discretization and provide a local estimate for the stabilization parameter in the presence of intersecting interfaces. In Section 6.4, we demonstrate the robustness of the proposed method on several benchmark examples for both perfectly-tied and perfectly-plastic sliding embedded interfaces with junctions. Finally, in Section 6.5, we provide a summary and concluding remarks. In Appendix A, we provide the

variational form for frictional sliding problems and also describe the algorithmic details regarding the assembly of local tangent matrix for elements with multiple interfaces.

## 6.2 Model problem and variational formulation

As a model problem, we consider Poisson's equation. This choice enables a better presentation of the local coercivity analysis, required to bound the stabilization parameter, by avoiding notational complexities. However, for completeness, we present the governing equations, the variational statement, and algorithmic details for frictional sliding problems in Appendix A.

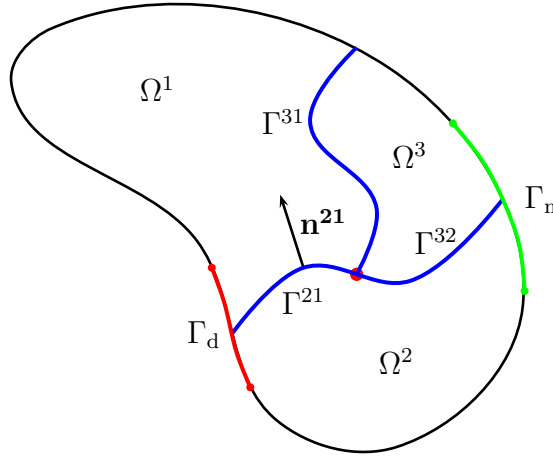


FIGURE 6.1 – Schematic of the problem domain. Domains  $\Omega^1$ ,  $\Omega^2$  and  $\Omega^3$  and the Dirichlet and Neumann boundaries  $\Gamma_d$ ,  $\Gamma_n$  are as shown. Each pair of domains  $\Omega^a$  and  $\Omega^b$  are separated by a shared internal boundary  $\Gamma^{ab}$ . The normal to the boundary segment  $\Gamma^{ab}$  is given by  $\mathbf{n}^{ab}$  and points outwards as shown for the interface  $\Gamma^{21}$ .

The governing equations in each domain  $\Omega^m$ , for  $m=1,2,3$  are given as:

$$\begin{aligned} \nabla \cdot \kappa^m \nabla u^m &= 0 & \text{in } \Omega^m, \\ u^m &= u_d & \text{on } \Gamma_d, \\ \kappa^m \nabla u^m \cdot \mathbf{n} &= 0 & \text{on } \Gamma_n. \end{aligned} \tag{6.1}$$



These are then coupled together at every interface  $\Gamma^{\text{ab}}$ , for  $\text{ab} = \{21, 31, 32\}$ , by the following conditions:

$$\begin{aligned} \llbracket \kappa \nabla u \rrbracket^{\text{ab}} \cdot \mathbf{n}^{\text{ab}} &= 0 \quad \text{on } \Gamma^{\text{ab}}, \\ \llbracket u \rrbracket^{\text{ab}} &= 0 \quad \text{on } \Gamma^{\text{ab}}. \end{aligned} \tag{6.2}$$

where by  $\llbracket (\cdot) \rrbracket^{\text{ab}}$ , we denote the jump  $(\cdot)^{\text{b}} - (\cdot)^{\text{a}}$  in a quantity  $(\cdot)$ . The interfaces,  $\Gamma^{\text{ab}}$  are allowed to intersect with one another and act as internal boundaries between the domains. The normal to each of these internal boundaries,  $\Gamma^{\text{ab}}$  is denoted by  $\mathbf{n}^{\text{ab}}$  and is pointing from  $\Omega^{\text{a}}$  to  $\Omega^{\text{b}}$ . Further, the Dirichlet and Neumann boundaries are denoted by  $\Gamma_{\text{d}}$  and  $\Gamma_{\text{n}}$  and are allowed to intersect with the internal boundaries. Figure 6.1 describes the problem set-up and lends concreteness to the definitions of the introduced quantities.

Now, neither the interfacial kinematics nor the specified Dirichlet constraint can be handled through standard approaches. Consequently, we propose a variational treatment for both using Nitsche's method.

### 6.2.1 Weak form

A weak statement of the model problem (6.1) – (6.2) is presented below. For sufficiently smooth functional spaces  $\mathcal{U}$  and  $\mathcal{V}$ , find  $u \in \mathcal{U}$ , such that for all  $w \in \mathcal{V}$  the following statement holds:

$$a(w, u) = l(u). \tag{6.3}$$

Here,

$$\begin{aligned} a(w, u) &= \sum_{\text{m}} \int_{\Omega^{\text{m}}} \nabla w^{\text{m}} \cdot \kappa^{\text{m}} \nabla u^{\text{m}} \, \text{d}\Omega - \sum_{\text{ab}} \int_{\Gamma^{\text{ab}}} \llbracket w \rrbracket^{\text{ab}} \langle \kappa \nabla u \rangle_{\gamma_{\text{ab}}} \cdot \mathbf{n}^{\text{ab}} \, \text{d}\Gamma \\ &\quad - \sum_{\text{ab}} \int_{\Gamma^{\text{ab}}} \llbracket u \rrbracket^{\text{ab}} \langle \kappa \nabla w \rangle_{\gamma_{\text{ab}}} \cdot \mathbf{n}^{\text{ab}} \, \text{d}\Gamma + \sum_{\text{ab}} \int_{\Gamma^{\text{ab}}} \llbracket w \rrbracket^{\text{ab}} \alpha^{\text{ab}} \llbracket u \rrbracket^{\text{ab}} \, \text{d}\Gamma \\ &\quad - \sum_{\text{r}} \int_{\Gamma_{\text{d}}/\Omega^{\text{r}}} w^{\text{r}} \kappa^{\text{r}} \nabla u^{\text{r}} \cdot \mathbf{n}_{\text{d}} \, \text{d}\Gamma - \sum_{\text{r}} \int_{\Gamma_{\text{d}}/\Omega^{\text{r}}} u^{\text{r}} \kappa^{\text{r}} \nabla w^{\text{r}} \cdot \mathbf{n}_{\text{d}} \, \text{d}\Gamma \\ &\quad + \sum_{\text{r}} \int_{\Gamma_{\text{d}}/\Omega^{\text{r}}} w^{\text{r}} \alpha^{\text{r}} u^{\text{r}} \, \text{d}\Gamma, \end{aligned}$$

$$l(u) = \sum_m \int_{\Omega^m} w^m f^m \, d\Omega - \sum_r \int_{\Gamma_{d/\Omega^r}} u_d \kappa^r \nabla w^r \cdot \mathbf{n}_d \, d\Gamma + \sum_r \int_{\Gamma_{d/\Omega^r}} w^r \alpha^r u_d \, d\Gamma,$$

where  $\sum_m$  denotes a sum over all the bulk domains and by  $\sum_{ab}$ , we denote the sum over all internal boundaries  $\Gamma^{ab}$ . Further, if we define the boundary of a domain  $\Omega^m$  by  $\partial\Omega^m$  then  $\Gamma_{d/\Omega^m}$  denotes the part of Dirichlet boundary that belongs to  $\partial\Omega^m$ . By  $\sum_r$ , we denote the sum over all domains  $\Omega^r$  such that  $\Gamma_{d/\Omega^r} \neq \emptyset$ .

The outward pointing normal at the Dirichlet boundary is denoted by  $\mathbf{n}_d$ . The stabilization parameter for each interface  $\Gamma^{ab}$  is denoted by  $\alpha^{ab}$  while that for the Dirichlet boundary segment  $\Gamma_{d/\Omega^r}$  is denoted by  $\alpha^r$ . Finally, the weighted average flux at the interface is given by  $\langle \kappa \nabla u \rangle_{\gamma_{ab}} \cdot \mathbf{n}^{ab} = (\gamma_{ab}^1 \kappa_a \nabla u_a + \gamma_{ab}^2 \kappa_b \nabla u_b) \cdot \mathbf{n}^{ab}$ , with  $\gamma_{ab}^1 + \gamma_{ab}^2 = 1$ .

### 6.3 Discretization

We construct a mesh partition of the domains  $\Omega^1$ ,  $\Omega^2$  and  $\Omega^3$  into a set of non-overlapping triangles. The underlying finite element mesh is independent of the interface geometry (see Figure 6.2 for a schematic). The approximation and the weighting functions are then given by:

$$\mathbf{u}^h = \sum_m \sum_{i \in I^m} H^m \mathbf{N}_i \mathbf{u}_i, \quad \mathbf{w}^h = \sum_m \sum_{i \in I^m} H^m \mathbf{N}_i \mathbf{w}_i, \quad (6.4)$$

where,  $I^m$  is the set of all nodes whose supports overlap the domain  $\Omega^m$  and  $H^m$  is the characteristic function given by

$$H^m(\mathbf{x}) = \begin{cases} 1 & \text{if } \mathbf{x} \in \Omega^m, \\ 0 & \text{otherwise.} \end{cases} \quad (6.5)$$

Although equivalent to the X-FEM with Heaviside enrichment, the Hansbo approach described above is attractive from an implementational stand-point. Unlike

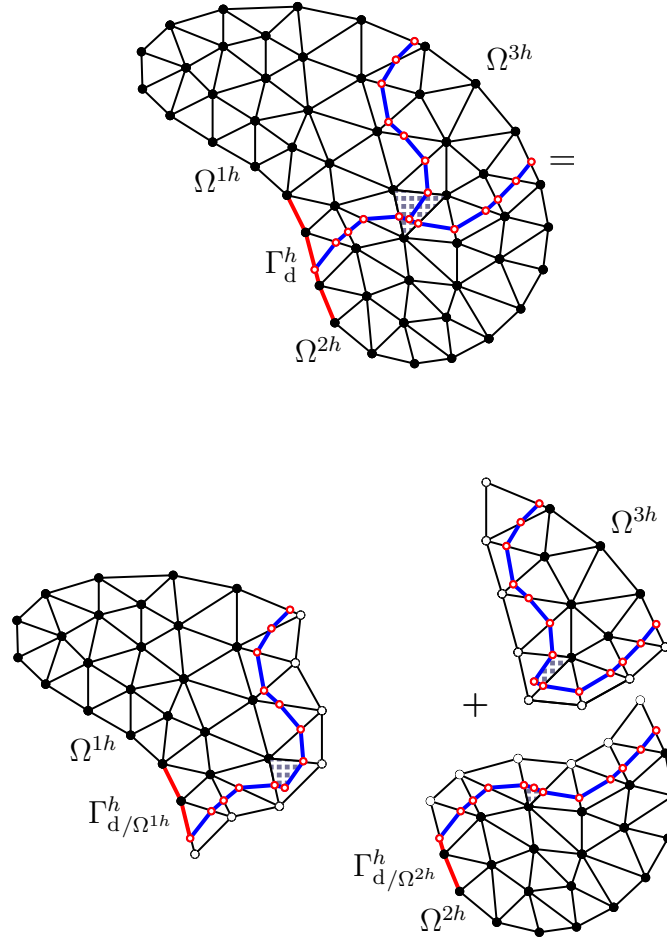


FIGURE 6.2 – Spatial discretization for the embedded interface formulation. The black circles are the physical nodes corresponding to the background mesh and the hollow circles are the phantom nodes. The red hollow circles represent the discretization of the embedded interface within an element.

the approach of Daux et al. (2000) and Siavelis et al. (2012), the Hansbo approach does not require a junction function and is also directly extendable to any number of intersecting interfaces. Moreover, unlike the approach of Simone et al. (2006) where particular care needs to be exercised for elements with junctions to avoid linear dependencies, the Hansbo approach results in a linearly independent basis by construction (see Robbins and Voth (2007) for a more detailed discussion). Finally, it is also worth mentioning that the Hansbo approach decouples the interfacial kine-

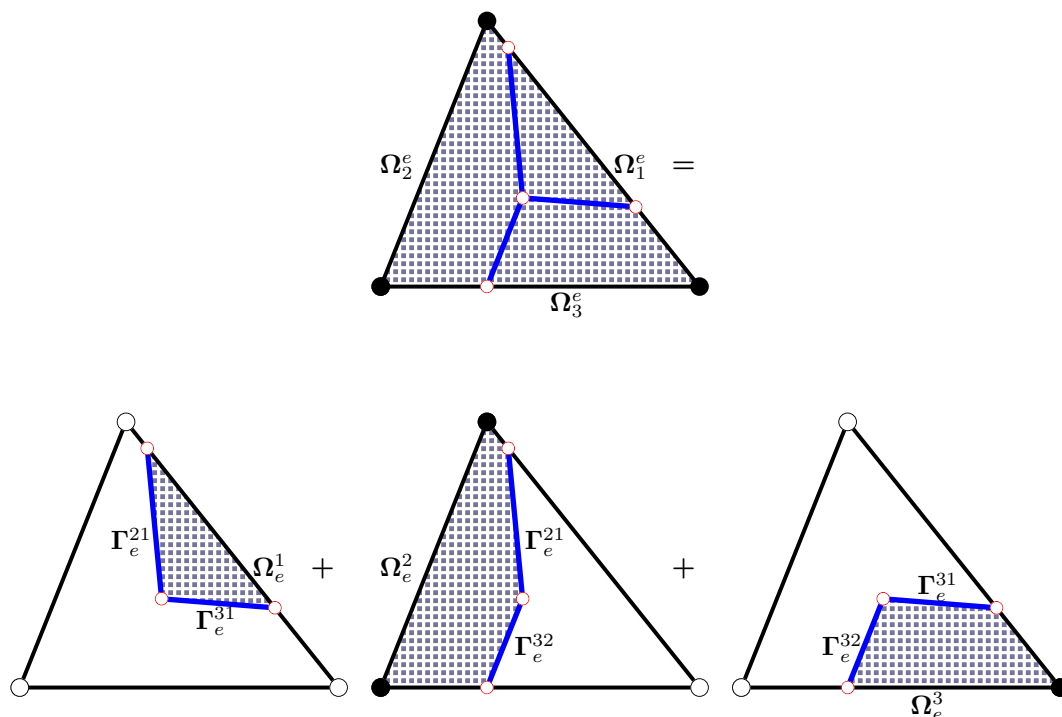


FIGURE 6.3 – Decomposition of a parent element with three local domains into corresponding children. The shaded region in the element represents the physical volume of the element. The black circles represent the physical nodes while the black hollow circles represent the phantom nodes. The red circles represent the discretization of the embedded surface.

matics from the choice of the enrichment functions. The interfacial kinematics are enforced through variational means in a discontinuous Galerkin sense. This provides the capability to model a wide range of interfacial kinematics (for instance, perfectly tied - material interfaces, tangential sliding and decohesion) within the same framework.

From an implementational perspective, the only noticeable difference from the case we discussed in Chapter 5 arises for elements with multiple interfaces. Within the described framework, a background element is replaced by as many duplicate elements as its original constituent domains. See for example, the shaded triangular element containing the junction in Figure 6.2. Further, for elements with junctions, it

is also likely that a background element edge might be intersected more than once and result in a child element that has no physical nodes (see Figure 6.3). Such elements can also be handled naturally within the described phantom node framework and do not warrant any additional consideration.

### 6.3.1 Stability

Stability is established by choosing stabilization parameters that establish coercivity of the discrete bilinear form at an element level. However, from the definition of the bilinear form (6.3), the following scenarios are likely to yield a distinct local bilinear form and consequently distinct requirements for the stabilization parameter:

- (a) Part of the Dirichlet boundary belongs to the element boundary:  $\Gamma_{d/\Omega_e} \neq \emptyset$ .
- (b) Element contains a single interface.
- (c) Element contains multiple interfaces.
- (d) Element boundary contains part of the Dirichlet boundary AND the element contains a single interface.
- (e) Element boundary contains part of the Dirichlet boundary AND the element contains multiple interfaces.

Case (a) has been previously addressed in Dolbow and Harari (2009) and case (b) has been discussed in Chapter 4. We now conduct the stability analysis for case (c) here. In particular, we focus on the case when the element contains multiple intersecting interfaces. For an element with multiple non-intersecting interfaces, the analysis yields identical results. For cases (d) and (e), for conciseness, we omit the analysis but directly provide the expression for the stabilization parameter.

We begin by recalling the definitions of “energy” and  $L_2$  norms:

$$\|\nabla u^m\|_{\Omega_e^m, \kappa^m}^2 = \int_{\Omega_e^m} \nabla u^m \cdot \kappa^m \nabla u^m \, d\Omega; \quad \|u^m\|_{\Gamma_e}^2 = \int_{\Gamma_e} u^m \cdot u^m \, d\Gamma. \quad (6.6)$$

We also recall that, for every interface  $\Gamma_e^{\text{ab}}$ , there exists a configuration dependent constant  $C_{\text{ab}}$  such that the following generalized inverse inequality holds:

$$\|\langle \kappa \nabla u \rangle_{\gamma_{\text{ab}}^e} \cdot \mathbf{n}^{\text{ab}}\|_{\Gamma_e^{\text{ab}}} \leq C_{\text{ab}} \|\nabla u\|_{\Omega_e^{\text{ab}, \kappa}}, \quad (6.7)$$

where  $\|\nabla u\|_{\Omega_e^{\text{ab}, \kappa}}^2 = \|\nabla u^{\text{a}}\|_{\Omega_e^{\text{a}, \kappa^{\text{a}}}}^2 + \|\nabla u^{\text{b}}\|_{\Omega_e^{\text{b}, \kappa^{\text{b}}}}^2$ .

*Element contains multiple intersecting boundaries*

Consider an element with multiple intersecting interfaces as shown in Figure 6.3. For such an element, from (6.3), the local bilinear form is:

$$\begin{aligned} a(u, u)^e &= \sum_{m=1}^3 \|\nabla u^m\|_{\Omega_e^m, \kappa^m}^2 - 2 \int_{\Gamma_e^{21}} [[u]]^{21} \langle \kappa \nabla u \rangle_{\gamma_{21}^e} \cdot \mathbf{n}^{21} \, d\Gamma \\ &\quad - 2 \int_{\Gamma_e^{31}} [[u]]^{31} \langle \kappa \nabla u \rangle_{\gamma_{31}^e} \cdot \mathbf{n}^{31} \, d\Gamma - 2 \int_{\Gamma_e^{32}} [[u]]^{32} \langle \kappa \nabla u \rangle_{\gamma_{32}^e} \cdot \mathbf{n}^{32} \, d\Gamma \\ &\quad + \alpha_e^{21} \|[[u]]^{21}\|_{\Gamma_e^{21}}^2 + \alpha_e^{31} \|[[u]]^{31}\|_{\Gamma_e^{31}}^2 + \alpha_e^{32} \|[[u]]^{32}\|_{\Gamma_e^{32}}^2. \end{aligned} \quad (6.8)$$

We establish a lower bound on  $\alpha_e^{21}$ ,  $\alpha_e^{31}$  and  $\alpha_e^{32}$  by ensuring coercivity of the local bilinear form (6.8). Now, the following inequalities can be derived:

$$\begin{aligned} a(u, u)^e &\geq \sum_{m=1}^3 \|\nabla u^m\|_{\Omega_e^m, \kappa^m}^2 - 2 \|[[u]]^{21}\|_{\Gamma_e^{21}} \|\langle \kappa \nabla u \rangle_{\gamma_{21}^e} \cdot \mathbf{n}^{21}\|_{\Gamma_e^{21}} \\ &\quad - 2 \|[[u]]^{31}\|_{\Gamma_e^{31}} \|\langle \kappa \nabla u \rangle_{\gamma_{31}^e} \cdot \mathbf{n}^{31}\|_{\Gamma_e^{31}} - 2 \|[[u]]^{32}\|_{\Gamma_e^{32}} \|\langle \kappa \nabla u \rangle_{\gamma_{32}^e} \cdot \mathbf{n}^{32}\|_{\Gamma_e^{32}} \\ &\quad + \alpha_e^{21} \|[[u]]^{21}\|_{\Gamma_e^{21}}^2 + \alpha_e^{31} \|[[u]]^{31}\|_{\Gamma_e^{31}}^2 + \alpha_e^{32} \|[[u]]^{32}\|_{\Gamma_e^{32}}^2, \end{aligned} \quad (6.9)$$

$$\begin{aligned}
&\geq \frac{1}{2} \left( \|\nabla u\|_{\Omega_e^{21,\kappa}}^2 - 4\| [u]^{21} \|_{\Gamma_e^{21}} \|\langle \kappa \nabla u \rangle_{\gamma_{21}^e} \cdot \mathbf{n}^{21}\|_{\Gamma_e^{21}} \right) \\
&+ \frac{1}{2} \left( \|\nabla u\|_{\Omega_e^{31,\kappa}}^2 - 4\| [u]^{31} \|_{\Gamma_e^{31}} \|\langle \kappa \nabla u \rangle_{\gamma_{31}^e} \cdot \mathbf{n}^{31}\|_{\Gamma_e^{31}} \right) \\
&+ \frac{1}{2} \left( \|\nabla u\|_{\Omega_e^{32,\kappa}}^2 - 4\| [u]^{32} \|_{\Gamma_e^{32}} \|\langle \kappa \nabla u \rangle_{\gamma_{32}^e} \cdot \mathbf{n}^{32}\|_{\Gamma_e^{32}} \right) \\
&+ \alpha_e^{21} \| [u]^{21} \|_{\Gamma_e^{21}}^2 + \alpha_e^{31} \| [u]^{31} \|_{\Gamma_e^{31}}^2 + \alpha_e^{32} \| [u]^{32} \|_{\Gamma_e^{32}}^2, \tag{6.10}
\end{aligned}$$

$$\begin{aligned}
&\geq \frac{1}{2} \left( \|\nabla u\|_{\Omega_e^{21,\kappa}}^2 - 4\| [u]^{21} \|_{\Gamma_e^{21}} C_{21} \|\nabla u\|_{\Omega_e^{21,\kappa}} \right) + \alpha_e^{21} \| [u]^{21} \|_{\Gamma_e^{21}}^2 \\
&+ \frac{1}{2} \left( \|\nabla u\|_{\Omega_e^{31,\kappa}}^2 - 4\| [u]^{31} \|_{\Gamma_e^{31}} C_{31} \|\nabla u\|_{\Omega_e^{31,\kappa}} \right) + \alpha_e^{31} \| [u]^{31} \|_{\Gamma_e^{31}}^2 \\
&+ \frac{1}{2} \left( \|\nabla u\|_{\Omega_e^{32,\kappa}}^2 - 4\| [u]^{32} \|_{\Gamma_e^{32}} C_{32} \|\nabla u\|_{\Omega_e^{32,\kappa}} \right) + \alpha_e^{32} \| [u]^{32} \|_{\Gamma_e^{32}}^2, \tag{6.11}
\end{aligned}$$

$$\begin{aligned}
&\geq \frac{1}{2} (\|\nabla u\|_{\Omega_e^{21,\kappa}} - C_{21} [u]^{21})^2 + \frac{1}{2} (\|\nabla u\|_{\Omega_e^{31,\kappa}} - C_{31} [u]^{31})^2 + \frac{1}{2} (\|\nabla u\|_{\Omega_e^{32,\kappa}} - C_{32} [u]^{32})^2 \\
&+ (\alpha_e^{21} - 2C_{21}^2) \| [u]^{21} \|_{\Gamma_e^{21}}^2 + (\alpha_e^{31} - 2C_{31}^2) \| [u]^{31} \|_{\Gamma_e^{31}}^2 + (\alpha_e^{32} - 2C_{32}^2) \| [u]^{32} \|_{\Gamma_e^{32}}^2. \tag{6.12}
\end{aligned}$$

Equation (6.9) follows from using Cauchy-Schwarz inequality in equation (6.8). Equation (6.10) follows from using the following identity:

$$\|\nabla u\|_{\Omega_e^{21,\kappa}}^2 + \|\nabla u\|_{\Omega_e^{31,\kappa}}^2 + \|\nabla u\|_{\Omega_e^{32,\kappa}}^2 = 2 \sum_{m=1}^3 \|\nabla u^m\|_{\Omega_e^m,\kappa^m}^2, \tag{6.13}$$

while equations (6.11) and (6.12) follow on using the generalized inverse estimate (6.7) in (6.10). It is clear from (6.12) that the stability of the bilinear form can be ensured if  $\alpha_e^{ab} \geq 2C_{ab}^2$ , for  $ab = \{21, 31, 32\}$ .

A lower bound on the configuration dependent constant,  $C_{ab}$  is now provided as discussed in Chapter 4 and results in:

$$\alpha_e^{ab} = 2 \times \text{meas}(\Gamma_e^{ab}) \left( \frac{(\gamma_{ab}^{1e})^2 \kappa^b}{\text{meas}(\Omega_e^b)} + \frac{(\gamma_{ab}^{2e})^2 \kappa^a}{\text{meas}(\Omega_e^a)} \right), \tag{6.14}$$

where the weights are chosen to minimize the stabilization parameter:

$$\gamma_{ab}^{1e} = \frac{\text{meas}(\Omega_e^b)/\kappa^b}{\text{meas}(\Omega_e^b)/\kappa^b + \text{meas}(\Omega_e^a)/\kappa^a}; \quad (6.15)$$

$$\gamma_{ab}^{2e} = \frac{\text{meas}(\Omega_e^a)/\kappa^a}{\text{meas}(\Omega_e^b)/\kappa^b + \text{meas}(\Omega_e^a)/\kappa^a}, \quad (6.16)$$

for  $ab = \{21, 31, 32\}$  respectively.

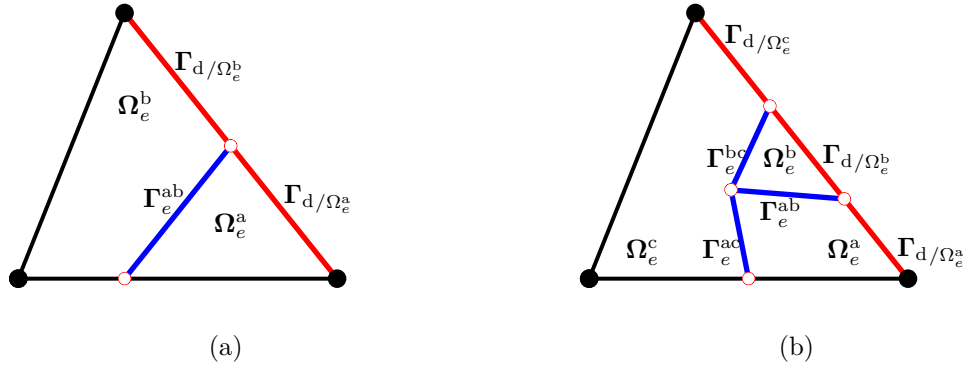


FIGURE 6.4 – Representative elements for the case when the element boundary contains part of the Dirichlet boundary and the element contains: a single embedded interface (left); multiple intersecting interfaces (right).

For cases (d) and (e), we obtain estimates for the stabilization parameters at both interfaces and the external Dirichlet boundaries by following a similar procedure as one described above. Consider an element with an embedded interface  $\Gamma_e^{ab}$  separating the regions  $\Omega_e^a$  and  $\Omega_e^b$ . Further, one of the element edges is also a part of the Dirichlet boundary as shown in Figure 6.4(a). For this case, the stabilization parameter at the interface,  $\Gamma_e^{ab}$  remains identical to one provided above by equation (6.14). The stabilization parameter for the Dirichlet boundary,  $\Gamma_{d/\Omega_e^m}$ , that is intersected by  $\Gamma_e^{ab}$  is given by the following expression:

$$\alpha_e^m = 2 \times \text{meas}(\Gamma_{d/\Omega_e^m}) \frac{\kappa^m}{\text{meas}(\Omega_e^m)} \quad \text{for } m = a, b. \quad (6.17)$$



For case (e), we can either have a scenario where an element edge that is a part of a Dirichlet boundary is intersected by one interface or intersected by more than one interface. For such elements, we choose an estimate based on the latter case as this results in a larger value. Figure 6.4(b) describes a representative element for case (e). The stabilization parameter at an interface  $\Gamma_e^{pq}$  and for the Dirichlet boundary segment  $\Gamma_{d/\Omega_e^m}$  is given by:

$$\begin{aligned}\alpha_e^{pq} &= 3 \times \text{meas}(\Gamma_e^{pq}) \left( \frac{(\gamma_{pq}^{1e})^2 \kappa^q}{\text{meas}(\Omega_e^q)} + \frac{(\gamma_{pq}^{2e})^2 \kappa^p}{\text{meas}(\Omega_e^p)} \right) \quad \text{for } pq = \{ab, ac, bc\}, \\ \alpha_e^m &= 3 \times \text{meas}(\Gamma_{d/\Omega_e^m}) \frac{\kappa^m}{\text{meas}(\Omega_e^m)} \quad \text{for } m = a, b, c.\end{aligned}\tag{6.18}$$

Finally, in concluding this section, we remark that for the case with intersecting interfaces, even with the proposed choice of weights, the expression for the stabilization parameter (6.14) might still yield a large value if the physical volume of two of its constituent domains is small. However, from the expression, we also notice that the expression is directly related to the length of the segment  $\text{meas}(\Gamma_e^{ab})$ . In a Voronoi tessellation, it is rather difficult to generate a configuration where this segment length is large and the physical volume of two constituent domains is small.

To further illustrate this point, in the next section, we present a sensitivity study where we show that the discrete system of equations is likely to remain well-conditioned regardless of the location of the junction within an element for the weighted Nitsche's method. Also, in all of the numerical examples presented in the next section, we do not have any control over the junction location since the Voronoi tessellation is generated from randomly seeded grain centers. However, we do not report unstable fluxes in any of these examples.

## 6.4 Numerical examples

We now validate the proposed method through various benchmark numerical studies in the presence of junctions. In order to accomplish this, we proceed by first examining the convergence behavior of the proposed method for linear problems. These convergence studies are conducted by measuring error in the standard  $L_2$  norm for the bulk variable and in the energy semi-norm for the gradient. We then investigate non-linear problems with perfectly plastic interfacial behavior. Again, to validate, at first we examine the performance of the method on numerical experiments that we conducted in Chapter 5. However, we now replicate those experiments such that the problem set-up has intersecting material interfaces and compare the obtained results with those previously presented in Chapter 5. Finally, to demonstrate the robustness of the proposed approach, we conduct numerical experiments on a polycrystalline geometry and qualitatively discuss the obtained results. Throughout the section, we refer to Nitsche's method with weights identically chosen equal to 0.5 as the classical Nitsche's method. Also, in the figure legends, we refer to the weighted Nitsche's method as  $\gamma$ -Nitsche's method.

We solve the non-linear equilibrium equations using the Newton-Raphson algorithm and as a convergence criterion, we specify a tolerance of  $1 \times 10^{-10}$  in the energy norm. To load the structure, we follow a displacement based incremental loading scheme as discussed in Chapter 5.

### 6.4.1 Sensitivity study

We begin with a numerical sensitivity study for an element that contains a triple junction. The interfaces are oriented such that they are parallel to the element edges as shown in Figure 6.5(a). We now vary the location of the junction while keeping the inclination of the interfaces fixed. The position of the junction is obtained using

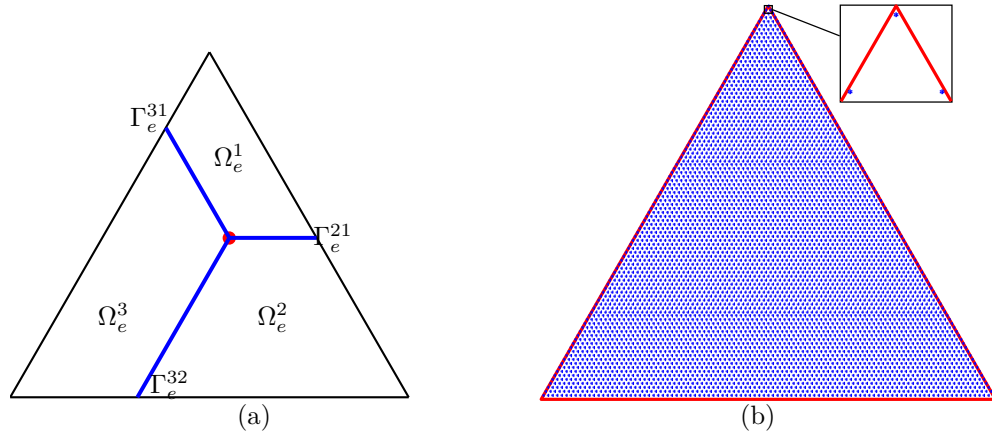


FIGURE 6.5 – Illustration of the parameters for the sensitivity study. On the left, for a particular position of the junction, the orientation of interfaces is shown. On the right, the various positions of the junction used in the study are shown as blue dots. The zoom on the right shows that the junction does not actually lie on the element edge.

a barycentric coordinate system such that:

$$\begin{aligned} x_J &= \lambda_1 x_1 + \lambda_2 x_2 + \lambda_3 x_3, \\ y_J &= \lambda_1 y_1 + \lambda_2 y_2 + \lambda_3 y_3, \end{aligned} \tag{6.19}$$

where  $(x_J, y_J)$ ,  $(x_1, y_1)$ ,  $(x_2, y_2)$ , and  $(x_3, y_3)$  represent the  $(x, y)$  locations for the junction and for the three vertices of the triangular element, respectively. For this study, we consider an equilateral triangle such that  $(x_1, y_1)$ ,  $(x_2, y_2)$  and  $(x_3, y_3)$  are chosen as  $(0, 0)$ ,  $(1, 0)$  and  $(0.5, \sqrt{3}/2)$  respectively.  $(\lambda_1, \lambda_2, \lambda_3)$  represent the barycentric coordinates such that the constraint,  $\lambda_1 + \lambda_2 + \lambda_3 = 1$ , holds. The position of the junction is now varied by generating two vectors, representing  $\lambda_1$  and  $\lambda_2$ , of 100 linearly spaced points between 0.0001 and 0.9999. The entire sample space for the position of junction used in this study is shown in Figure 6.5(b). A zoom of the element in the vicinity of the vertex  $(x_3, y_3)$  is shown in the inset of Figure 6.5(b). From the inset, we can see that for the conducted study, we can have a position for the junction that is arbitrarily close to the element edge and can, therefore, result in arbitrarily small volume fractions.

We now intend to study the conditioning of the discrete system of equations as

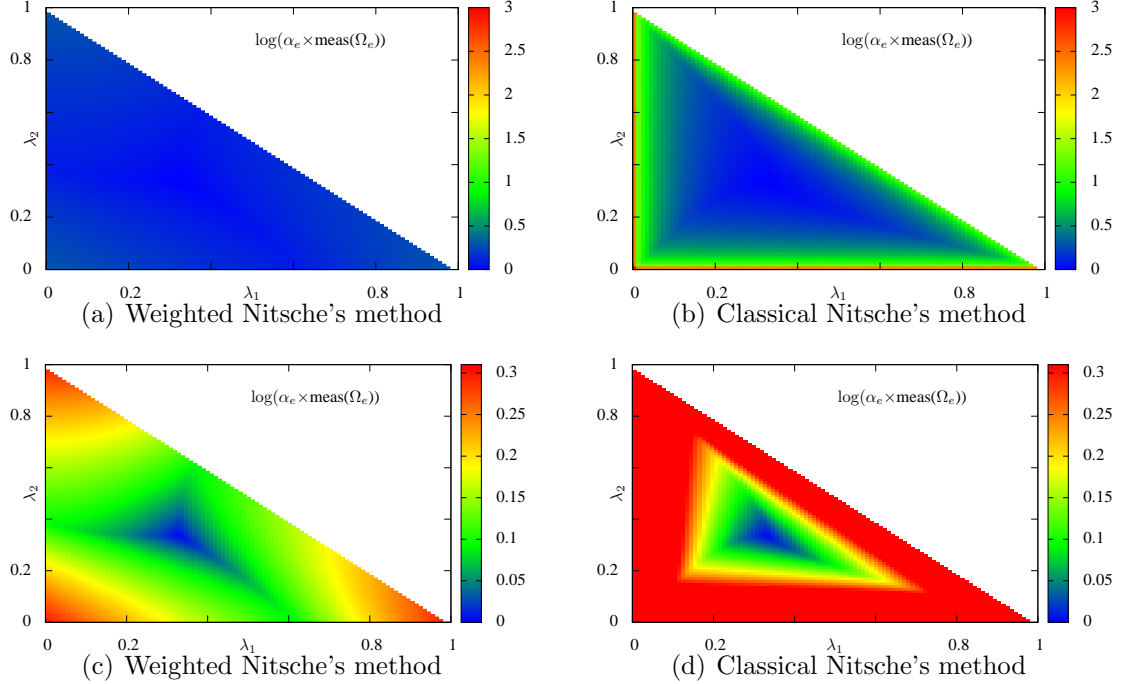


FIGURE 6.6 – Variation of stabilization parameter for the weighted (left) and the classical (right) Nitsche’s method as a function of the barycentric coordinates for the junction. The plotted values for the stabilization parameter are normalized with the element volume. The ranges for the contour plots on the top and bottom are different as seen from the figure legends. On top, the plots are over the entire range of variation of the stabilization parameter obtained from classical Nitsche’s method while on the bottom the range of variation for the stabilization parameter is that obtained from the weighted Nitsche’s method.

the position of junction is varied within an element. A measure of the conditioning of the discrete system of equations can be obtained by studying the rate at which the interfacial stiffness terms grow compared to the bulk stiffness terms. The bulk stiffness terms grow inversely with the element volume while the interfacial stiffness terms grow directly with the stabilization parameter. With this in mind, we normalize the stabilization parameter – obtained from the analytical expression defined in Section 6.3.1 – by multiplying it with the element volume. Also, it must be noted that for the presented case, we have three local stabilization parameters. We conduct the sensitivity analysis on the largest of these values because the largest value will most adversely affect the conditioning of the discrete system. The results of the

variation of this normalized value with the position of the junction within an element are plotted in Figure 6.6.

In the plot, we show a variation of the logarithm of the normalized stabilization parameter on the  $\lambda_1$ - $\lambda_2$  plane. It is instructive to plot the values this way because as either of  $\lambda_1$  or  $\lambda_2$  tend to zero, the position of the junction in the x-y plane tends towards an element edge. This normalized value varies over three orders of magnitude on using the classical Nitsche's method, therefore, we plot it on a log scale to capture this variation better. From Figures 6.6(a) and 6.6(c), we can see that for the weighted Nitsche's method, the normalized value for the stabilization parameter exhibits minimal sensitivity and varies between 1 and 2 (between 0 and 0.3 on the log-scale). This means that the interfacial terms in the stiffness matrix can, at most, grow up to twice as fast as the bulk stiffness terms but the order of magnitude of these terms remains the same. On the other hand, from Figures 6.6(b) and 6.6(d), clearly, the classical Nitsche's method exhibits a much greater sensitivity. In fact, when either of  $\lambda_1$  and  $\lambda_2$  are close to 0, the order of magnitude of the normalized stabilization parameter is  $10^2$  and in the worst possible case when both  $\lambda_1$  and  $\lambda_2$  tend to zero, the order of magnitude of the normalized value tends to  $10^3$ . As a consequence, the interfacial terms can grow up to 1000 times faster than the bulk stiffness terms which could lead to ill-conditioning. As shown in the subsequent examples, the generated mesh could very easily result in such elements and result in a poor approximation.

#### 6.4.2 Poisson's equation

We now conduct a numerical convergence study on a domain with junctions and a large variation in material properties. We consider a domain with multiple intersecting interfaces as shown in Figure 6.7. These interfaces partition the domain  $\Omega$  into 50 component sub-domains  $\Omega^m$ , with each sub-domain representing a single grain.

For this arbitrarily chosen configuration, the material conductivities are randomly distributed within the 50 grains such that they have a large contrast ratio. The conductivities vary between  $\kappa = 1$  and  $\kappa = 3.8 \times 10^6$  units and the distribution is as shown in Figure 6.7. The finite element mesh is chosen such that the regions with higher values of conductivity are more refined.

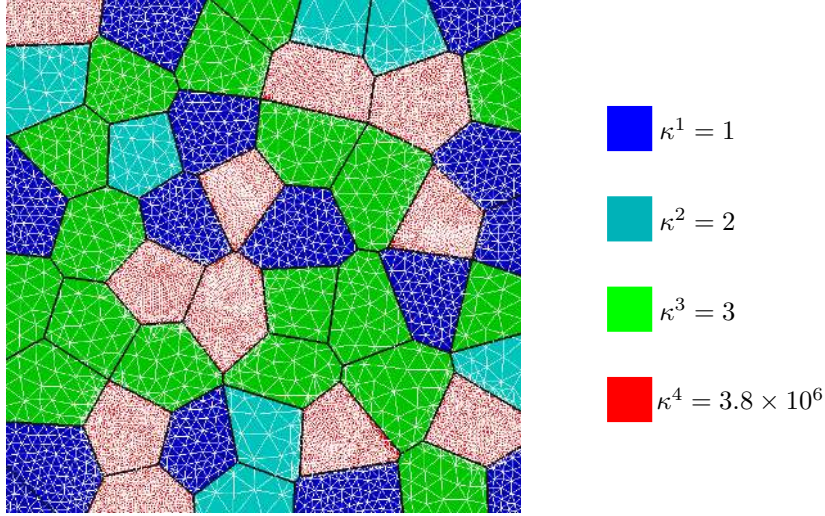


FIGURE 6.7 – Schematic of computational domain, finite element mesh and distribution of material properties.

We now solve the following jump problem on this domain:

$$\begin{aligned}
 \nabla \cdot \kappa^m \nabla u^m &= -f^m && \text{in } \Omega^m \\
 u &= 0 && \text{on } \Gamma_d = \{\mathbf{x} : x = 0 | 1\} \\
 \kappa^m \nabla u^m \cdot \mathbf{n} &= 0 && \text{on } \Gamma_n = \{\mathbf{x} : y = 0 | 1\} \\
 \llbracket \kappa \nabla u \rrbracket^{\text{ab}} \cdot \mathbf{n} &= 0 && \text{on } \Gamma^{\text{ab}} \\
 \llbracket u \rrbracket^{\text{ab}} &= u^{\text{b}}(\mathbf{x}) - u^{\text{a}}(\mathbf{x}) && \text{on } \Gamma^{\text{ab}}
 \end{aligned} \tag{6.20}$$

where  $\Gamma^{\text{ab}}$  is the set of all material interfaces arising in the problem domain. Choosing  $f^m = 1$ , the equation admits an analytical solution given by:

$$u(\mathbf{x}) = \begin{cases} \frac{(3\kappa^1 + (\kappa^2 + \kappa^3 + \kappa^4)/3)x}{4\kappa^m(\kappa^1 + (\kappa^2 + \kappa^3 + \kappa^4)/3)} - \frac{x^2}{2\kappa^m} & \text{in } \Omega^m \end{cases}$$

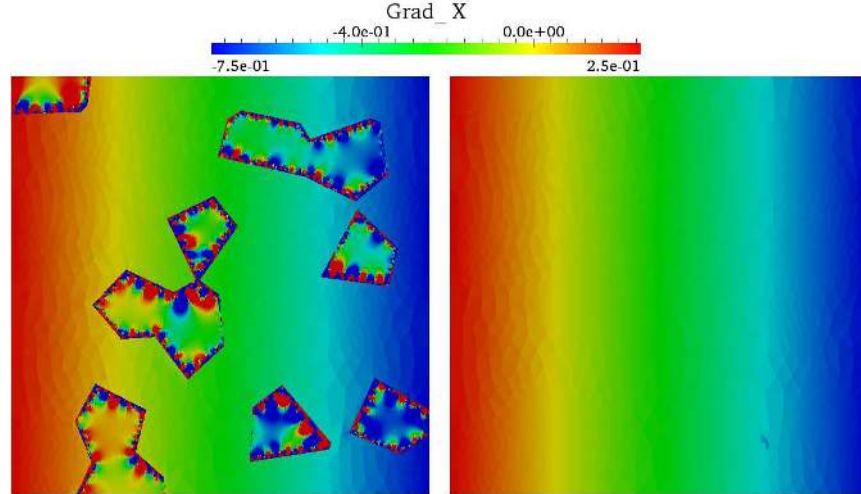
As specified by the interfacial conditions, the solution admits jumps in the primary variable but has a continuous flux across the material interfaces. For the given

distribution of material properties, since  $\kappa^4 \gg \{\kappa^1, \kappa^2, \kappa^3\}$  the product  $\kappa^m \nabla u^m$  varies linearly and is given as  $\kappa^m \nabla u^m \sim 0.25 - x$ . This product represents hydraulic conductivity in porous media flow and local heat flux in thermal conduction problems. The accurate recovery of this quantity is therefore essential in many physical problems. Figure 6.8(a) shows the variation of  $\kappa^m \nabla u^m$  recovered from both the classical Nitsche’s method and the weighted Nitsche’s method. While the weighted Nitsche’s method returns the expected linearly varying behaviour, the classical Nitsche’s method, as noticed before in the case of a single interface (Chapter 4), exhibits wildly oscillatory behavior at every interface where there is a sharp contrast in material properties.

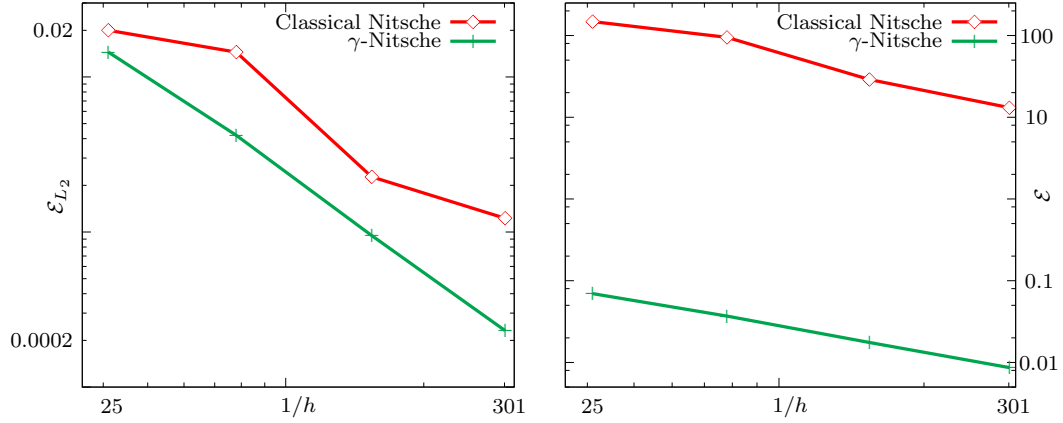
The closed form analytical solution also allows us to perform a numerical mesh convergence study for this problem. The results of this convergence study show sub-optimal rates in  $L_2$  norm for the bulk variable with classical Nitsche’s method. For  $\kappa^m \nabla u^m$ , the classical Nitsche’s method converges in the “energy” semi-norm with mesh refinement but the errors are two orders of magnitude higher than those obtained from the weighted Nitsche’s method. The weighted Nitsche’s method converges at the expected rates, *i.e.* quadratically for the bulk variable in the  $L_2$  norm and linearly for  $\kappa^m \nabla u^m$  in the “energy” semi-norm. The weighted Nitsche’s method also returns a much higher accuracy in both cases. The successful performance of the weighted Nitsche’s method in the presence of junctions and large contrast ratios for linear problems motivates our next study where we investigate its behaviour for a perfectly plastic interface.

### 6.4.3 Composite beam study

To begin with, we return to the composite beam study we conducted earlier in Chapter 5 of this work. However, we now investigate this problem in the presence of junctions and then compare the results with those obtained when we had a single



(a) Variation of  $\kappa^m \nabla u^m$  with both classical (left) and weighted Nitsche's method (right)



(b) Convergence study in  $L_2$  norm for the bulk variable

(c) Convergence study in “energy” norm for  $\kappa^m \nabla u^m$

FIGURE 6.8 – Top: Variation of  $\kappa^m \nabla u^m$  across the domain. Bottom left: Convergence in  $L_2$  norm for the bulk variable. Bottom right: Convergence in energy norm for  $\kappa^m \nabla u^m$

embedded interface. For completeness, we repeat the problem set-up and loading conditions here.

We consider a composite beam of length  $L = 16$  units and height  $H = 4$  units being loaded under compression in a state of plane stress. The domain  $\Omega = (-L/2, L/2) \times (-H/2, H/2)$  is partitioned into five component sub-domains by seven embedded interfaces as shown in Figure 6.9. To reproduce the problem set-up consid-



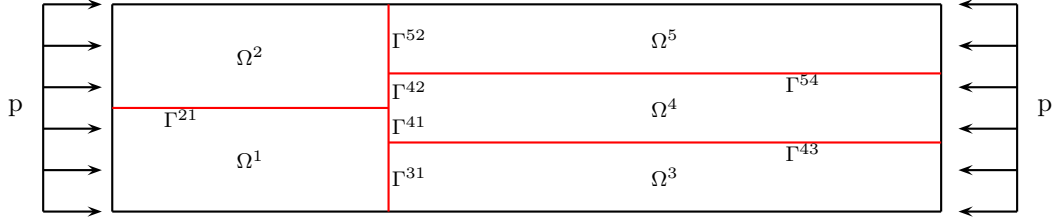
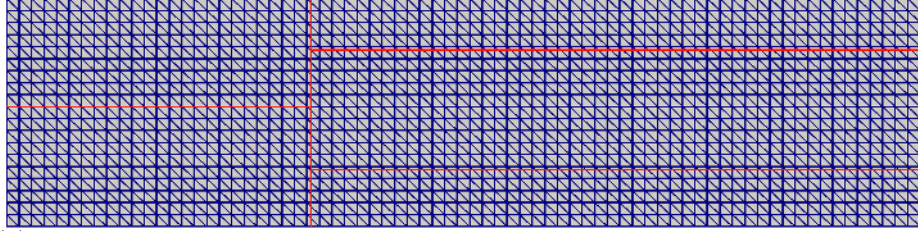


FIGURE 6.9 – Illustration of the geometry and loading conditions for a beam with several material interfaces loaded in pure compression.

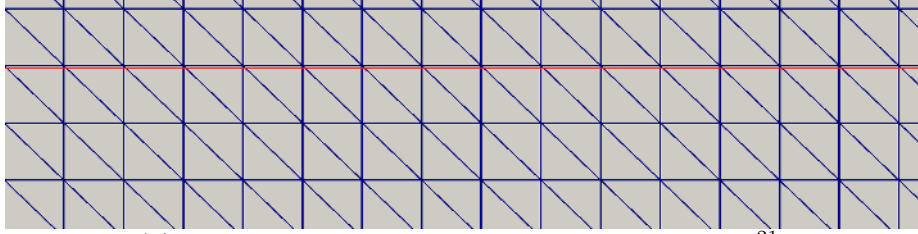
ered in Chapter 5, we choose identical Young’s moduli for all the sub-domains such that  $E^m = 1000$  units where  $m \in [1, 5]$ . We also choose the Poisson’s ratios in the sub-domains  $\Omega^1$  and  $\Omega^2$  to be identical such that  $\nu^1 = \nu^2 = 0.3$ . We further require that the sub-domains to the right of the vertical interface,  $\Gamma = \Gamma^{31} \cup \Gamma^{41} \cup \Gamma^{42} \cup \Gamma^{52}$ , have identical Poisson’s ratios:  $\nu^3 = \nu^4 = \nu^5 = 0$ .

We now recover a problem where sliding occurs only along the vertical interface  $\Gamma$  if we can ensure that the interfaces  $\Gamma^{21}$ ,  $\Gamma^{43}$  and  $\Gamma^{54}$  are in perfect contact. We ensure this by specifying a large value for the yield traction on these interfaces, in particular, we choose  $t_Y = 10^{16}$  units. For the vertical interface  $\Gamma$ , we vary the yield traction  $t_Y$  between 0 units and  $10^{16}$  units to simulate different material response ranging from frictionless sliding to perfect contact along this interface. We discretize the domain using a structured triangular mesh with 73 divisions in the x-direction and 19 divisions in the y-direction as shown in Figure 6.10(a). A zoom of the interface  $\Gamma^{21}$  in Figure 6.10(b) also highlights that the chosen discretization results in arbitrarily small elements.

We assume a uniaxial state of stress  $\sigma_{xx} = -25$  units for the beam. For the prescribed state of stress, the displacement field is linear and continuous in the x-direction while it is linear and discontinuous in the y-direction. In particular, the



(a) Finite element mesh used in computations for the beam with several interfaces.



(b) Zoom of the mesh near the embedded interface  $\Gamma^{21}$ .

FIGURE 6.10 – Illustration of the finite element mesh used in the composite beam study.

displacement field is given by:

$$u_x(\mathbf{x}) = \frac{\sigma_{xx}}{E}x \quad \text{in } \Omega, \quad u_y^m(\mathbf{x}) = y\epsilon_{yy}^m \quad \text{in } \Omega^m, \quad (6.21)$$

where the strain  $\epsilon_{yy}^m$  is:

$$\epsilon_{yy}^m(\mathbf{x}) = -\frac{\nu^m}{E^m}\sigma_{xx} \quad \text{in } \Omega^m \quad (6.22)$$

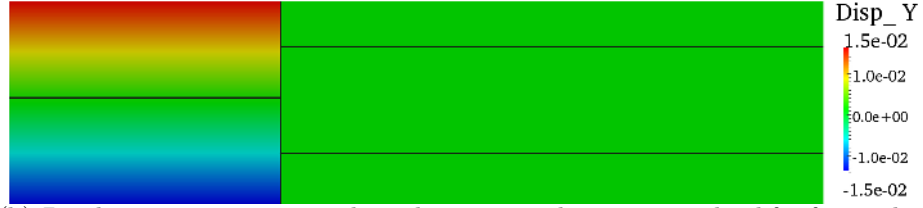
These displacement values are now applied as Dirichlet boundary conditions on the boundaries  $\Gamma_d = \{\mathbf{x} : x = -L/2|L/2\}$ . The remaining boundaries are considered traction-free.

In Figures 6.11(a) and 6.11(b), we plot the displacement contours in both the x and y-directions respectively at the maximum prescribed load for the case when the yield traction  $t_Y$  on the vertical interface  $\Gamma$  is 0 units. From the figures, we see that the obtained results are in excellent agreement with the expected analytical fields. In the x-direction, the displacement is linear and continuous throughout the domain. In the y-direction, while the displacements are linear in domains  $\Omega^1$  and  $\Omega^2$  and continuous across them, we also clearly see the discontinuity across the vertical

interface  $\Gamma$  that arises from the discontinuous Poisson's ratio.



(a) Displacement contours in the x-direction at the maximum load for frictionless sliding along  $\Gamma$ .



(b) Displacement contours in the y-direction at the maximum load for frictionless sliding along  $\Gamma$ .

FIGURE 6.11 – Displacement contours at the maximum prescribed load in the composite beam study.

In Figures 6.12(a) – 6.12(f), we plot the y-component of the displacement field when the beam is back in the fully unloaded state for different values of yield traction on the vertical interface  $\Gamma$ . This serves as an indication for the residual plastic strain in the beam. As expected, for the cases when the interface is perfectly tied - 6.12(f) and when the interface is sliding under frictionless contact - 6.12(a), the residual plastic strain is zero due to the reversible nature of these kinematically linear problems. However, for all other cases, we end up with some residual plastic strain which is decreasing with increasing yield traction. Finally, we also remark that the contours are in excellent agreement with those obtained for a single embedded interface in Chapter 5 indicating that the numerical method is performing well in the presence of both junctions and nonlinear interfacial response.

#### 6.4.4 Unit cell geometry

We now consider a geometry similar to that examined previously by Ghahremani (1980) and subsequently by Simone et al. (2006). The chosen geometry might arise

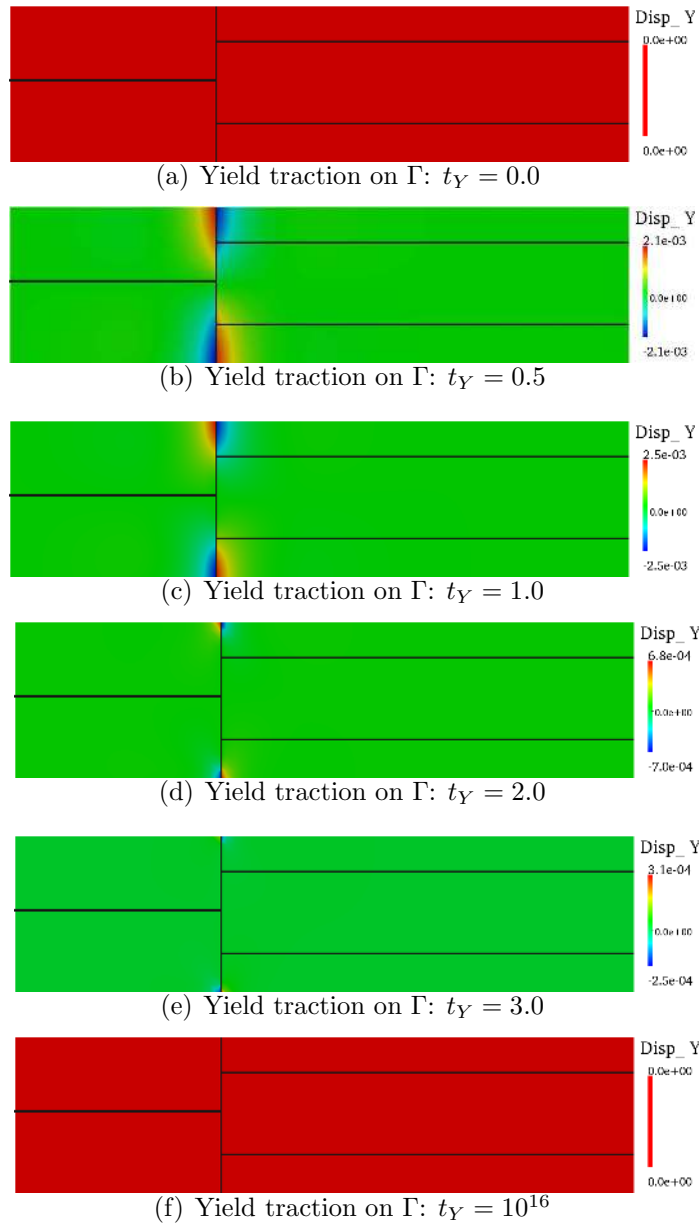


FIGURE 6.12 – Residual strain in the fully unloaded state with yield stress at the interface presented in ascending order. Case (a) represents a frictionless sliding problem while case (f) represents a perfectly tied interface. All the other sub-cases allow for frictional sliding.

as a unit cell under the assumption of modeling polycrystals as a periodic array of hexagonal cells. However, in the present context, our motivation is not so much to investigate micro-mechanical response but to better understand the performance of the proposed numerical method in handling such geometries efficiently.

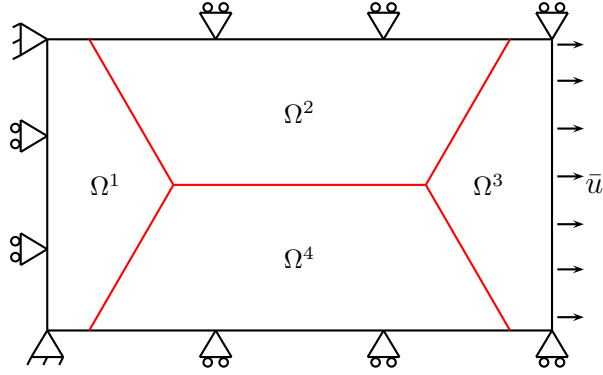


FIGURE 6.13 – Illustration of the geometry and loading conditions for the unit-cell problem.

To that end, we now set up a test problem where the described geometry is being loaded in uniaxial tension through a prescribed displacement  $\bar{u} = 3 \times 10^{-3}$  units on the extreme right boundary. All other boundaries are considered to be constrained in the normal direction through rollers. The geometry and loading conditions are described in Figure 6.13. The grain boundaries are considered to behave under perfect plasticity. We use a structured triangular mesh with 101 divisions in the x-direction and 77 divisions in the y-direction to discretize the domain. The material properties are chosen identically in all the sub-domains with the Young's modulus  $E^m = 1000$  units and the Poisson's ratio  $\nu^m = 0.3$ . We now study the response of the structure by varying the yield traction between the boundaries for four distinct cases: (a)  $t_Y = 0.0$  units, (b)  $t_Y = 0.10$  units, (c)  $t_Y = 0.25$  units, and (d)  $t_Y = 10^{16}$  units. We plot the von Mises stress for all the four cases in Figure 6.15. We also plot the displacement contours in the x and y directions for the case with  $t_Y = 0.10$  units in Figures 6.14(a) and 6.14(b) respectively.

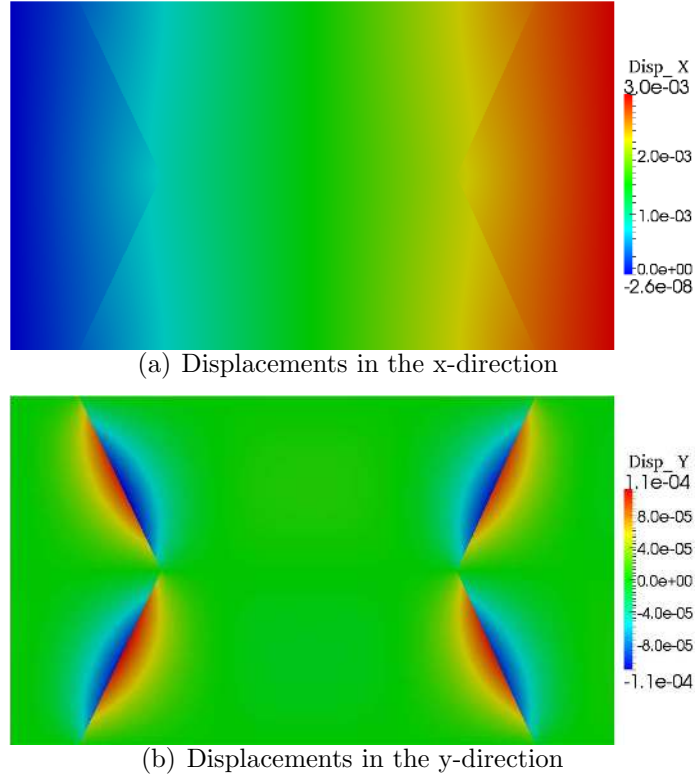


FIGURE 6.14 – Displacement field for the unit-cell problem for the case when the yield traction between the boundaries is  $t_Y = 0.10$  units.

Firstly, from the displacement contours in Figure 6.14, the discontinuity in the displacement due to the tangential sliding along the inclined interface is clearly evident. Further, from the von Mises stress plot in Figure 6.15, we also clearly see the stress concentrations near the triple junctions as well as at the points where the interface intersects the external boundaries. At the external boundary, the enforced boundary conditions are such that sliding along the interface is prevented. Similarly, the triple points provide an internal constraint against sliding in order to ensure compatibility. At both these points, this increased resistance against sliding causes considerably higher stress build-up than in other parts of the domain.

Further, we also notice that as the yield traction on the grain boundaries increases, the stress concentration at the junctions is reduced. This reduction is expected because an increased value of yield traction signifies higher resistance to sliding and

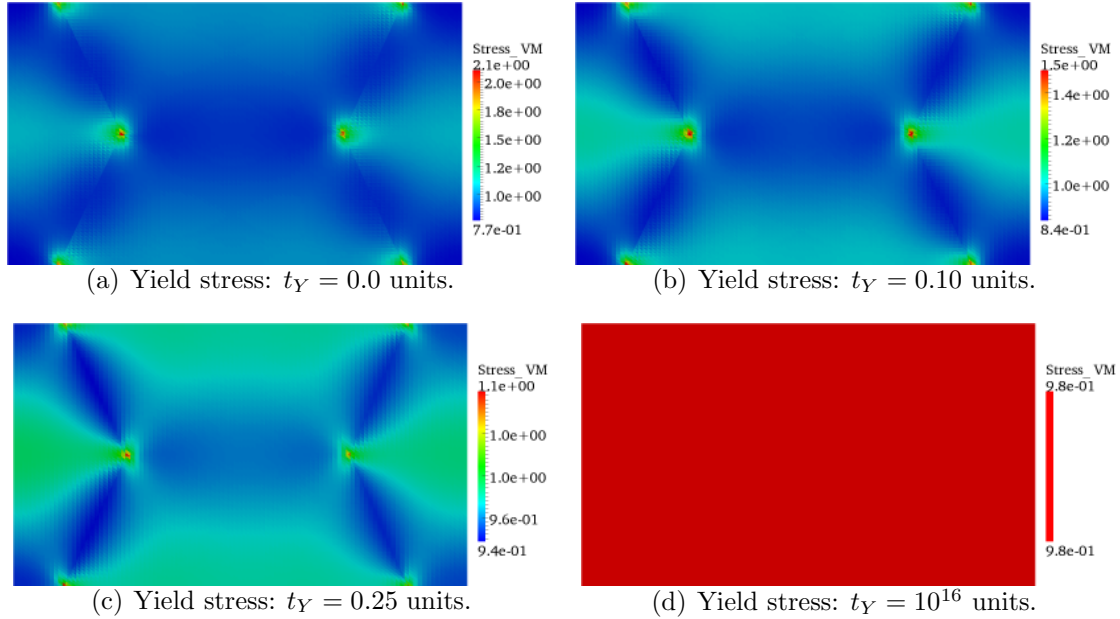


FIGURE 6.15 – von Mises stress for the unit-cell problem presented in ascending order of yield tractions at the interface.

therefore the geometric constraint at the junctions that prevents sliding has lesser bearing than it has on a freely sliding grain boundary. We also note that as the yield traction on the grain boundaries increases, stress is transferred much more effectively across the grain boundaries and the stress contours are much flatter and tending towards a uniform state of stress. Finally, in case (d), the large value of the yield traction makes the interfaces behave as they would in a perfectly tied scenario. For the prescribed loading, we would expect a uniform stress state in the entire problem domain and the results shown in Figure 6.15(d) agree with this expectation.

#### 6.4.5 Polycrystalline geometry

##### *Shear loading*

We now consider the shear deformation of a polycrystalline material at the meso-scale. We consider the polycrystalline microstructure to be generated from a Voronoi tessellation of the computational domain  $\Omega = [0, 1] \times [0, 1]$ . In order to generate the Voronoi tessellation, we randomly distribute the grain centers in the computational

domain  $\Omega$ . We however follow Wei and Anand (2004) and impose a numerical tolerance between any two of these randomly generated grain centers. This prevents the Voronoi tessellation from generating uncharacteristically small grains and also prevents a large difference between the grain volumes. We also set a numerical tolerance to prevent the generation of extremely small edges from this tessellation. Consequently, the procedure to generate microstructures that perform well in finite element computations is automated and we do not have to discard any generated realizations.

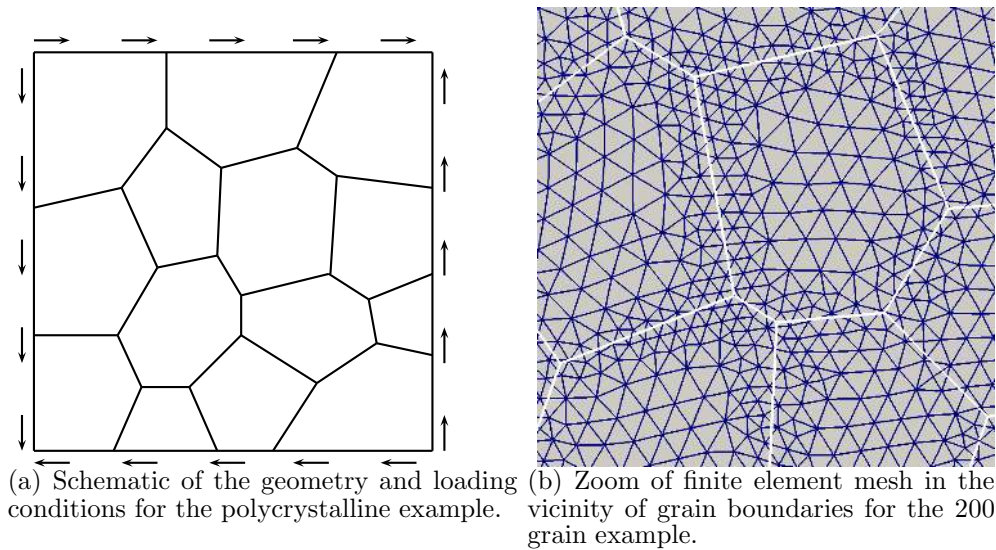


FIGURE 6.16 – Description of the geometry, loading conditions and finite element mesh for studying shear response of a polycrystalline specimen.

We now qualitatively study the effect of grain size on the material response by varying the number of grains in the domain. We specifically consider three cases, when the domain is discretized by (a) 14 grains, (b) 50 grains, and (c) 200 grains. In order to represent the heterogeneous character of materials at this length scale, we randomly distribute the materials properties for each of the grains between  $E = 1.8 \times 10^5$  units and  $E = 2.1 \times 10^5$  units. Further, the grain boundary characteristic property is also varied randomly between  $t_Y = 35$  units and  $t_Y = 50$  units. The



distribution of both the bulk and grain boundary properties is shown in Figures 6.17(a) through 6.17(f).

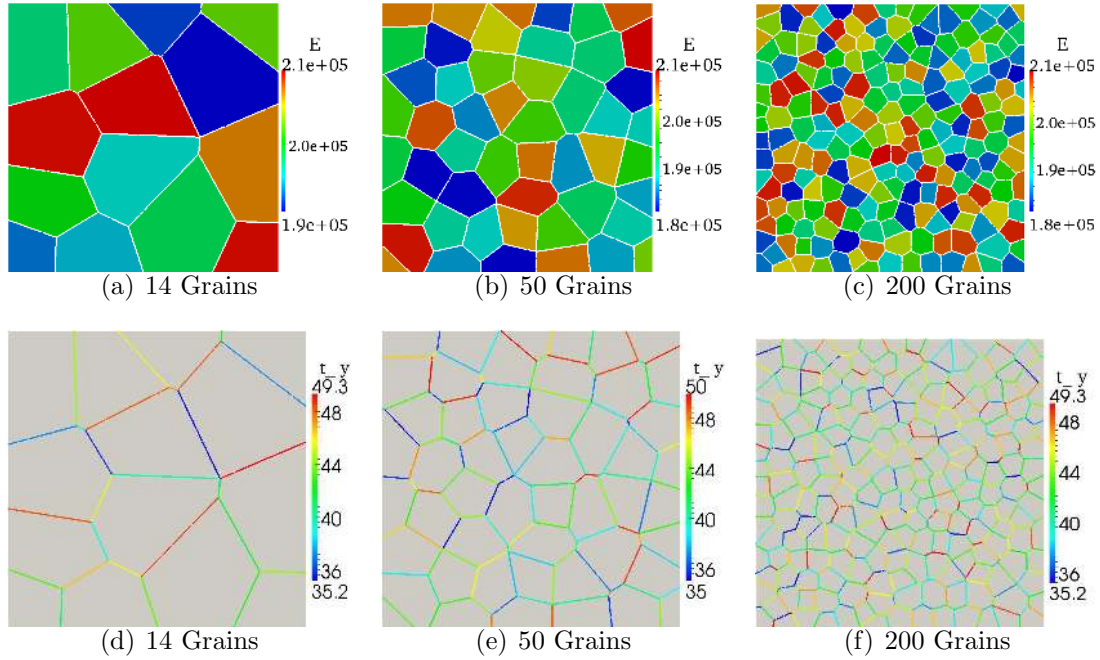


FIGURE 6.17 – Distribution of Young’s moduli (top) and yield tractions (bottom) for randomly generated polycrystalline specimens.

We use an unstructured finite element mesh for each case generated using the open-sourced finite element meshing program Gmsh (Geuzaine and Remacle, 2009). The nonlinear character of the response near the grain boundaries requires greater resolution in this region. The generated mesh is thus more refined near the grain boundaries. Figure 6.16(b) presents the zoom of the mesh in the vicinity of a grain boundary for the case with 200 grains. Notice also from this plot that we have no control over the location of junction within an element and we can potentially have arbitrarily small elements. The number of elements is progressively increased for each case. For the case with 14 grains, the originally generated mesh consists of 10,071 constant strain triangular elements and the modified mesh after accounting for enriched elements consists of 11,035 elements. For the case with 200 grains, the mesh consists of 39,502 elements before accounting for enrichments and 48,261

elements after the enriched elements have been added.

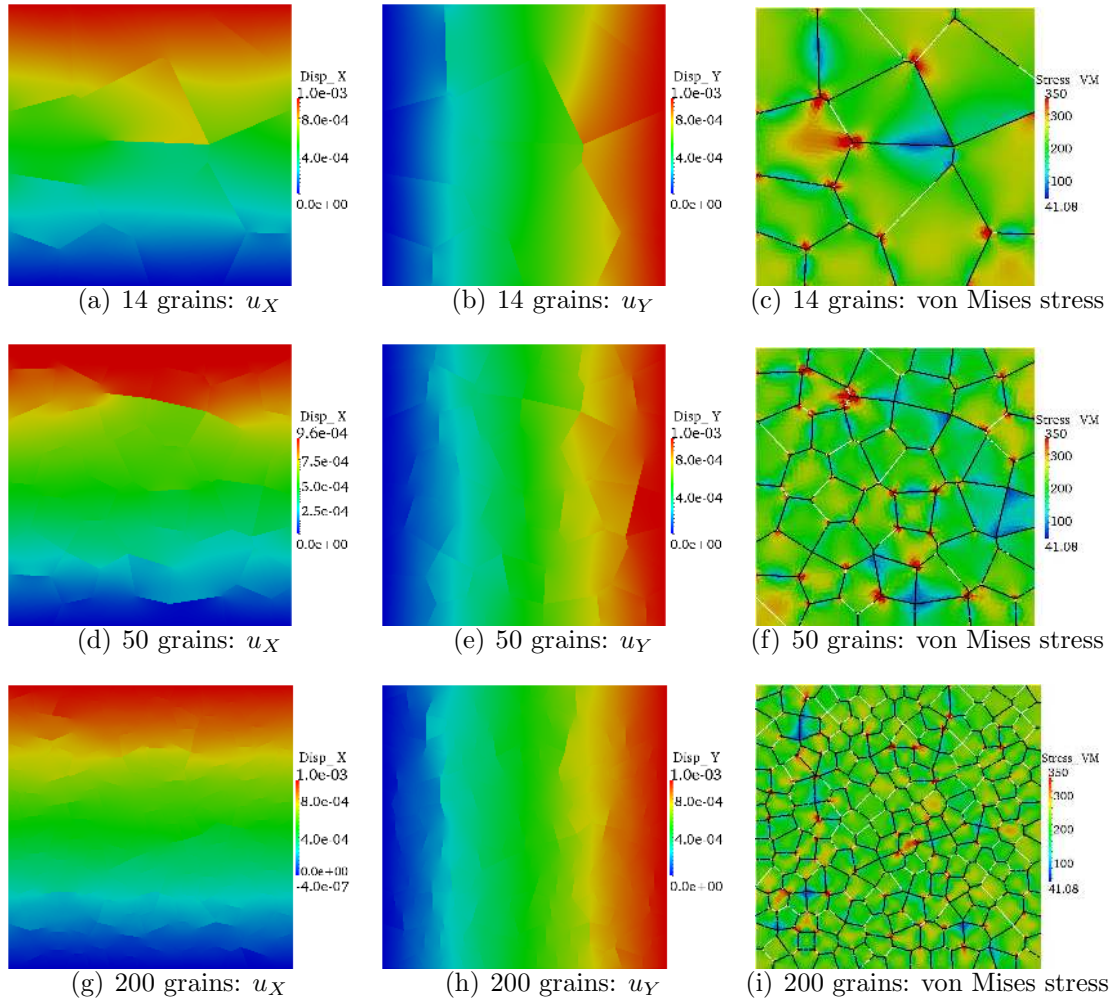


FIGURE 6.18 – Magnitude of displacement and von Mises stress for the polycrystalline geometry subjected to shear loading. Number of grains is varied from 14 to 200. Discontinuities in displacements are clearly evident in the contour plots at the grain boundaries. Stress concentrations at the junctions are also clearly visible. Grain boundary regions shown in black are under slip while those shown in white are sticking.

The loading is prescribed on the computational domain through Kinematic Uniform Boundary Conditions (KUBC), as discussed in the works of Kanit et al. (2003) and Huet (1990). Recalling the definition presented in Kanit et al. (2003), under the assumptions of KUBC, for a point  $\mathbf{x}$  belonging to the boundary  $\Gamma_d$  the prescribed displacement is specified as:

$$\mathbf{u}(\mathbf{x}) = \mathcal{E} \cdot \mathbf{x} \quad \forall \mathbf{x} \in \Gamma_d \quad (6.23)$$

The macroscopically averaged strain and stress tensors are given by:

$$\mathcal{E} = \frac{1}{V} \int_V \varepsilon \, dV; \quad \Sigma = \frac{1}{V} \int_V \sigma \, dV \quad (6.24)$$

Now, in order to prescribe a pure shear state, we consider that only the off-diagonal terms in the macroscopic strain tensor are populated. Here, at the maximum prescribed load, we arbitrarily set these terms as 0.001.

$$\mathcal{E} = \begin{bmatrix} 0 & 0.001 \\ 0.001 & 0 \end{bmatrix} \quad (6.25)$$

We now use equation (6.25) in equation (6.23) to prescribe the displacement values, incrementally, on all the external boundaries.

We plot the displacement contours and von Mises stress plots for all the three cases, at the maximum prescribed load, in Figures 6.18(a) through 6.18(i). The displacement discontinuity that arises due to the tangential sliding along the grain boundaries is clearly evident in the contour plots. Further, it is also clear from the von Mises stress plots that stress concentration arises at the junctions of the grain boundaries due to the geometrical constraint that prevents sliding. At the prescribed load, the grain boundaries that are sliding tangentially are shown in black while the grain boundaries that continue to be in perfect contact are shown in white. It is clear from the plot again that at the maximum prescribed load, a large percentage of grain boundaries is in slip. However, on comparing the cases with 50 and 200 grains, we also notice that as the number of junctions increases, due to the increased geometrical constraints against sliding, a greater percentage of total grain boundaries is maintaining perfect contact.

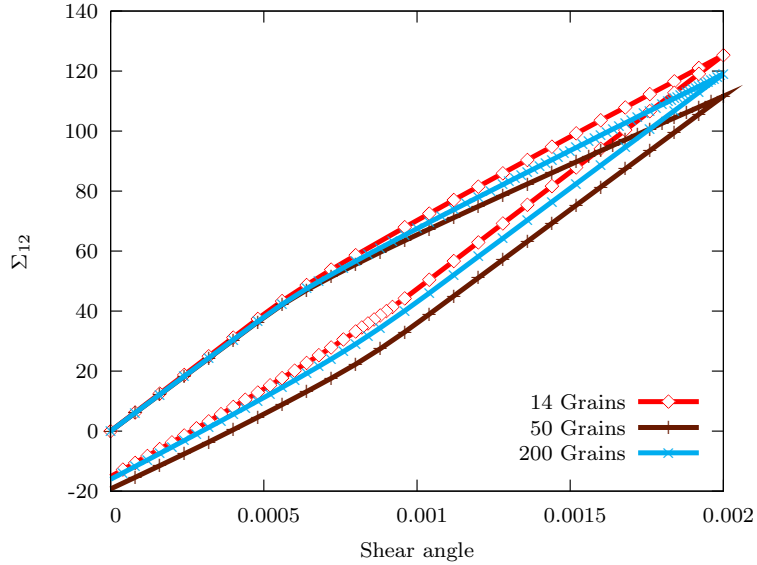


FIGURE 6.19 – Spatially averaged stress-strain plot.

Finally, we also plot the macroscopically averaged shear stress  $\Sigma_{12}$ , given by equation (6.24), against the shear angle for all the three cases in Figure 6.19. From the plot, we can see that as the number of grains increases from 14 to 50, the material response clearly becomes softer. We can attribute this to the fact that as the number of grains increases, the grain boundaries occupy a greater percentage of the total volume. Consequently, a larger percentage of the total volume is slipping and shows up as the softened material response in Figure 6.19. However, as the number of grains is further increased from 50 to 200, not only does the ratio of grain boundary volume to the total volume increase but it also results in an increase in the number of junctions. Now, even though a larger percentage of grain boundary volume provides with an increased potential for sliding, the increased number of junctions offers greater resistance against sliding. These competing mechanisms show up in Figure 6.19 as an increased shear resistance for the case with 200 grains as compared to the case with 50 grains.

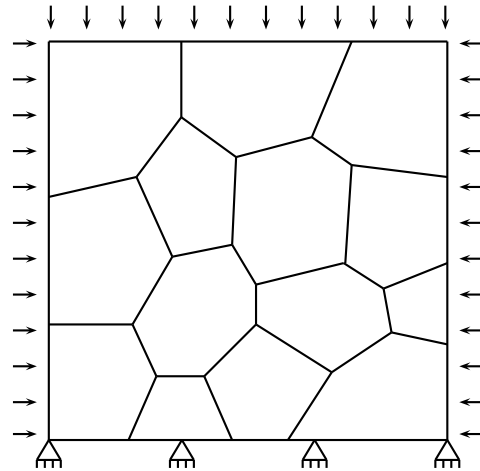
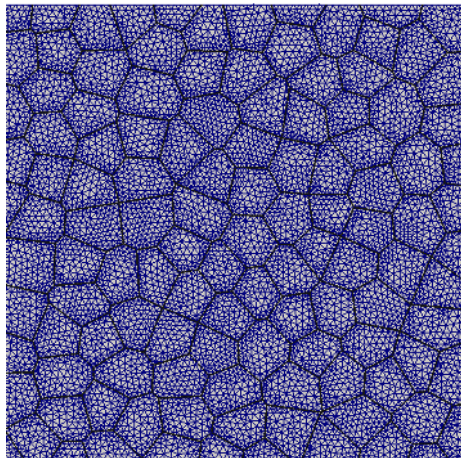


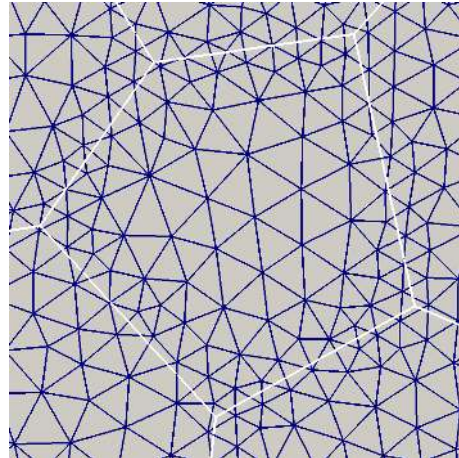
FIGURE 6.20 – Schematic of the geometry and loading conditions for studying the effect of confinement on a polycrystalline sample.

### *Effect of confinement*

As a final example, we study the effect of lateral confinement on the shear resistance of alumina ceramic. We consider a unit square domain tessellated with 120 grains and meshed with a finite element mesh consisting of 16294 constant strain elements before enrichment and 20681 elements after enrichment. The geometry and loading



(a) Finite element mesh for a 120 grained microstructure.



(b) Zoom of finite element mesh in the vicinity of grain boundaries.

FIGURE 6.21 – Finite element mesh used for studying the effect of confinement on a polycrystalline specimen with 120 grains.

conditions are shown in Figure 6.20. The finite element mesh used in computations is shown in Figure 6.21(a) and a zoom of the finite element mesh in the vicinity of a grain boundary is shown in Figure 6.21(b). The material properties are chosen to represent those of aluminium *viz.* the modulus of elasticity  $E = 70$  GPa and the Poisson's ratio  $\nu = 0.3$ .

This example is inspired by a similar study conducted to investigate the effect of confining pressure on the compressive strength of alumina ceramic by Warner and Molinari (2006). However, their material model allowed for fracture along grain boundaries. In the present study, we are only permitting grain boundary sliding and consequently the compressive strength will remain unaffected. Due to the presence of tangential sliding, we would expect the shear strength, on the other hand, to be influenced by confining pressure.

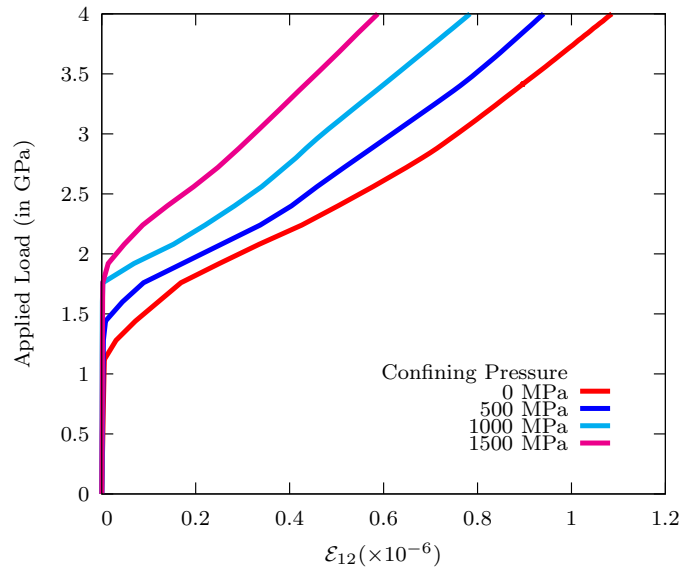


FIGURE 6.22 – Spatially averaged stress-strain plot to study the effect of confining pressure.

Following their work, we neglect the affect of anisotropy in the grains and assume that the grain interiors follow a linear elastic constitutive relationship. Also following their work, we choose the shear strength along grain boundaries as  $t_Y = 0.63$  GPa.

We apply a compressive load of 4 GPa on the top surface, incrementally, following the load-stepping procedure described at the beginning of this section. However, the confining pressure was applied as a uniform constant pressure throughout. We now vary the confining pressure from 0-1500 MPa in steps of 500 MPa. We plot the macroscopically averaged shear strain with the applied loading.

The results of this study are shown in Figure 6.22. From the plot, we clearly see that as the confining pressure is increased, there is a decrease in the macroscopically averaged shear strain for the same value of applied load. In fact, we notice that not only does sliding begin at a larger load as confining pressure is increased but also that the shear modulus – given by the slope of the stress-strain curve once sliding begins – is directly dependent on the confining pressure.

Under the applied uniaxial compressive loading, shear deformation along the grain boundaries results from the lateral expansion of the material due to Poisson’s effect. However, application of confinement pressure prevents this lateral expansion resulting from Poisson’s effect. As a result, sliding along grain boundaries is now much more constrained than in the absence of confining pressure and hence shows up as an increased resistance in the shear response for the material. These results bear resemblance with those reported by Warner and Molinari (2006) on the effect of confining pressure on the compressive strength of alumina ceramic.

We conclude this section by remarking that the performance of this method in handling such complicated geometries without explicitly meshing them is encouraging and further demonstrates its utility in handling complex micro-mechanical or geo-mechanical problems.

## 6.5 Conclusion

We presented a Nitsche stabilized embedded method to enforce stiff kinematic constraints over embedded interfaces with junctions. In particular, we derive a geometric

expression for the Nitsche stabilization parameter for constant strain elements in the presence of intersecting interfaces such that the method remains stable. We also provide algorithmic details (see Appendix A) to extend the Nitsche stabilized method for frictional sliding problems – presented in Chapter 5 – to embedded interfaces with junctions. Through several numerical examples, we demonstrate the performance of this for both kinematically linear and non-linear problems. Indeed, the presented approach results in expected rates of convergence in both bulk and interfacial quantities for linear problems even in the presence of junctions. The presented approach is also very robust in the presence of both large heterogeneities and arbitrarily small elements. We also show through a perfectly plastic interfacial law that the method is capable of handling topologically challenging problems that could arise in micro-mechanical modeling and geomechanics in an efficient manner.

Going forward, we remark that while the developed approach has significant promise in handling micro-mechanical and geo-mechanical problems, there are several simplifying assumptions that need to be relaxed before it could become truly predictive in nature. For instance, it would be interesting to extend the method to handle large sliding at the interface. It will also be interesting to include nonlinearities in the bulk constitutive behavior. Moreover, it will be useful to extend the method to handle more challenging interfacial constitutive behavior that could model combined sliding and separation. As for the extension of the method to three-dimensional problems is concerned, though conceptually identical, one needs to account for the implementational challenges associated with (a) computational geometry for random Voronoi tessellations and (b) the algorithmic aspects of frictional sliding on a surface.



## Conclusion

In this dissertation, we developed a finite element method that can treat interfacial phenomena in a computationally efficient manner. The proposed approach allows an arbitrary representation of the interface geometry from the underlying finite element mesh. This flexibility translates to significant savings in computational time for problems where the interface exists on a complex topology or evolves in time. However, a robust enforcement of interfacial kinematics is central to the performance of the proposed approach.

We showed that many existing approaches to enforce these interfacial kinematics suffer from either a lack of stability or a lack of variational consistency. The lack of variational consistency translates to poor constraint enforcement while the lack of stability results in a spurious oscillatory behavior for the interfacial field. In many ways, there is a competition between how well the constraint is enforced versus the recovery of the interfacial quantity of interest. Moreover, if explicit time stepping is desired, then we also have an additional competing parameter – the critical time step. Perhaps the key question that needed addressing concerned the development of an approach that would resolve this competition without adversely affecting either

parameter. To accomplish this, we developed stable and stabilized methods for enforcing constraints on embedded interfaces, with particular focus on the stabilized approach.

The stable method relied on algorithmic coarsening of the multiplier space and carefully constructing a basis for the multipliers from the underlying bulk shape functions. The stable method is certainly a robust choice to enforce constraints for this class of problems. However, it does suffer from certain shortcomings. It requires additional pre-processing to coarsen the multiplier space. This coarsening needs to be performed every time the interface changes orientation and this could add computational expense for evolving interface problems or statistical calculations in micro-mechanics. Also, the stable method is incompatible with explicit dynamics and is, in general, more intrusive to implement in existing finite element packages.

The stabilized method, on the other hand, is a primal method based on Nitsche's variational form. The method eliminates the need for construction of a stable multiplier space by replacing the multiplier field with its physical interpretation. However, the stability of the method and consequently the performance depend strongly on certain method parameters that arise in this variational form. We identified a relationship between these parameters and enabled a way to systematically eliminate them in terms of other known quantities, namely mesh-size and material constants. In fact, for lower-order elements used in this dissertation, we provided explicit expressions for these parameters. With these choices, we extended Nitsche's approach to elastostatics, elastodynamics and small-deformation frictional sliding over embedded interfaces (including junctions).

There are several avenues that could be explored in future. From a method development perspective, the proposed stabilized method could be extended to handle non-linearities in the bulk behavior. In the past, there have been open questions concerning the development of a variational form for non-linear elasticity. However,

proceeding through a Lagrange multiplier approach and replacing the multipliers through their physical interpretation could help us answer some of these questions. The impact of this nonlinearity however, on the choice of stabilization and weighting remains to be seen. Further, extensions to bulk plasticity would also be of interest. Clearly, the limitations of standard linear triangles and tetrahedral elements could pose challenges in this regard.

Modeling large sliding over these embedded surfaces using the proposed weighted approach could be useful in many areas of computational geo-sciences. Additionally, the promising performance of the weighted Nitsche approach in frictional contact over embedded surfaces makes it an attractive choice for such problems even with standard finite element approximations where often similar issues with constraint enforcement are experienced.

From a physical perspective, the approach could be used to model more challenging interfacial behavior with combined sliding and separation. In particular, exploring the effects of grain boundary sliding and separation in nano-crystalline materials could be an interesting problem in computational materials science. Many challenging questions concerning the effects of grain size on the competition between bulk and grain-boundary dissipation mechanisms remain to be thoroughly examined for lack of a cost-efficient solver. This approach could provide a convenient platform to explore some of these questions.

# Appendix A

## A Nitsche Stabilized Embedded Finite Element Method for Junctions

In this appendix, we give the governing equations and discretized forms for frictional sliding problems with junctions.

### A.0.1 Frictional Sliding

The governing equations in each of the domains shown in Figure 6.1 are given, in indicial notation, by:

$$\begin{aligned}\sigma_{ij,j}^m &= 0 && \text{in } \Omega^m, \\ u_i^m &= u_i^d && \text{on } \Gamma_d/\Omega^m, \\ \sigma_{ij}^m n_j^m &= h_i && \text{on } \Gamma_n/\Omega^m,\end{aligned}\tag{A.1}$$

where  $\sigma_{ij}^m$  and  $u_i^m$  denote the components of the stress and displacement fields in domain  $\Omega^m$ , respectively, and  $n_j^m$  the components of the unit outward normal. The displacement is fixed to the prescribed field  $u_i^d$  on the Dirichlet portion of the boundary, and  $h_i$  denotes the prescribed tractions on the Neumann portion of the boundary. By  $\Gamma_d/\Omega_m$  and  $\Gamma_n/\Omega_m$ , we denote the part of Dirichlet and Neumann boundaries that belong to  $\partial\Omega^m$  - the boundary of the domain  $\Omega^m$ .

### A.0.2 Weak form

The weak statement of equilibrium for the above strong form is given as: Find  $u_i \in \mathcal{U}_i$  such that

$$\begin{aligned}
& \sum_m \int_{\Omega^m} w_{(i,j)}^m \sigma_{ij}^m \, d\Omega - \sum_r \int_{\Gamma_{d/\Omega^r}} w_i^r t_i^r \, d\Gamma - \sum_r \int_{\Gamma_{d/\Omega^r}} u_i^r t_i^r w^r \, d\Gamma \\
& + \sum_r \int_{\Gamma_{d/\Omega^r}} w_i^r \alpha^r u_i^r \, d\Gamma - \sum_{ab} \int_{\Gamma^{ab}} \llbracket w_i \rrbracket^{ab} t_i^{ab} \, d\Gamma = \sum_s \int_{\Gamma_{n/\Omega^s}} w_i^s h_i \, d\Gamma \quad (\text{A.2}) \\
& - \sum_r \int_{\Gamma_{d/\Omega^r}} u_i^d t_i^r w^r \, d\Gamma + \sum_r \int_{\Gamma_{d/\Omega^r}} w_i^r \alpha^r u_i^d \, d\Gamma \quad \forall w_i \in \mathcal{V}_i,
\end{aligned}$$

where  $\sum_m$  denotes a sum over all the bulk domains and by  $\sum_{ab}$ , we denote the sum over all internal boundaries  $\Gamma^{ab}$ . By  $\sum_r$  and  $\sum_s$ , we denote the sum over all domains  $\Omega^r$  and  $\Omega^s$  such that  $\Gamma_{d/\Omega^r} \neq \emptyset$  and  $\Gamma_{n/\Omega^s} \neq \emptyset$ , respectively. Also,  $t_i^m = \sigma_{ij}^m n_j^m$  represent the components of the traction field obtained by projecting the stress from the corresponding domains on to the boundary of the domain. Finally,  $\alpha^r$  represents the stabilization parameter on the Dirichlet boundary and  $\llbracket w_i \rrbracket^{ab}$  represents the jump  $w_i^b - w_i^a$  on the interface  $\Gamma^{ab}$ .

### A.0.3 Discretization

We now discretize the weak form introduced above (A.2) using the overlapping element approach described in Section 6.3. For elements without interfaces and for elements with a single embedded interface, the resulting local matrices remain identical to those previously described in Chapter 5. Here, for simplicity, we focus only on a tangent matrix for an element with a junction. For those elements which also contain a Dirichlet boundary, to the tangent matrix presented below, contribution from terms that weakly enforce Dirichlet boundary conditions needs to be added. Also, we assume that the load is applied incrementally and solve the non-linear set

---

**Algorithm 5** Updating interfacial tractions for an element with intersecting embedded interfaces at the (n+1)-th load step and k-th iteration

---

**for all** local embedded-interfaces  $\Gamma_e^{\text{ab}}$  **do**  
  **for all** gauss-points  $gp$  **do**  
    Compute jump in the displacements:  $\llbracket \mathbf{u} \rrbracket_{n+1}^{\text{ab}(k)}$   
    Compute trial traction:  $\mathbf{t}_{n+1}^{\text{abtrial}, (k)} = \langle \boldsymbol{\sigma}_{n+1}^{(k)} \rangle_{\gamma_{\text{ab}}} \mathbf{n}^{\text{ab}} - \alpha_e^{\text{ab}} \mathbf{I} \llbracket \mathbf{u} \rrbracket_{n+1}^{\text{ab}(k)}$   
    Compute trial yield function:  $\phi_{n+1}^{\text{abtrial}, (k)} = \phi^{\text{ab}}(t_{\tau, n+1}^{\text{abtrial}, (k)}) = \|t_{\tau, n+1}^{\text{abtrial}, (k)}\| - t_Y^{\text{ab}}$   
  
    **if** yield condition is not violated:  $\phi_{n+1}^{\text{abtrial}, (k)} \leq 0$  **then**  
      Trial state is true state:  $\mathbf{t}_{n+1}^{\text{ab}(k)} = \mathbf{t}_{n+1}^{\text{abtrial}, (k)}$   
    **else**  
      Normal direction - no yielding:  $t_{N, n+1}^{\text{ab}(k)} = t_{N, n+1}^{\text{abtrial}, (k)}$   
      Tangential direction - return mapping algorithm:  
      
$$\Delta\beta = \frac{\phi_{n+1}^{\text{abtrial}, (k)}}{\alpha_e^{\text{ab}}}$$
      
$$t_{\tau, n+1}^{\text{ab}(k)} = t_{\tau, n+1}^{\text{abtrial}, (k)} - \alpha_e^{\text{ab}} \Delta\beta \frac{t_{\tau, n+1}^{\text{abtrial}, (k)}}{\|t_{\tau, n+1}^{\text{abtrial}, (k)}\|}$$
      
$$\llbracket u \rrbracket_{\tau, n+1}^{\text{abpl}, (k)} = \llbracket u \rrbracket_{\tau, n}^{\text{abpl}} + \Delta\beta \frac{t_{\tau, n+1}^{\text{abtrial}, (k)}}{\|t_{\tau, n+1}^{\text{abtrial}, (k)}\|}$$
      Update  $\alpha_e^{\text{ab}}$   
    **end if**  
  **end for**  
**end for**

---

of equations using a Newton-Raphson iterative scheme. We provide expressions for the tangent matrix at the (n+1)-th load step and k-th iteration but omit the iteration and load counters to simplify notation.

*Tangent matrix for an element with a junction*

Consider an element with three local domains  $\Omega_e^p$ ,  $\Omega_e^q$ , and  $\Omega_e^r$  and three intersecting interfaces  $\Gamma_e^{\text{pq}}$ ,  $\Gamma_e^{\text{pr}}$ , and  $\Gamma_e^{\text{qr}}$  as shown in Figure A.1. For such an element, the tangent matrix can be written as:

$$\mathbf{k} = \begin{bmatrix} \mathbf{k}^{\text{b,p}} + \mathbf{k}_d^{\text{c,p}} & \mathbf{k}_{\text{od,pq}}^{\text{c,p}} & \mathbf{k}_{\text{od,pr}}^{\text{c,p}} \\ \mathbf{k}_{\text{od,pq}}^{\text{c,q}} & \mathbf{k}^{\text{b,q}} + \mathbf{k}_d^{\text{c,q}} & \mathbf{k}_{\text{od,qr}}^{\text{c,q}} \\ \mathbf{k}_{\text{od,pr}}^{\text{c,r}} & \mathbf{k}_{\text{od,qr}}^{\text{c,r}} & \mathbf{k}^{\text{b,r}} + \mathbf{k}_d^{\text{c,r}} \end{bmatrix}, \quad (\text{A.3})$$

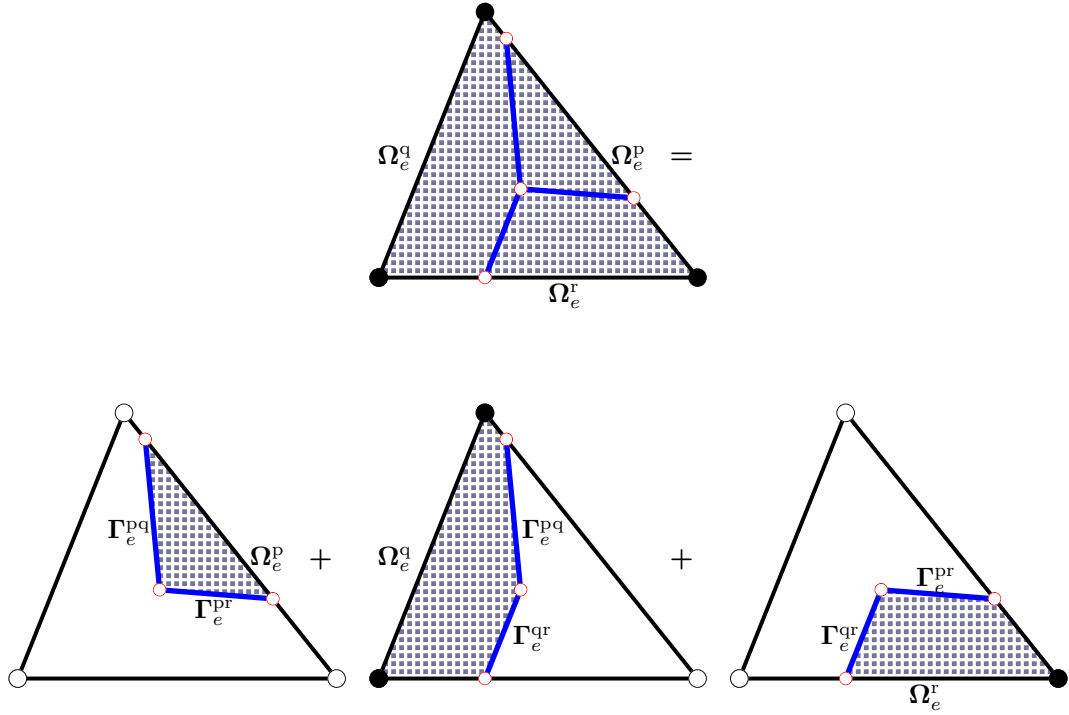


FIGURE A.1 – Representative element with three intersecting interfaces separating the local domains  $\Omega_e^p$ ,  $\Omega_e^q$  and  $\Omega_e^r$ .

Here,  $\mathbf{k}^{b,m}$  represents the bulk contribution to the element tangent matrix and is given as:

$$\mathbf{k}^{b,m} = \int_{\Omega_e^m} \mathbf{B}^T \mathbf{D}^m \mathbf{B} \, d\Omega_e \quad \text{for } m = p, q, r. \quad (\text{A.4})$$

The other terms result from linearization of contact tractions and depend on whether the interface is in the stick-state or slip-state. As discussed in Chapter 5, a trial-state return-mapping algorithm is used to determine if the interface is sliding and to update tractions. The contribution to the leading-diagonal block from linearization of contact tractions is denoted by  $\mathbf{k}_d^{c,m}$  for  $m = p, q$  and  $r$ . For a perfectly-tied interface, these local matrices are computed from:

$$\begin{aligned}
\mathbf{k}_d^{c,p} &= \int_{\Gamma_e^{pq}} \mathbf{N}^T \boldsymbol{\alpha}_e^{pq} \mathbf{N} \, d\Gamma_e - \gamma_{pq}^{1e} \int_{\Gamma_e^{pq}} \mathbf{N}^T (\mathbf{n}^{pq})^T \mathbf{D}^p \mathbf{B} \, d\Gamma_e \\
&+ \int_{\Gamma_e^{pr}} \mathbf{N}^T \boldsymbol{\alpha}_e^{pr} \mathbf{N} \, d\Gamma_e - \gamma_{pr}^{1e} \int_{\Gamma_e^{pr}} \mathbf{N}^T (\mathbf{n}^{pr})^T \mathbf{D}^p \mathbf{B} \, d\Gamma_e \\
\mathbf{k}_d^{c,q} &= \int_{\Gamma_e^{pq}} \mathbf{N}^T \boldsymbol{\alpha}_e^{pq} \mathbf{N} \, d\Gamma_e - \gamma_{pq}^{2e} \int_{\Gamma_e^{pq}} \mathbf{N}^T (\mathbf{n}^{qp})^T \mathbf{D}^q \mathbf{B} \, d\Gamma_e \\
&+ \int_{\Gamma_e^{qr}} \mathbf{N}^T \boldsymbol{\alpha}_e^{qr} \mathbf{N} \, d\Gamma_e - \gamma_{qr}^{1e} \int_{\Gamma_e^{qr}} \mathbf{N}^T (\mathbf{n}^{qr})^T \mathbf{D}^q \mathbf{B} \, d\Gamma_e \\
\mathbf{k}_d^{c,r} &= \int_{\Gamma_e^{pr}} \mathbf{N}^T \boldsymbol{\alpha}_e^{pr} \mathbf{N} \, d\Gamma_e - \gamma_{pr}^{2e} \int_{\Gamma_e^{pr}} \mathbf{N}^T (\mathbf{n}^{rp})^T \mathbf{D}^r \mathbf{B} \, d\Gamma_e \\
&+ \int_{\Gamma_e^{qr}} \mathbf{N}^T \boldsymbol{\alpha}_e^{qr} \mathbf{N} \, d\Gamma_e - \gamma_{qr}^{2e} \int_{\Gamma_e^{qr}} \mathbf{N}^T (\mathbf{n}^{rq})^T \mathbf{D}^r \mathbf{B} \, d\Gamma_e
\end{aligned} \tag{A.5}$$

By  $\mathbf{k}_{od,pq}^{c,p}$ , we denote the contribution to the off-diagonal block resulting from the linearization of contact tractions on interface  $\Gamma_e^{pq}$  for the child element with physical volume  $\Omega_e^p$ . We follow the same convention in naming the other off-diagonal contributions. While in perfect contact, these off-diagonal contributions are given as:

$$\begin{aligned}
\mathbf{k}_{od,pq}^{c,p} &= - \int_{\Gamma_e^{pq}} \mathbf{N}^T \boldsymbol{\alpha}_e^{pq} \mathbf{N} \, d\Gamma_e + \gamma_{pq}^{2e} \int_{\Gamma_e^{pq}} \mathbf{N}^T (\mathbf{n}^{qp})^T \mathbf{D}^q \mathbf{B} \, d\Gamma_e, \\
\mathbf{k}_{od,pr}^{c,p} &= - \int_{\Gamma_e^{pr}} \mathbf{N}^T \boldsymbol{\alpha}_e^{pr} \mathbf{N} \, d\Gamma_e + \gamma_{pr}^{2e} \int_{\Gamma_e^{pr}} \mathbf{N}^T (\mathbf{n}^{rp})^T \mathbf{D}^r \mathbf{B} \, d\Gamma_e, \\
\mathbf{k}_{od,pq}^{c,q} &= - \int_{\Gamma_e^{pq}} \mathbf{N}^T \boldsymbol{\alpha}_e^{pq} \mathbf{N} \, d\Gamma_e + \gamma_{pq}^{1e} \int_{\Gamma_e^{pq}} \mathbf{N}^T (\mathbf{n}^{pq})^T \mathbf{D}^p \mathbf{B} \, d\Gamma_e, \\
\mathbf{k}_{od,qr}^{c,q} &= - \int_{\Gamma_e^{qr}} \mathbf{N}^T \boldsymbol{\alpha}_e^{qr} \mathbf{N} \, d\Gamma_e + \gamma_{qr}^{2e} \int_{\Gamma_e^{qr}} \mathbf{N}^T (\mathbf{n}^{rq})^T \mathbf{D}^r \mathbf{B} \, d\Gamma_e, \\
\mathbf{k}_{od,pr}^{c,r} &= - \int_{\Gamma_e^{pr}} \mathbf{N}^T \boldsymbol{\alpha}_e^{pr} \mathbf{N} \, d\Gamma_e + \gamma_{pr}^{1e} \int_{\Gamma_e^{pr}} \mathbf{N}^T (\mathbf{n}^{pr})^T \mathbf{D}^p \mathbf{B} \, d\Gamma_e, \\
\mathbf{k}_{od,qr}^{c,r} &= - \int_{\Gamma_e^{qr}} \mathbf{N}^T \boldsymbol{\alpha}_e^{qr} \mathbf{N} \, d\Gamma_e + \gamma_{qr}^{1e} \int_{\Gamma_e^{qr}} \mathbf{N}^T (\mathbf{n}^{qr})^T \mathbf{D}^q \mathbf{B} \, d\Gamma_e.
\end{aligned} \tag{A.6}$$



The stabilization parameters in the normal and tangential directions are evaluated as described in Chapter 5 of this study. While the interface is still under perfect contact, the stabilization parameters in both normal and tangential directions are chosen identically. Both the stabilization parameter and the interfacial weights are computed as described in Section 6.3.1. When the interface begins sliding, the stiffness is evaluated from the particular form of the interfacial law. The algorithm described in Chapter 5 for elements with a single embedded interface can be generalized to handle elements with multiple interfaces by looping over all the interfaces local to the current element (see Algorithm 5).

# Bibliography

- Annavarapu, C., Hautefeuille, M., and Dolbow, J. E. (2012a), “A robust Nitsche’s formulation for interface problems,” *Computer Methods in Applied Mechanics and Engineering*, 225–228, 44 – 54.
- Annavarapu, C., Hautefeuille, M., and Dolbow, J. E. (2012b), “Stable imposition of stiff constraints in explicit dynamics for embedded finite element methods,” *International Journal for Numerical Methods in Engineering*, 92, 206–228.
- Annavarapu, C., Hautefeuille, M., and Dolbow, J. E. (2013a), “A Nitsche stabilized finite element method for frictional sliding problems. Part I: Single interface,” *under review*.
- Annavarapu, C., Hautefeuille, M., and Dolbow, J. E. (2013b), “A Nitsche stabilized finite element method for frictional sliding problems. Part II: Intersecting interfaces,” *under review*.
- Aragón, A. M., Duarte, C. A., and Geubelle, P. H. (2010), “Generalized finite element enrichment functions for discontinuous gradient fields,” *International Journal for Numerical Methods in Engineering*, 82, 242–268.
- Arnold, D., Brezzi, F., Cockburn, B., and Marini, L. (2002), “Unified analysis of discontinuous Galerkin methods for elliptic problems,” *SIAM journal on numerical analysis*, pp. 1749–1779.
- Babuška, I. (1973), “The Finite Element Method with Lagrange Multipliers,” *Numer. Math.*, 20, 179–192.
- Babuška, I. and Melenk, J. M. (1997), “The partition of unity method,” *Int. J. Numer. Meth. Engng.*, 40, 727–758.
- Baiges, J., Codina, R., Henke, F., Shahmiri, S., and Wall, W. A. (2012), “A symmetric method for weakly imposing Dirichlet boundary conditions in embedded finite element meshes,” *International Journal for Numerical Methods in Engineering*, 90, 636–658.

- Barbosa, H. J. C. and Hughes, T. J. R. (1991), “The finite element method with Lagrange multipliers on the boundary: circumventing the Babuska-Brezzi condition,” *Computer Methods in Applied Mechanics and Engineering*, 85, 109 – 128.
- Béchet, E., Moës, N., and Wohlmuth, B. (2009), “A stable Lagrange multiplier space for stiff interface conditions within the extended finite element method,” *Int. J. Numer. Meth. Engng.*, 78, 931–954.
- Becker, R., Hansbo, P., and Stenberg, R. (2003), “A finite element method for domain decomposition with non-matching grids,” *ESAIM: Mathematical Modelling and Numerical Analysis*, 37, 209–225.
- Becker, R., Burman, E., and Hansbo, P. (2011), “A hierarchical NXFEM for fictitious domain simulations,” *International Journal for Numerical Methods in Engineering*, 86, 549–559.
- Bedrossian, J., von Brecht, J. H., Zhu, S., Sifakis, E., and Teran, J. M. (2010), “A second order virtual node method for elliptic problems with interfaces and irregular domains,” *J. Comput. Phys.*, 229, 6405–6426.
- Belytschko, T., Liu, W., and Moran, B. (2000), *Nonlinear finite elements for continua and structures*, Wiley.
- 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, 609–635.
- Bhat, H. S., Olives, M., Dmowska, R., and Rice, J. R. (2007), “Role of fault branches in earthquake rupture dynamics,” *Journal of Geophysical Research: Solid Earth*, 112, n/a–n/a.
- Brezzi, F. (1974), “Existence, Uniqueness and Approximation of Saddle-point problems arising from Lagrange Multipliers,” *RAIRO-Oper. Res.*, 8, 129–151.
- Brezzi, F. and Fortin, M. (1991), *Mixed and hybrid finite element methods*, Springer-Verlag New York, Inc., New York, NY, USA.
- 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, 2680 – 2686.
- Burman, E. and Hansbo, P. (2012), “Fictitious domain finite element methods using cut elements: II. A stabilized Nitsche method,” *Applied Numerical Mathematics*, 62, 328 – 341.

- Burman, E. and Zunino, P. (2006), “A domain decomposition method based on weighted interior penalties for advection-diffusion-reaction problems,” *SIAM J. Numer. Anal.*, 44, 1612–1638.
- Cai, Z., Ye, X., and Zhang, S. (2011), “Discontinuous Galerkin Finite Element Methods for Interface Problems: A Priori and A Posteriori Error Estimations,” *SIAM J. Numer. Anal.*, 49, 1761–1787.
- Carey, G. F., Chow, S. S., and Seager, M. K. (1985), “Approximate boundary-flux calculations,” *Comput. Methods Appl. Mech. Engrg.*, 50, 107–120.
- Carpenter, N. J., Taylor, R. L., and Katona, M. G. (1991), “Lagrange constraints for transient finite element surface contact,” *Int. J. Numer. Meth. Engng.*, 32, 103–128.
- Chern, I.-L. and Shu, Y.-S. (2007), “A coupling interface method for elliptic interface problems,” *J. Comput. Phys.*, 225, 2138–2174.
- Chessa, J., Smolinski, P., and Belytschko, T. (2002), “The extended finite element method (XFEM) for solidification problems,” *International Journal for Numerical Methods in Engineering*, 53, 1959–1977.
- Codina, R. and Baiges, J. (2009), “Approximate imposition of boundary conditions in immersed boundary methods,” *International Journal for Numerical Methods in Engineering*, 80, 1379–1405.
- Combescure, A., Gravouil, A., Baietto-Dubourg, M.-C., Elguedj, E., Ribeaucourt, R., and Ferrié, E. (2005), “Extended finite element method for numerical simulation of 3D fatigue crack growth,” in *Life Cycle Tribology Proceedings of the 31st Leeds-Lyon Symposium on Tribology Held at Trinity and All Saints College, Horsforth, Leeds, UK 7th-10th September 2004*, eds. G. D. D. Dowson, M. Priest and A. Lubrecht, vol. 48 of *Tribology and Interface Engineering Series*, pp. 323 – 328, Elsevier.
- Coon, E., Shaw, B., and Spiegelman, M. (2011), “A Nitsche-extended finite element method for earthquake rupture on complex fault systems,” *Computer Methods in Applied Mechanics and Engineering*, 200, 2859 – 2870.
- Coroneos, R. M. (2012), “Structural Analysis and Optimization of a Composite Fan Blade for a Future Aircraft Engine,” *NASA Glenn Research Center, Cleveland, Ohio, Internal Report*.
- Crisfield, M. A. (2000), “Re-visiting the contact patch test,” *International Journal for Numerical Methods in Engineering*, 48, 435–449.
- Curtis, J. B. (2002), “Fractured Shale-Gas Systems,” *AAPG Bulletin*, 86, 1921–1938.

- 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, 1741–1760.
- de Borst, R., Remmers, J. J., and Needleman, A. (2006), “Mesh-independent discrete numerical representations of cohesive-zone models,” *Engineering Fracture Mechanics*, 73, 160 – 177.
- Dolbow, J. E. and Franca, L. P. (2008), “Residual-free bubbles for embedded Dirichlet problems,” *Comput. Methods Appl. Mech. Engrg.*, 197, 3751–3759.
- Dolbow, J. E. and Harari, I. (2009), “An efficient finite element method for embedded interface problems,” *Int. J. Numer. Meth. Engng.*, 78, 229–252.
- Dolbow, J. E., Moës, N., and Belytschko, T. (2000), “Discontinuous enrichment in finite elements with a partition of unity method,” *Finite Elem. Anal. Des.*, 36, 235–260.
- Dolbow, J. E., Moës, N., and Belytschko, T. (2001), “An extended finite element method for modeling crack growth with frictional contact,” *Comput. Methods Appl. Mech. Engrg.*, 190, 6825–6846.
- Durand-Riard, P., Caumon, G., and Muron, P. (2010), “Balanced restoration of geological volumes with relaxed meshing constraints,” *Computers and Geosciences*, 36, 441 – 452.
- Elguedj, T., Gravouil, A., and Combescure, A. (2007), “A mixed augmented Lagrangian-extended finite element method for modelling elasticplastic fatigue crack growth with unilateral contact,” *International Journal for Numerical Methods in Engineering*, 71, 1569–1597.
- Elguedj, T., Gravouil, A., and Maigre, H. (2009), “An explicit dynamics extended finite element method. Part 1: Mass lumping for arbitrary enrichment functions,” *Computer Methods in Applied Mechanics and Engineering*, 198, 2297 – 2317.
- Embar, A., Dolbow, J. E., and Harari, I. (2010), “Imposing Dirichlet boundary conditions with Nitsche’s method and spline-based finite elements,” *Int. J. Numer. Meth. Engng.*, 83, 877–898.
- Ern, A., Stephansen, A. F., and Zunino, P. (2009), “A discontinuous Galerkin method with weighted averages for advection-diffusion equations with locally small and anisotropic diffusivity,” *IMA J. Numer. Anal.*, 29, 235–256.
- Espinosa, H. D. and Zavattieri, P. D. (2003), “A grain level model for the study of failure initiation and evolution in polycrystalline brittle materials. Part I: Theory and numerical implementation,” *Mechanics of Materials*, 35, 333 – 364.

- Fedkiw, R. P., Aslam, T., Merriman, B., and Osher, S. (1999), “A Non-oscillatory Eulerian Approach to Interfaces in Multimaterial Flows (the Ghost Fluid Method),” *Journal of Computational Physics*, 152, 457 – 492.
- Gansner, E. R. and North, S. C. (2000), “An open graph visualization system and its applications to software engineering,” *Sotw. Pract. Expr.*, 30, 1203–1233.
- Gerstenberger, A. and Wall, W. A. (2010), “An embedded Dirichlet formulation for 3D continua,” *International Journal for Numerical Methods in Engineering*, 82, 537–563.
- Geuzaine, C. and Remacle, J.-F. (2009), “Gmsh: A 3-D finite element mesh generator with built-in pre- and post-processing facilities,” *International Journal for Numerical Methods in Engineering*, 79, 1309–1331.
- Ghahremani, F. (1980), “Effect of grain boundary sliding on steady creep of polycrystals,” *International Journal of Solids and Structures*, 16, 847 – 862.
- Glowinski, R., Pan, T.-W., and Périaux, J. (1994), “A fictitious domain method for Dirichlet problem and applications,” *Computer Methods in Applied Mechanics and Engineering*, 111, 283 – 303.
- 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, 1449–1475.
- Grégoire, D., Maigre, H., Réthoré, J., and Combescure, A. (2007), “Dynamic crack propagation under mixed-mode loading Comparison between experiments and X-FEM simulations,” *International Journal of Solids and Structures*, 44, 6517 – 6534.
- Grégoire, D., Maigre, H., and Combescure, A. (2009), “New experimental and numerical techniques to study the arrest and the restart of a crack under impact in transparent materials,” *International Journal of Solids and Structures*, 46, 3480 – 3491.
- Griebel, M. and Alexander, S. M. (2002), “A Particle-Partition of Unity Method—Part V: Boundary Conditions,” in *Geometric Analysis and Nonlinear Partial Differential Equations*, eds. S. Hildebrandt and H. Karcher, pp. 517–540, Springer-Verlag.
- Gudmundsson, A., Berg, S. S., Lyslo, K. B., and Skurtveit, E. (2001), “Fracture networks and fluid transport in active fault zones,” *Journal of Structural Geology*, 23, 343 – 353.

- Hansbo, A. and Hansbo, P. (2002), “An unfitted finite element method, based on Nitsche’s method, for elliptic interface problems,” *Comput. Methods Appl. Mech. Engrg.*, 191, 5537–5552.
- Hansbo, A. and Hansbo, P. (2004), “A finite element method for the simulation of strong and weak discontinuities in solid mechanics,” *Computer Methods in Applied Mechanics and Engineering*, 193, 3523 – 3540.
- Hansbo, A., Hansbo, P., and Larson, M. G. (2003), “A finite element method on composite grids based on Nitsche’s method,” *ESAIM: Mathematical Modelling and Numerical Analysis*, 37, 495–514.
- 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, 40–64.
- Hettich, T. and Ramm, E. (2006), “Interface material failure modeled by the extended finite-element method and level sets,” *Computer Methods in Applied Mechanics and Engineering*, 195, 4753 – 4767.
- Hettich, T., Hund, A., and Ramm, E. (2008), “Modeling of failure in composites by X-FEM and level sets within a multiscale framework,” *Computer Methods in Applied Mechanics and Engineering*, 197, 414 – 424.
- Huang, H., Long, T., Wan, J., and Brown, W. (2011), “On the use of enriched finite element method to model subsurface features in porous media flow problems,” *Computational Geosciences*, 15, 721–736.
- Huet, C. (1990), “Application of variational concepts to size effects in elastic heterogeneous bodies,” *Journal of the Mechanics and Physics of Solids*, 38, 813 – 841.
- Hughes, T. J. R. (2000), *The finite element method: linear static and dynamic finite element analysis*, Dover Publications.
- Hughes, T. J. R. and Liu, W. K. (1978), “Implicit-Explicit Finite Elements in Transient Analysis: Implementation and Numerical Examples,” *Journal of Applied Mechanics*, 45, 375–378.
- Ji, H. and Dolbow, J. E. (2004), “On strategies for enforcing interfacial constraints and evaluating jump conditions with the extended finite element method,” *Int. J. Numer. Meth. Engng.*, 61, 2508–2535.
- Kanit, T., Forest, S., Galliet, I., Mounoury, V., and Jeulin, D. (2003), “Determination of the size of the representative volume element for random composites: statistical and numerical approach,” *International Journal of Solids and Structures*, 40, 3647 – 3679.

- Khoei, A. and Mousavi, S. T. (2010), “Modeling of large deformation – Large sliding contact via the penalty X-FEM technique,” *Computational Materials Science*, 48, 471 – 480.
- Khoei, A. and Nikbakht, M. (2007), “An enriched finite element algorithm for numerical computation of contact friction problems,” *International Journal of Mechanical Sciences*, 49, 183 – 199.
- Khoei, A., Shamloo, A., and Azami, A. (2006), “Extended finite element method in plasticity forming of powder compaction with contact friction,” *International Journal of Solids and Structures*, 43, 5421 – 5448.
- Kim, T. Y., Dolbow, J. E., and Laursen, T. A. (2007), “A mortared finite element method for frictional contact on arbitrary interfaces,” *Comput. Mech.*, 39, 223–235.
- Ladevéze, P. (1999), *Nonlinear Computational Structural Mechanics - New Approaches and Non-Incremental Methods of Calculation*, Springer-Verlag.
- Laursen, T. A. (2003), *Computational Contact and Impact Mechanics: Fundamentals of Modeling Interfacial Phenomena in Nonlinear Finite Element Analysis*, Springer.
- Laursen, T. A., Puso, M. A., and Sanders, J. (2010), “Mortar contact formulations for deformable-deformable contact: Past contributions and new extensions for enriched and embedded interface formulations,” *Comput. Methods Appl. Mech. Engrg.*, In Press, Corrected Proof, –.
- Legrain, G., Cartraud, P., Perreard, I., and Moës, N. (2011), “An X-FEM and level set computational approach for image-based modelling: Application to homogenization,” *International Journal for Numerical Methods in Engineering*, 86, 915–934.
- Leveque, R. J. and Li, Z. (1994), “The Immersed Interface Method for Elliptic Equations with Discontinuous Coefficients and Singular Sources,” *SIAM Journal on Numerical Analysis*, 31, pp. 1019–1044.
- Lew, A. J. and Buscaglia, G. C. (2008), “A discontinuous-Galerkin-based immersed boundary method,” *International Journal for Numerical Methods in Engineering*, 76, 427–454.
- Lew, A. J. and Negri, M. (2011), “Optimal convergence of a discontinuous-Galerkin-based immersed boundary method,” *ESAIM: Mathematical Modelling and Numerical Analysis*, 45, 651–674.
- Li, W. and Zabaras, N. (2009), “A virtual environment for the interrogation of 3D polycrystalline microstructures including grain size effects,” *Computational Materials Science*, 44, 1163 – 1177.



- 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, 1489–1512.
- Liu, F. and Borja, R. I. (2009), “An extended finite element framework for slow-rate frictional faulting with bulk plasticity and variable friction,” *International Journal for Numerical and Analytical Methods in Geomechanics*, 33, 1535–1560.
- Liu, F. and Borja, R. I. (2010a), “Finite deformation formulation for embedded frictional crack with the extended finite element method,” *International Journal for Numerical Methods in Engineering*, 82, 773–804.
- Liu, F. and Borja, R. I. (2010b), “Stabilized low-order finite elements for frictional contact with the extended finite element method,” *Computer Methods in Applied Mechanics and Engineering*, 199, 2456 – 2471.
- Marshall, S. T., Cooke, M. L., and Owen, S. E. (2008), “Effects of Nonplanar Fault Topology and Mechanical Interaction on Fault-Slip Distributions in the Ventura Basin, California,” *Bulletin of the Seismological Society of America*, 98, 1113–1127.
- Mayr, M. (2010), *Different sliding laws on embedded interfaces using Lagrange multipliers, penalty method and Nitsche’s method*, Diplomarbeit, Technische Universität München, Munich, Germany.
- Menouillard, T., Rthor, J., Combescure, A., and Bung, H. (2006), “Efficient explicit time stepping for the eXtended Finite Element Method (X-FEM),” *International Journal for Numerical Methods in Engineering*, 68, 911–939.
- Menouillard, T., Réthoré, J., Moës, N., Combescure, A., and Bung, H. (2008), “Mass lumping strategies for X-FEM explicit dynamics: Application to crack propagation,” *Int. J. Numer. Meth. Engng.*, 74, 447–474.
- Meyers, M., Mishra, A., and Benson, D. (2006), “Mechanical properties of nanocrystalline materials,” *Progress in Materials Science*, 51, 427 – 556.
- Moës, N., Dolbow, J. E., and Belytschko, T. (1999), “A finite element method for crack growth without remeshing,” *Int. J. Numer. Meth. Engng.*, 46, 131–150.
- Moës, N., Cloirec, M., Cartraud, P., and Remacle, J. F. (2003), “A computational approach to handle complex microstructure geometries,” *Computer Methods in Applied Mechanics and Engineering*, 192, 3163 – 3177.
- Moës, N., Béchet, E., and Tourbier, M. (2006), “Imposing Dirichlet boundary conditions in the extended finite element method,” *Int. J. Numer. Meth. Engng.*, 67, 1641–1669.

- Mourad, H. M., Dolbow, J., and Harari, I. (2007), “A bubble-stabilized finite element method for Dirichlet constraints on embedded interfaces,” *Int. J. Numer. Meth. Engng.*, 69, 772–793.
- Nistor, I., Guiton, M. L. E., Massin, P., Moës, N., and Géniaut, S. (2009), “An X-FEM approach for large sliding contact along discontinuities,” *International Journal for Numerical Methods in Engineering*, 78, 1407–1435.
- Nitsche, J. (1971), “Über ein Variationsprinzip zur Lösung von Dirichlet-Problemen bei Verwendung von Teilräumen, die keine Randbedingungen unterworfen sind,” *Abh. Math. Univ. Hamburg*, 36, 9–15.
- Oden, J. and Pires, E. (1984), “Algorithms and numerical results for finite element approximations of contact problems with non-classical friction laws,” *Computers and Structures*, 19, 137 – 147.
- Osher, S. and Fedkiw, R. (2002), *Level Set Methods and Dynamic Implicit Surfaces*, vol. 153, Springer-Verlag.
- Peskin, C. S. (2002), “The immersed boundary method,” *Acta Numerica*, 11, 479–517.
- Pitkäranta, J. (1979), “Boundary subspaces for the finite element method with Lagrange multipliers,” *Numerische Mathematik*, 33, 273–289.
- Pitkäranta, J. (1980), “Local stability conditions for the Babuka method of Lagrange multipliers,” *Mathematics of Computation*, 35, 1113–1129.
- Pitkäranta, J. (1981), “The finite element method with Lagrange multipliers for domains with corners,” *Mathematics of Computation*, 37, 13–30.
- Plesch, A., Shaw, J. H., Benson, C., Bryant, W. A., Carena, S., Cooke, M., Dolan, J., Fuis, G., Gath, E., Grant, L., Hauksson, E., Jordan, T., Kamerling, M., Legg, M., Lindvall, S., Magistrale, H., Nicholson, C., Niemi, N., Oskin, M., Perry, S., Planansky, G., Rockwell, T., Shearer, P., Sorlien, C., Sss, M. P., Suppe, J., Treiman, J., and Yeats, R. (2007), “Community Fault Model (CFM) for Southern California,” *Bulletin of the Seismological Society of America*, 97, 1793–1802.
- Rangarajan, R., Lew, A., and Buscaglia, G. C. (2009), “A discontinuous-Galerkin-based immersed boundary method with non-homogeneous boundary conditions and its application to elasticity,” *Computer Methods in Applied Mechanics and Engineering*, 198, 1513 – 1534.
- Remmers, J. J. C., Wells, G. N., and de Borst, R. (2001), “Analysis of delamination growth with discontinuous elements,” *Solids, Structures and Coupled Problems in Engineering, Proceedings of the Second European Conference on Computational Mechanics*, 2, 906 – 907.

- Remmers, J. J. C., Borst, R. d., and Needleman, A. (2003), “A cohesive segments method for the simulation of crack growth,” *Computational Mechanics*, 31, 69–77.
- Robbins, J. and Voth, T. (2007), “An extended finite element formulation for modeling the response of polycrystalline materials to shock loading,” *15<sup>th</sup> APS Topical Conference on Shock Compression of Condensed Matter, Kohla Coast, HI, 24-29 June*.
- Rooney, M., Roberts, J. C., Murray, G. M., and Romensko, B. M. (2000), “Advanced Materials: Challenges and Opportunities,” *Johns Hopkins APL Technical Digest*, 21, 516–527.
- Rozycki, P., Moes, N., Bechet, E., and Dubois, C. (2008), “X-FEM explicit dynamics for constant strain elements to alleviate mesh constraints on internal or external boundaries,” *Computer Methods in Applied Mechanics and Engineering*, 197, 349 – 363.
- Sanders, J. D., Dolbow, J. E., and Laursen, T. A. (2009), “On the methods for stabilizing constraints over enriched interfaces in elasticity,” *Int. J. Numer. Meth. Engng.*, 78, 1009–1036.
- Sanders, J. D., Laursen, T. A., and Puso, M. A. (2012), “A Nitsche embedded mesh method,” *Computational Mechanics*, 49, 243–257.
- Sanyal, S. and Butler, S. J. (2005), “An Analysis of Power Generation Prospects from Enhanced Geothermal Systems,” *Proceedings of World Geothermal Congress*.
- Shabir, Z., der Giessen, E. V., Duarte, C., and Simone, A. (2011), “The role of cohesive properties on intergranular crack propagation in brittle polycrystals,” *Modelling and Simulation in Materials Science and Engineering*, 19, 35006–35026.
- Siavelis, M., Guiton, M. L. E., Massin, P., and Moës, N. (2012), “Large sliding contact along branched discontinuities with X-FEM,” *Computational Mechanics*, pp. 1–19.
- Simo, J. and Hughes, T. (1998), *Computational Inelasticity*, Interdisciplinary applied mathematics: Mechanics and materials, Springer.
- Simo, J. and Laursen, T. (1992), “An augmented lagrangian treatment of contact problems involving friction,” *Computers and Structures*, 42, 97 – 116.
- Simone, A. (2004), “Partition of unity-based discontinuous elements for interface phenomena: computational issues,” *Commun. Numer. Meth. Engng.*, 20, 465–478.
- Simone, A., Duarte, C. A., and Van der Giessen, E. (2006), “A Generalized Finite Element Method for polycrystals with discontinuous grain boundaries,” *International Journal for Numerical Methods in Engineering*, 67, 1122–1145.

- Stenberg, R. (1998), “Mortaring by a method of JA Nitsche,” *Computational mechanics, New Trends and Applications, CIMNE, Barcelona, Spain*.
- Sukumar, N., Srolovitz, D. J., Baker, T. J., and Prévost, J.-H. (2003), “Brittle fracture in polycrystalline microstructures with the extended finite element method,” *International Journal for Numerical Methods in Engineering*, 56, 2015–2037.
- Tester, J. W., Anderson, B., Batchelor, A., Blackwell, D., DiPippo, R., Drake, E., Garnish, J., Livesay, B., Moore, M., Nichols, K., et al. (2006), “The future of geothermal energy: Impact of enhanced geothermal systems (EGS) on the United States in the 21st century,” *Massachusetts Institute of Technology, DOE contract DE-AC07-05ID14517 final report*.
- Warner, D. and Molinari, J. (2006), “Micromechanical finite element modeling of compressive fracture in confined alumina ceramic,” *Acta Materialia*, 54, 5135 – 5145.
- Wei, Y. and Anand, L. (2004), “Grain-boundary sliding and separation in polycrystalline metals: application to nanocrystalline fcc metals,” *Journal of the Mechanics and Physics of Solids*, 52, 2587 – 2616.
- Wriggers, P. (2006), *Computational Contact Mechanics*, Springer.
- Wriggers, P. and Zavarise, G. (2008), “A formulation for frictionless contact problems using a weak form introduced by Nitsche,” *Comput. Mech.*, 41, 407–420.
- Wriggers, P., Van, T. V., and Stein, E. (1990), “Finite element formulation of large deformation impact-contact problems with friction,” *Computers and Structures*, 37, 319 – 331.
- Yvonnet, J., Quang, H., and He, Q.-C. (2008), “An XFEM/level set approach to modelling surface/interface effects and to computing the size-dependent effective properties of nanocomposites,” *Computational Mechanics*, 42, 119–131.
- Zohdi, T. I. and Wriggers, P. (2005), *An introduction to computational micromechanics*, vol. 20, Springer.
- Zunino, P. (2009), “Discontinuous Galerkin Methods Based on Weighted Interior Penalties for Second Order PDEs with Non-smooth Coefficients,” *J. Sci. Comput.*, 38, 99–126.
- Zunino, P., Cattaneo, L., and Colciago, C. M. (2011), “An unfitted interface penalty method for the numerical approximation of contrast problems,” *Appl. Numer. Math.*, 61, 1059–1076.

# Biography

Srinivas Chandrasekhar Annavarapu was born in Raipur, India on December 10<sup>th</sup>, 1985. He graduated from the Indian Institute of Technology, Madras (IIT-M) in Chennai, India with a B. S. in Civil Engineering in 2007. In the fall of 2007, he joined the Department of Civil and Environmental Engineering as a graduate fellow and received his doctorate in April, 2013.

In the fall of 2012, he was awarded the Mahato fellowship by the Pratt School of Engineering at Duke University. He is also the recipient of Travel fellowships from the International Center of Numerical Methods in Engineering (CIMNE), Barcelona and United States Association of Computational Mechanics (USACM) in summer 2012.

While at Duke, he has authored and co-authored several peer-reviewed journal articles, such as, “Robust imposition of Dirichlet boundary conditions on embedded surfaces.” (Hautefeuille et al., 2012), “Stable imposition of stiff constraints in explicit dynamics for embedded finite element methods.” (Annavarapu et al., 2012b), and “A robust Nitsche’s formulation for interface problems.” (Annavarapu et al., 2012a). In addition, the journal articles, ”A Nitsche Stabilized Embedded Finite Element Method for Frictional Sliding Problems. Part I: Single Interface” (Annavarapu et al., 2013a) and ”A Nitsche Stabilized Embedded Finite Element Method for Frictional Sliding Problems. Part II: Intersecting Interfaces” (Annavarapu et al., 2013b) were also authored by him at Duke University and are currently in revision.